# SKEWIT, BRACKEN, AND KRAKEN:
# METHODS FOR ANALYZING A COMPLEX, BUT INVISIBLE WORLD

by

Jennifer Lu

A dissertation submitted to The Johns Hopkins University in conformity with the requirements for the degree of Doctor of Philosophy

Baltimore, Maryland

December, 2020

# Abstract

As the DNA of the invisible world provides insight into the countless microscopic organisms living amongst us, the integrity of these genomes and the methods by which we analyze them become increasingly important. In the following, I introduce methods for both evaluating genomic integrity and analyzing microbial communities.

For the analysis of bacterial genomes, I developed SkewIT (Skew Index Test) based on GC Skew, a bacterial genome phenomenon wherein the two replication strands of the same chromosome contain different proportions of guanine and cytosine nucleotides. SkewIT calculates a single metric representing the degree of GC skew for a single genome. Applied across 15,000+ complete bacterial genomes, SkewIT quickly detects assembly patterns and highlights potential bacterial mis-assemblies.

Although eukaryotic microorganisms are abundant worldwide and as human pathogens, eukaryotic pathogen genomes are underrepresented in genomic databases and contain significant contamination. I therefore developed a bioinformatics system for eliminating contamination, generating a “clean” eukaryotic pathogen database (EuPathDB-Clean) of nearly 400 genomes. With the final database, I identify eukaryotic pathogens in human samples, demonstrating the increased sensitivity and reduction in false positives of the final database as compared to the originally contaminated genomes.

As metagenomics captures the genomic data of all microbial organisms in any environment, I developed Bracken (Bayesian Reestimation of Abundance after Classification with KrakEN) for a quick and accurate characterization of the full microbial environment. Bracken uses the taxonomic assignments made by Kraken, a very fast read-level classifier, along with information about the genomes themselves to estimate abundance at the species level, the genus level, or above. I demonstrate that Bracken produces accurate abundance estimates even when a sample contains multiple near-identical species for both shotgun metagenomics projects and for 16S ribosomal RNA (rRNA) bacterial projects.

SkewIT, Bracken, and EuPathDB-Clean are all publicly available for use in future metagenomics projects.

Primary Reader: Professor Steven L. Salzberg, Ph.D.

Secondary Reader: Professor Ben Langmead, Ph.D.

# Acknowledgments

Throughout my academic journey, a number of people have contributed to both my academic success and personal growth. Here, I endeavor to recognize them all for their support, guidance, encouragement, and company as I overcome numerous obstacles over the last few years.

I would first like to recognize and thank my supervisor, **Professor Steven L Salzberg**, without whom none of this would be possible. Thank you for introducing me to a world where my computational skills have tangible benefits and where my developed software has far-reaching uses. Thank you for brainstorming research ideas in order to find the perfect ones that aligned with both my skills and passions. And I am especially thankful to have grown immensely in my writing and presentation skills, learning by experience from writing scientific papers alongside you and watching you convey varied scientific results to different audiences.

Thank you to **Professor Ben Langmead** for being the first to introduce me to the world of computational genomics before I even knew I wanted to apply for a Ph.D. Thank you for being a wonderful professor and later, advisor, teaching me how to use the simplest of coding languages to execute complex algorithms that are the backbone to well-known and widely-used genomics software. Thank you for your persistent, never-ending guidance and leadership with all things relating to Kraken and Kraken 2, taking the initiative to propel these projects forward, allowing them to continue being an integral part of my research experience. Finally, thank you for the many conversations where you provided invaluable advice about choosing a research path and what can lie ahead.

I would like to acknowledge **Professor Winston Timp** for welcoming me as a ”pseudo” lab member, providing me with many, many different ideas of potential projects to pursue and the tools with which to pursue them. Thank you for your honest feedback and support when I began to learn about the world of nanopore sequencing, and for your support even through my failed inner harbor water experiments. Along the same lines, I am incredibly thankful to your lab members (current and former) **Rachael Workman** and **Norah Hilger** especially, for their never-ending patience as I forayed into the world of actual laboratory work with nanopore sequencing and DNA extraction, both for the redwood/wheat genome assembly projects and just for Steven’s genomics class. Despite my limited background in laboratory work, both lab members welcomed me into their space and gave me so much of their time and support to help me be able to say I know how to sequence using nanopore (sometimes).

To **Florian Breitwieser, Ph.D.**, thank you a hundred times over for being the best post-doctoral mentor I could have ever asked for. From even before I joined the lab, you became a mentor to me, being incredibly patient when I knew absolutely nothing. Thank you for being a constant inspiration, through your amazing achievements prior to your post-doc and through your incredible work ethics. I was and still am in awe of Pavian and the work you put into such an invaluable tool for our lab and I am beyond thankful for your patience and endurance as I asked many many questions, breaking Pavian over and over again. Working alongside you for the infectious disease projects was both intimidating and inspiring, and I am beyond thankful for you for truly making my PhD experience a million times better. Also, you have changed my programming experience forever by just teaching me about tmux, which I could not survive without.

For the rest of the entire **SalzbergLab**; research engineers, fellow students, and their significant others; thank you for supporting me throughout the journey, even when I was tired and encountered various barriers in my life. Thank you for putting up with my seemingly random physical presence and my overwhelming excitement when I do show up to lab. Thank you for sharing in your lives with us, as we ate together, drank together, shared secrets together. But beyond all of that, thank you for supporting me in all of my research endeavors, being present for my various poster sessions or presentations, providing me with new ideas, and being the best listening ears through every moment of my Ph.D.

Throughout these past few years, a few dear friends have made a wonderful and lasting impact on my life and I want to take this moment to also extend my thanks to them. First, thank you to **Jami Cheng Trumbo** for being a wonderful, loving sister to me in every way, for always being there for me in my stressful, difficult times, for cheering me on throughout my graduate experience, for being the best kitty auntie, and for being the best listening ear and food buddy. Next, thank you to **Andrea Kim Jiang**, for being the best study, Starbucks, bubble tea, Korean food friend in the world and for loving on my kitties any chance you get. Thank you for every moment we spent together studying alongside one another or just remembering how to relax by watching whatever random Korean show was on our mind. Thank you for matching me in my excitement and joy over the most ridiculous things and for loving me in all the ways. Thank you to **Alexandriya Emonds**, for being my best BME PhD friend from day one when we bonded over balloons and McDonalds. Although we never discuss work and see each other maybe once a month, I absolutely treasure your unwavering support when I share about my insane life. We haven't known each other for very long, but I know you will always defend me, fight for me, listen to me, never judge me, and always love me. Thank you for always encouraging me and for reminding me to be the best boss PhD student I can be. Lastly, but certainly not least of all, thank you to **Annabelle and Max** for being the most loving, fluffy companions throughout my Ph.D, for staying by my side during my late nights of working, for always welcoming me when I return home, and for being the best study companions in the entire world.

Finally, I extend the most thanks to my family. **Mom and Dad**, I love you both. Without either of you, I could not be where I am today. Although you didn't say much, you both worked countless hours throughout my life to support me. You spent many hours driving back and forth from my apartment to home, just to bring me all the food in the world and to make sure I'm happy, safe, supported, and healthy. You had an immense amount of patience with me, waiting so many years for me to get this degree, but even then, you always told me that you just wanted me to be happy. Also, a million thank yous to my brother, **Jeffrey Lu**, for always pushing me to be a better version of myself and for always showing me how much you care for me.

# Dedication

This thesis is dedicated to my Mom and Dad who always support me, who worked for years to give me the best life possible, who constantly worry about me, who constantly care about me, and who constantly love me. Thank you to my Mom for always trying to feed me, for loving my kitties, for spending all of your time and energy making sure I'm happy and successful. Thank you to my Dad for working nonstop to support me, for spoiling my cats, and for buying all the fruit in the world for me.

# Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

Metagenomics is a rapidly growing field of study, driven in part by our ability to generate enormous amounts of DNA sequence rapidly and inexpensively. Since the human genome was first published in 2001 [1, 2], sequencing technology has become approximately one million times faster and cheaper, making it possible for individual labs to generate as much sequence data as the entire Human Genome Project in just a few days. In the context of metagenomics experiments, this makes it possible to sample a complex mixture of microbes by ”shotgun” sequencing, which involves simply isolating DNA, preparing the DNA for sequencing, and sequencing the mixture as deeply as possible. Shotgun sequencing is relatively unbiased compared to targeted sequencing methods [3], including widely-used 16S ribosomal RNA sequencing, and it has the additional advantage that it captures any species with a DNA-based genome, including eukaryotes that lack a 16S rRNA gene.

Along with the technological advances, the number of available DNA sequences has also grown exponentially over the past decade. Two of the largest and most widely-used nucleotide databases are GenBank [4] (and which is mirrored by the EMBL and DDBJ databases [4, 5]) and RefSeq, a curated subset of GenBank [5]. This rich resource of sequenced genomes now makes it possible to sequence uncultured, unprocessed microbial DNA from almost any environment, ranging from soil to the deep ocean to the human body, and use computational sequence comparisons to identify many of the formerly hidden species in these environments [6].

As sequencing technology continues to improve and the number of genomic sequences continues to grow, the computational tools used for metagenomics have become extremely critical for the accurate analysis of any microbial sample. Throughout my PhD research, I have developed multiple tools and pipelines for metagenomics analysis, improving upon the available genomic databases while providing new tools for analyzing any microbial environment.

## 1.1 Genome Databases for Metagenomics

Bacterial genomes encompass the vast majority of sequencing data available, making up 92% of the genome assemblies in NCBI Genbank database and 94% of the genome assemblies in the curated NCBI Refseq database. For sequences to be entered into RefSeq, curators at NCBI perform both automated and manual checks to ensure minimal contamination and high sequence quality. Despite these efforts, multiple studies have identified contamination in RefSeq and other publicly available genome databases [7–11]. NCBI requires RefSeq assemblies to have an appropriate genome length as compared to existing genomes from the same species, and it labels assemblies as “complete” if the genome exists in one contiguous sequence per chromosome, with no unplaced scaffolds and with all chromosomes present. However, NCBI does not perform additional checks, most of which would be computationally expensive, to ensure that a genome was assembled correctly. Therefore, in **Chapter 2**, I propose a new method, SkewIT (Skew Index Test), for validating bacterial genome assemblies based on the phenomenon of bacterial GC-skew. I applied this method to 15,067 complete bacterial genomes in RefSeq, identifying many potential misassemblies as well as trends in GC-skew that are characteristic of some bacterial clades.

While bacteria have been extensively sequenced, eukaryotic microbes have been lacking in major genomic databases, with sequences for eukaryotic microbes representing only 1% of the number of genomes in Genbank, and only 88 genomes being labeled ”complete genomes”. However, given the importance of such genomic representation, in **Chapter 3**, I endeavored to develop a process for removing contamination from the 390 draft eukaryotic microbial genomes, allowing such genomes to be used in metagenomics projects downstream.

# 1.2 Metagenomics Analysis Tools

## 1.2.1 Kraken, Kraken 2, and Bracken

With the wide range of genomic information available, metagenomics analysis required a fast and accurate method for comparing environmental sequences against the large genomic databases. Therefore, Kraken was developed in 2014 by Wood et al. to quickly and accurately classify each sequencing read by comparing exact-match kmers [12]. However, despite the speed and accuracy of Kraken's classification algorithm, Kraken only provided the most specific classification possible, assigning reads across the taxonomic tree. With some reads ”stranded” at higher taxonomic levels if there lacked a species-specific kmer, analysis of the overall composition of a metagenomics sample proved difficult. Using the accurate Kraken classification information, I developed Bracken (Bayesian Reestimation of Abundance with KrakEN) which calculates species-level or genus-level abundances from the Kraken classification numbers. In **Chapter 4**, I describe and evaluate the Bracken method on both simulated reads and a real skin microbiome sample.

Since the development of Kraken in 2014, the number of genomes available has grown exponentially, resulting in an exponential growth in Kraken database sizes. However, to continue to allow Kraken users to classify their metagenomics samples without requiring even more computational resources, Wood et al. released Kraken 2 in

2018 [13]. Kraken 2 maintained the same accuracy as Kraken 1 and compatibility with Bracken, with decreased database sizes. In **Chapter 4**, I demonstrate Bracken's continued compatibility and accuracy with Kraken 2 while also proving Bracken's accurate abundance estimation in Kraken 2's clade exclusion experiments.

### 1.2.2 16S Sequence Analysis

While Kraken's alignment-free algorithm proved to be highly accurate and fast for shotgun metagenomics sequence analysis, Kraken had not previously been evaluated for 16S ribosomal RNA sequencing analysis. However, the release of Kraken 2 included added support for 16S databases, allowing both Kraken 2 and Bracken to provide fast and accurate analysis of 16S rRNA sequencing samples. Therefore, in **Chapter 5**, I designed and ran experiments to evaluate the accuracy of both Kraken 2 and Bracken when applied to 16S rRNA experiments and compared to the leading 16S classifier, the Quantitative Insights into Microbial Ecology (QIIME) software package [14, 15].

# Chapter 2

# SkewIT: GC skew

**Portions of this chapter originally appeared in:**

J. Lu, S. L. Salzberg "SkewIT: Skew Index Test for detecting mis-assembled bacterial genomes." bioRxiv, 2020.02.27.968214 `https://doi.org/10.1101/2020.02.27.968214`. (*Submitted for publication*).

**Related software:**

`https://jenniferlu717.shinyapps.io/SkewIT/`

`https://github.com/jenniferlu717/SkewIT`

## 2.1 Introduction

Two of the largest and most widely-used nucleotide databases are GenBank [4], which has been a shared repository for more than 25 years (and which is mirrored by the EMBL and DDBJ databases [4,5]), and RefSeq, a curated subset of GenBank [5]. For sequences to be entered into RefSeq, curators at NCBI perform both automated and manual checks to ensure minimal contamination and high sequence quality. Despite these efforts, multiple studies have identified contamination in RefSeq and other publicly available genome databases [7–11]. NCBI requires Refseq assemblies to have an appropriate genome length as compared to existing genomes from the same species, and it labels assemblies as "complete" if the genome exists in one contiguous sequence per chromosome, with no unplaced scaffolds and with all chromosomes present. However, NCBI does not perform additional checks, most of which would be computationally expensive, to ensure that a genome sequence was assembled correctly. In this study, I propose a new method, SkewIT (Skew Index Test), for validating bacterial genome assemblies based on the phenomenon of GC skew. I applied this method to 15,067 complete bacterial genomes in RefSeq, identifying many potential misassemblies as well as trends in GC skew that are characteristic of some bacterial clades.

## Bacterial GC skew

GC skew is a non-homogeneous distribution of nucleotides in bacterial DNA strands first discovered in the mid-1990s [16, 17]. Although double-stranded DNA must contain precisely equal numbers of cytosine (C) and guanine (G) bases, the distribution of these nucleotides along a single strand in bacterial chromosomes may be asymmetric. Analysis of many bacterial chromosomes has revealed two distinct compartments, one that is more G-rich and the other that is more C-rich.

Most bacterial genomes are organized into single, circular chromosomes. Replication of the circular chromosomes begins at a single point known as the origin of replication (*ori*) and proceeds bidirectionally until reaching the replication terminus (*ter*). Because the replication process only adds DNA nucleotides to the 3' end of a DNA strand, it must use two slightly different DNA synthesis methods to allow bidirectional replication of the circular chromosome. The leading strand is synthesized continuously from the 5' to 3' end. The lagging strand, in contrast, is synthesized by first creating small Okazaki DNA fragments [18] that are then added to the growing strand in the 3' to 5' direction.

These two slightly different replication processes lead to different mutational biases. Notably, the DNA polymerase replicating the leading strand has a higher instance of hydrolytic deamination of the cytosine, resulting in C→ T (thymine) mutations [19]. However, the replication mechanisms for the lagging strand have a higher instance of repair of the same C→ T mutation [20]. These differences between the leading and lagging strands result in GC skew, where the leading strand contains more Gs than Cs, while the lagging strand has more Cs than Gs.

Linear bacterial genomes also exhibit GC skew despite the difference in genome organization. For example, DNA replication of *Borrelia burgdorferi* begins at the center of the linear chromosome and proceeds bidirectionally until reaching the chromosome ends [21, 22]. This bidirectional replication shows the same GC skew pattern seen on circular chromosomes.

## Quantitative measurements of GC skew

Since the 1990s, GC skew has been used as a quantitative measure of the guanine and cytosine distribution along a genome sequence, where GC skew is computed using the formula (G-C)/(G+C), where G is the number of guanines and C is the number of cytosines in a fixed-size window [17]. GC skew plots are generated by calculating GC skew in adjacent or overlapping windows across the full length of a bacterial genome [16]. Analysis of these plots confirmed the separation of many bacterial genomes into a leading strand with largely positive GC skew and a lagging strand with negative GC skew. The GC skew effect is strong enough that it can be used to identify, within a few kilobases, the *ori/ter* locations.

GC skew plots then evolved into cumulative skew diagrams, which sum the GC skew value in adjacent windows along the bacterial genome [17]. These diagrams sometimes allow more precise identification of the *ori/ter* locations, where the origin is located at the global minimum and the terminus is at the global maximum.

## GC skew Applications and Analyses

Over the last two decades, researchers have employed both GC skew and cumulative GC skew (CGS) diagrams to analyze bacterial genomes. Initial studies confirmed that GC skew was a strong indicator of the direction of replication in the genomes of *Escherichia coli* [23], *Bacillus subtilis*, *Haemophilus influenzae*, and *Borrelia burgdorferi* [16]. In 1998, Mclean et. al. compared GC skew among 9 bacterial genomes and 3 archaeal genomes, revealing strong GC skew in all 9 bacteria but weak or no GC skew signals in the archaeal genomes [24]. In 2002, Rocha et. al. used CGS to predict ori/ter locations for 15 bacterial genomes [25] and in 2017, Zhang et. al. analyzed GC skew across more than 2000 bacterial genomes [26].

Although GC skew has been used as an indicator of the replication strand in thousands of bacterial genomes, it is rarely used as a means to validate genome assemblies. However, the association between GC skew and replication is strong enough that when a genome has a major mis-assembly such as a translocation or inversion, the GC skew plot is clearly disrupted [27]. While existing mis-assembly detection methods (e.g. QUAST [28], REAPR [29], misFinder [30]) require the reads used in genome assembly and/or a reference sequence, GC skew can indicate a potential mis-assembly from the genome sequence alone.

In this paper, I introduce SkewIT (Skew Index Test) as an efficient method to calculate the degree of GC skew in a genome. The SkewIT allows us to quickly analyze all 15,000+ complete bacterial genomes in NCBI's RefSeq library by assigning each genome a single SkewI (Skew Index) value representing the degree of GC skew. I then use the SkewI value to compare GC skew across bacterial clades without requiring GC skew or CGS diagrams. Below, I demonstrate how the degree of GC skew tends to be conserved within certain bacterial taxa; e.g. *Klebsiella* species have high values of the SkewI, while *Bordetella* have much lower values. During this analysis, I discovered that bacterial genomes with outlier values of SkewIT are highly likely to contain mis-assemblies. Using my newly defined metric, I identify multiple potentially mis-assembled chromosomal sequences in the Refseq library of complete bacterial genomes.

## 2.2 Method

SkewIT quantifies GC skew patterns by assigning a single value between 0 and 1 to the complete chromosomal sequence of a bacterial genome, where higher values indicate greater GC skew, and lower values indicate that no GC skew pattern was detected. **Figure 2.1** illustrates the overall method.

Although many published bacterial genome assemblies set the start of the published assembly (i.e., position 1) at the origin of replication, many other bacterial genomes set coordinate 1 arbitrarily. (Because the genomes are circular, there is no unambiguous choice for the beginning of the sequence. DNA databases only contain linear sequences, and therefore some coordinate must be chosen as position 1.) Therefore, I first "circularize" each bacterial genome of size $L$ by appending the first $L/2$ bases of the genome to the end, resulting in a sequence length of $1.5L$ (**Figure 2.1**). This ensures that the full genome starting from the origin of replication will be contained within one of the subsequences of length L between positions 0 and L/2.

Next, I select a GC skew window size $w$ and split the genome into $1.5L/w$ adjacent windows; e.g., for a 1-megabase genome with a 10-Kb window length, I would create 150 windows. In each window $i \in [1, 2, \cdots 1.5L/w]$, I count the frequency of guanine (G) and cytosine (C) bases. Traditionally, GC skew was calculated for each window using **Equation** (2.1):

$$\text{GC-Skew} = \frac{G - C}{G + C} \tag{2.1}$$

Although the GC skew formula accounts for the relative quantities of G and C bases, my method only evaluates which base is more prominent in each window. **Figure 2.1** demonstrates how I convert the GC skew formula into a simplified version that

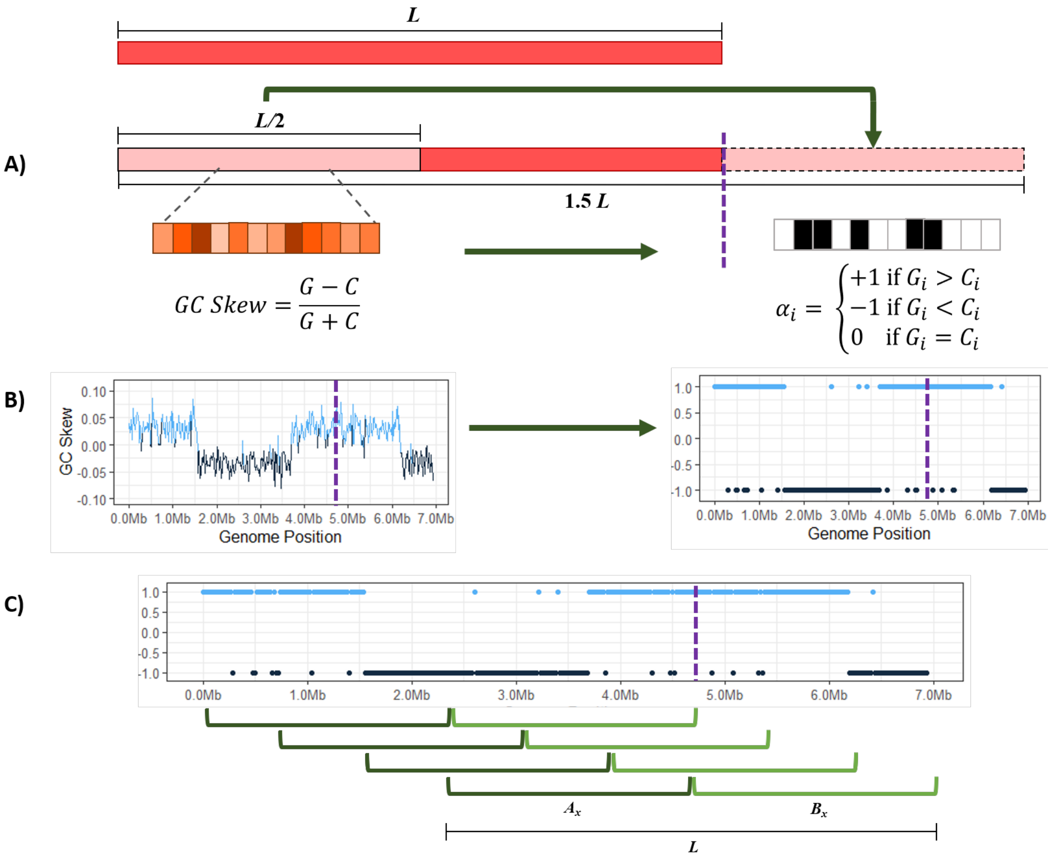

L
L/2
A)
1.5 L
$GC\ Skew = \frac{G-C}{G+C}$
$\alpha_i = \begin{cases} +1 \text{ if } G_i > C_i \\ -1 \text{ if } G_i < C_i \\ 0 \text{ if } G_i = C_i \end{cases}$
B)
GC Skew
0.10
0.05
0.00
-0.05
-0.10
0.0Mb 1.0Mb 2.0Mb 3.0Mb 4.0Mb 5.0Mb 6.0Mb 7.0Mb
Genome Position
1.0
0.5
0.0
-0.5
-1.0
Genome Position
C)
$A_x$
$B_x$
L

Figure 2.1: **The SkewIT algorithm.** A genome of length L is ”circularized” by taking the first half of the sequence (L/2) and concatenating that sequence onto the end of the genome (A). The algorithm then splits the sequence into many shorter windows of length $w$. We assign each window an $\alpha$ value [1,-1,0] based on whether there are more Gs, Cs, or equal quantities of both. (B) The GC skew statistic is shown (left) plotted across the E. coli genome, with a purple dotted line showing where the original sequence ended, prior to concatenating 1/2 of the genome to the end. The plot on the right shows the $\alpha$ value plotted for the same genome. (C) SkewIT finds the location in the genome with the greatest difference in GC skew between the two strands of the genome, by using a pair of sliding windows to find the greatest sum of differences between the $\alpha$ values for the two partitions. SkewIT assumes that the two strands are nearly identical in length, allowing for a difference in partition size up to 8% of the total genome size.

instead assigns each window a score $\alpha_i$ using **Equation** (2.2):

$$\alpha_i = \begin{cases} +1 & \text{if } G_i > C_i \\ -1 & \text{if } G_i < C_i \\ 0 & \text{if } G_i = C_i \end{cases} \tag{2.2}$$

I evaluate the ”skewness” of the genome using a sliding window of size $L$, sliding over one window width at a time. Each window $x \in [1, 2, \cdots 0.5L/w]$ is first split into two equal partitions that each cover 50% of the original genome. I then calculate the sum the $\alpha_i$ values for each partition and determine the absolute difference in sum of GC skew values between the partitions as shown in **Equation** (2.3) and **Figure 2.1**:

$$|A_x - B_x| = \left| \sum_{i=x}^{x+L/2w} \alpha_i - \sum_{i=x+L/2w}^{x+L/w} \alpha_i \right| \tag{2.3}$$

$A_x$ is the sum of the $\alpha$ values within the partition, and $B_x$ is the sum of the $\alpha$ values for the second partition. For example, **Equation** (2.4) shows how I calculate $|A_1 - B_1|$, the skewness for the first sliding window from a genome.

$$|A - B| = \left| \sum_{i=1}^{L/2w} \alpha_i - \sum_{i=L/2w}^{L/w} \alpha_i \right| \tag{2.4}$$

Then, in order to allow for the leading and lagging strands to be slightly different in length, I move the transition point between the two partitions a small distance (4% of the genome length by default) to the left and right, allowing the leading strand to be anywhere between 46% and 54% of the genome length, and recalculating the difference in sums of $\alpha$ values. The transition point is chosen to maximize $|A_x - B_x|$ for this window.

Finally, I determine the maximum value of $|A_x - B_x|$, which gives me the window where the greatest difference exists between the GC content of the two partitions of the genome. In order to be provide a consistent value between 0 and 1 despite genome length $L$ or window size $w$, I define the skew index ($SkewI$) as the following normalized value:

$$SkewI = \frac{w}{L}\text{max}|A_x - B_x| \tag{2.5}$$

## 2.3 Results and Discussion

I applied the SkewIT method to the complete bacterial genomes from NCBI RefSeq Release 97 (released on November 4, 2019). I only evaluated bacterial chromosomes that were $>$ 50,000bp in length and excluded plasmids from this analysis. In total, I tested 15,067 genomes representing 4,471 species and 1,148 genera.

First, I compared SkewI values using the various window sizes $w$ of 10Kb, 15Kb, 20Kb, 25Kb, and 30Kb (**Figure 2.2**). From my analysis, smaller window sizes (10Kb and 15Kb) caused the SkewI values across all bacterial genomes to be lower, as SkewI was more sensitive to local fluctuations in polarity. However, as window sizes become too large, I was no longer able to accurately calculate SkewI for smaller genomes. Therefore, I selected a window size of 20Kb for calculating SkewI across all genomes and for the following analyses.

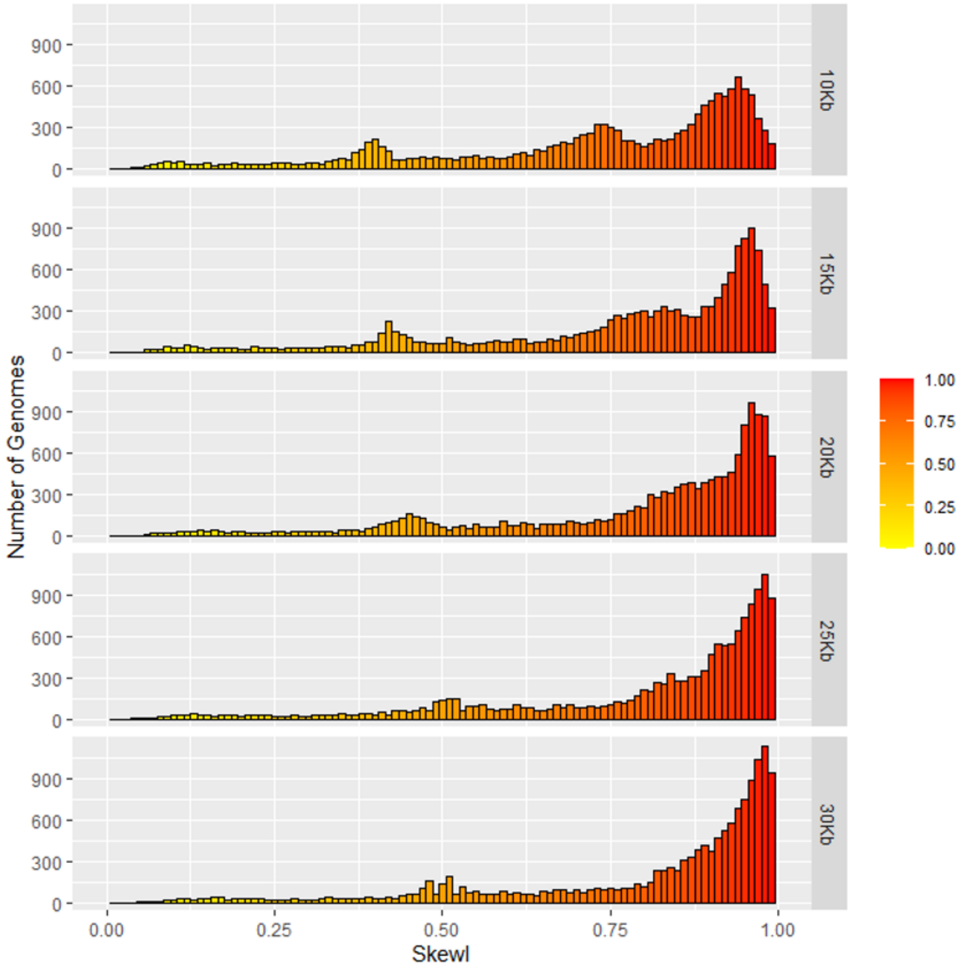


Figure 2.2: **SkewI Comparisons for window sizes 10Kb, 15Kb, 20Kb, 25Kb, 30Kb** This figure compares the full distribution of SkewI values for all 15,067 genomes using different window sizes.

Overall, analysis of all bacteria revealed that most genomes have strong GC skew patterns, with relatively few having SkewI values less than 0.5 (**Figure 2.3**). In order to isolate and analyze bacterial genomes with unusually low SkewI values, I separated the bacterial genomes by clades, revealing characteristic SkewI distributions for individual genera (**Figure 2.4**). For example, genomes from the genera of *Bacillus*,

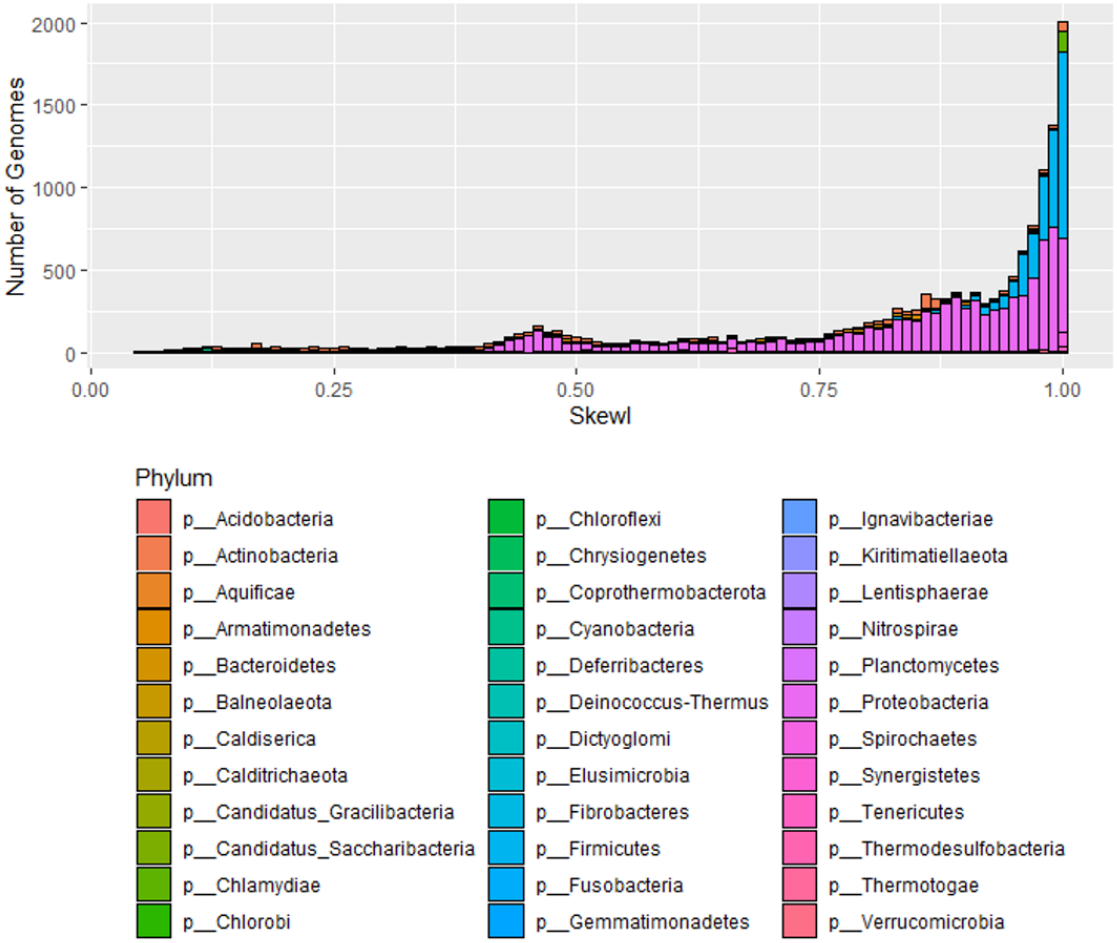


Figure 2.3: **Skew index (SkewI) for all Refseq 97 Bacteria.** This figure displays the full range of SkewI values for all complete bacterial chromosomes in Refseq Release 97, colored by phylum.

*Escherichia*, and *Salmonella* have consistently high SkewI values, with a mean close to 0.9. However, *Bordetella* genomes have far lower SkewI values, with a mean of 0.52. Additionally, while genomes in the *Klebsiella* and *Brucella* genera all have similar SkewI values (and therefore similar amounts of GC skew), genomes from the *Campylobacter* and *Corynebacterium* genera demonstrated much less consistent amounts of GC skew, with a wide range of SkewI values.

Given the differences between genera, I evaluated abnormalities in GC skew by setting a threshold for each genus that would allow me to flag genomes that might have assembly problems. For each genus with 10 or more genomes, I set a SkewI threshold

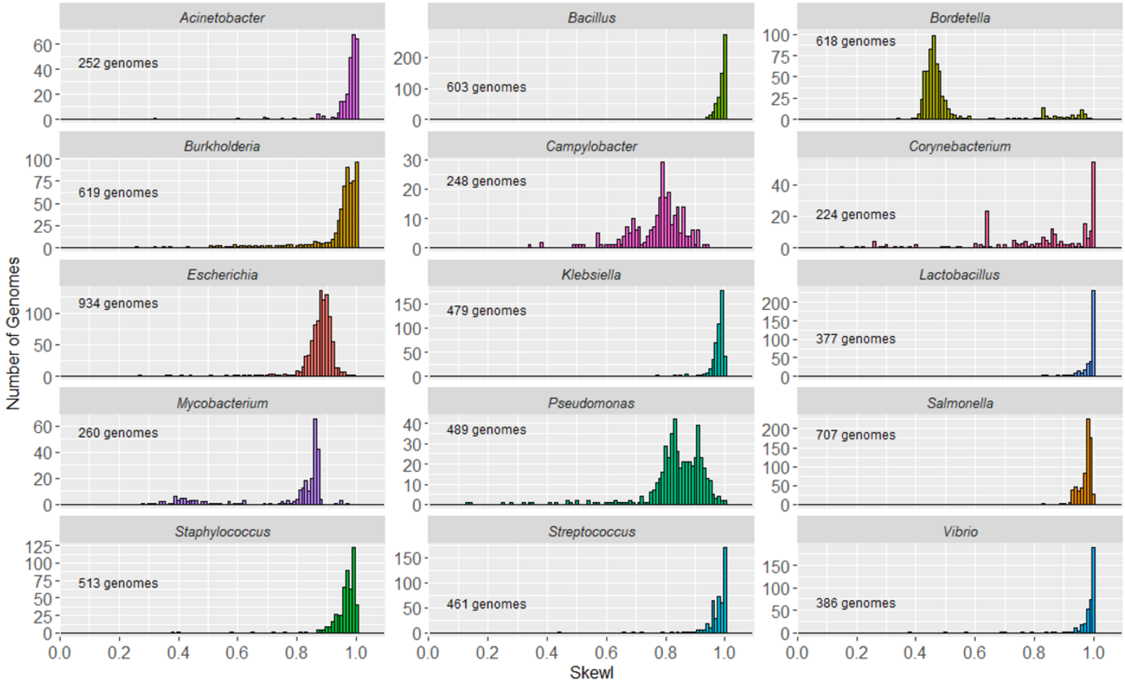


Figure 2.4: **Skew index (SkewI) per genus.** This figure shows the distribution of SkewI values for the 12 bacterial genera with the greatest number of fully sequenced genomes.

at two standard deviations below the mean. If a genome's SkewI exceeded the threshold, then I considered that bacterial genome to be within the expected range for that genera. However, if a genome's SkewI was below the threshold, then I considered that genome to be possibly mis-assembled.

From my analysis, 161 genera of the total 1,148 analyzed contain 10 or more genomes. These 161 genera represent 12,846 of the 15,067 bacterial genomes analyzed, with 423 genomes having SkewI values below the threshold for their particular genus. **Table 2.1** lists the SkewI statistics for the 12 bacterial genera with the greatest number of complete genomes.

Table 2.1: **Average SkewI values for the 12 bacterial genera with the largest number of complete genomes.** The threshold was set at 2 standard deviations below the mean.

| Genus | Genome Count | Mean SkewI | SkewI St. Dev. | SkewI Threshold | Genomes Below Threshold | Mean GC-content (%) |
|---|---|---|---|---|---|---|
| Escherichia | 934 | 0.8729 | 0.0620 | 0.7489 | 30 | 50.68 |
| Salmonella | 707 | 0.9682 | 0.0393 | 0.8896 | 15 | 52.15 |
| Burkholderia | 619 | 0.9323 | 0.1086 | 0.7151 | 39 | 67.42 |
| Bordetella | 618 | 0.5152 | 0.1474 | 0.2204 | 0 | 67.52 |
| Bacillus | 603 | 0.9848 | 0.04452 | 0.8957 | 10 | 41.31 |
| Staphylococcus | 513 | 0.9605 | 0.0538 | 0.8530 | 10 | 33.13 |
| Pseudomonas | 489 | 0.8359 | 0.1095 | 0.6170 | 20 | 63.09 |
| Klebsiella | 479 | 0.9746 | 0.03153 | 0.9115 | 17 | 57.23 |
| Streptococcus | 461 | 0.9743 | 0.0451 | 0.8840 | 12 | 33.44 |
| Vibrio | 386 | 0.9802 | 0.0559 | 0.8685 | 9 | 45.69 |
| Lactobacillus | 377 | 0.9799 | 0.0612 | 0.8574 | 11 | 42.99 |
| Mycobacterium | 260 | 0.7589 | 0.1730 | 0.4129 | 21 | 66.09 |
| Acinetobacter | 252 | 0.9715 | 0.0649 | 0.8418 | 7 | 39.37 |
| Campylobacter | 248 | 0.7714 | 0.0930 | 0.5853 | 13 | 30.98 |
| Corynebacterium | 224 | 0.8220 | 0.2001 | 0.4204 | 16 | 55.15 |

In order to investigate the genomes with SkewI values below the threshold, I focused on genome assemblies with accompanying read data in the NCBI Sequence Read Archive (SRA) that could be used to validate the assembly. Although there were 434 genomes with SkewI values below the threshold for their particular genus, 325 of these genome assemblies (75%) did not provide the reads used for assembly. 23 genome assemblies provided only short read data while 30 provided long read data. Only 56 of the 434 genomes (13%) listed both long and short reads used for genome assembly. For example, both the *Chlamydia* and *Corynebacterium* genera contained 16 genomes with low SkewI values relative to the expected SkewI for that genus. However, for both of these genera, all 16 genome assemblies did not provide any read data. NCBI also listed no read data for the 11 *Lactobacillus* genomes below threshold and the 10 *Bacillus* genomes below the SkewI threshold. For the genomes and genera where read data was available, I identified several potentially mis-assembled *Escherichia* and *Burkholderia* genomes. Additionally, I were able to identify an interesting phenomenon in *Mycobacterium* genomes relating GC-Skew to GC-content. The following sections describes these findings.

### 2.3.1 Escherichia

For the *Escherichia* genus, RefSeq contains 934 complete genomes, with an average SkewI value of 0.87 and a threshold of 0.75 (**Figure 2.5**). While the majority of

*Escherichia* genomes had SkewI values above the threshold, one of them, *Escherichia coli O121 strain RM8352* (*E. coli O121*), had a SkewI of 0.275, which appeared far too low. In an effort to validate this assembly, I aligned the original raw reads back to the genome while also comparing *E. coli O121* to *Escherichia coli M8*, which has a more-typical SkewI of 0.877. Initial analysis of the GC skew plots for both *E. coli* genomes revealed a clear difference between the genomes, as shown in **Figure 2.5**. For *E. coli M8*, the GC skew plot shows that almost precisely half the genome has more Gs than Cs, and the other half has more Cs than Gs, as is typical for this species. In *E. coli O121*, by contrast, a much larger portion of the forward strand has more Gs than Cs. I then aligned *E. coli O121* against *E. coli M8* (using used NUCmer [31]), revealing a large inversion in *E. coli O121* from position 2,583,081 to 4,963,263. Alignment of assembly reads to each genome using Bowtie2 [32] revealed gaps in coverage at the points flanking both ends of the inversion in *E. coli O121*, suggested that the assembly is incorrect in those regions (**Figure 2.5**).

Because there were no reads supporting the inversion from 2,583,081 to 4,963,263 in *E. coli O121*, I replaced this sequence with its reverse complement and repeated my analysis. The new *E. coli O121* genome has a SkewI of 0.77 with an evenly divided GC skew plot (**Figure 2.5D**). Comparison of the new *E. coli O121* against *E. coli M8* shows a much more consistent 1-to-1 alignment between the two genomes, with only one small inversion remaining.

### 2.3.2 Burkholderia

The *Burkholderia* genomes have a mean SkewI of 0.932 with a SkewI threshold of 0.715 (**Figure 2.6A**). Although there are 619 finished chromosomes from the *Burkholderia*

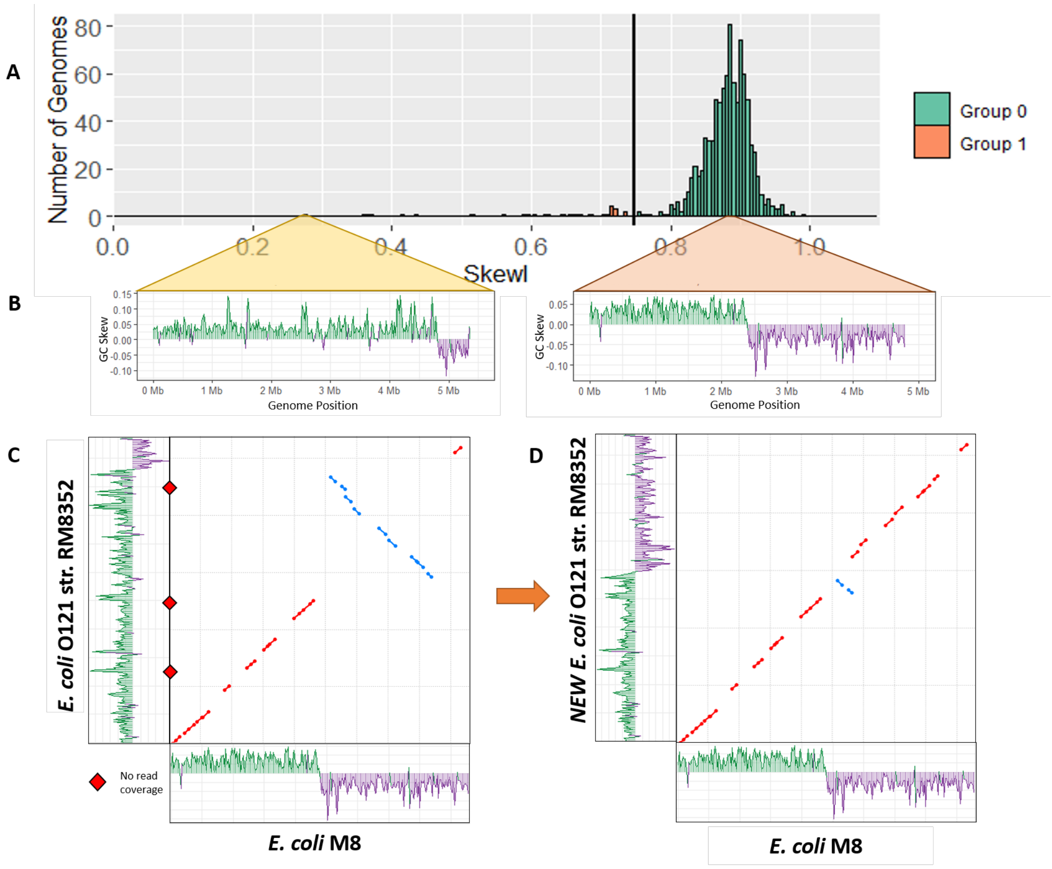


Figure 2.5: **Escherichia skew index values.** **A)** SkewI for all 934 *Escherichia* genomes. The threshold (vertical black line) is at 0.711. **B)** GC skew plots for *Escherichia coli O121 strain RM8352* and *Escherichia coli M8.* *E. coli O121* has an unusually low SkewI of 0.223, while *E. coli M8* has a SkewI of 0.86, which is typical for this genus. **C)** Initial alignment between the two *E. coli* genomes revealed a large inversion. Alignment of the assembly reads revealed locations with no read coverage (red diamonds) *E. coli O121* at both ends of the inversion. **D)** Flipping the inversion in strain RM8352 produced a much more consistent alignment between the *E. coli* genomes (dot plot), and restored the GC skew plot to a more normal appearance (shown along the y axis).

genus, they represent only 270 individual organisms; each *Burkholderia* strain typically has 2-3 chromosomes. **Figure 2.6B** shows the SkewI distribution based on chromosome. There is no significant difference in SkewI between chromosomes.

Further analysis of the individual genomes with SkewI values below the threshold revealed significant differences between the SkewI values for the three chromosomes of *Burkholderia contaminans MS14*. Notably, chromosome 2 had a SkewI of 0.322 while chromosomes 1 and 3 had SkewIs of 0.869 and 0.909 respectively (**Figure 2.6C**). By comparison, the three chromosomes of a different strain, *Burkholderia contaminans SK875*, all had very high SkewIs of 0.978, 1.000, and 1.000.

Aligning the raw *B. contaminans MS14* assembly reads against the three chromosomes using Bowtie2 [32] revealed many locations with no read coverage, suggesting that the full read set used for the assembly was not available in the NCBI SRA. I then aligned the *B. contaminans MS14* chromosomes against the same chromosomes for *B. contaminans SK875* and observed multiple large-scale disagreements between the chromosomes. While chromosome 3 from both strains aligned nearly perfectly, only 50% of chromosome 1 and 2 of MS14 aligned to the same corresponding chromosome of *B. contaminans SK875* (**Figure 2.6D**).

I then aligned chromosome 1 of *B. contaminans MS14* to chromosome 2 of *B. contaminans SK875* and vice versa and discovered that the sequences of *B. contaminans MS14* appeared mis-assembled (**Figure 2.6E**). Based on the differences in alignment

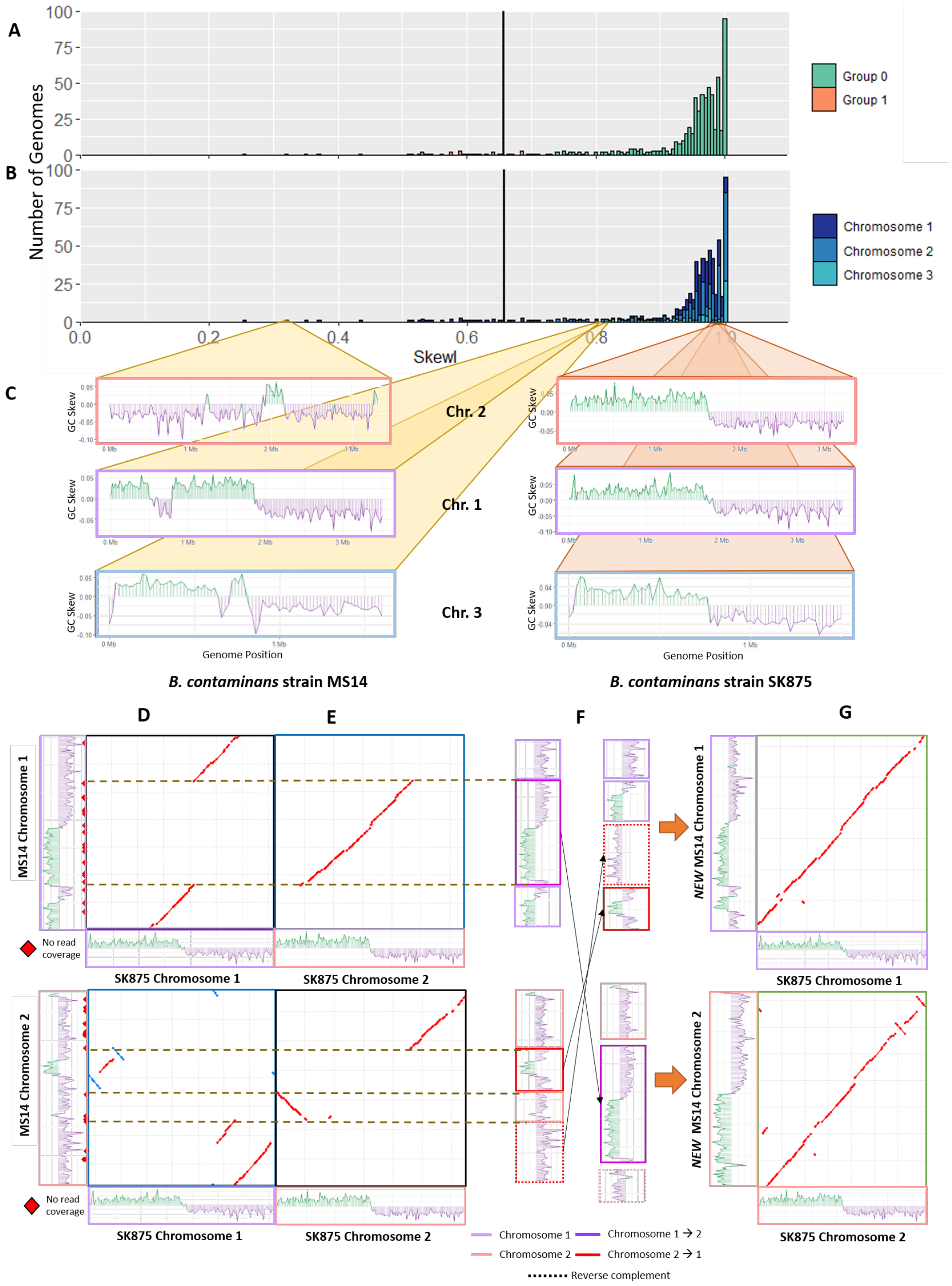


Figure 2.6: **Burkholderia skew index values.** **A)** SkewI for all 934 *Burkholderia* genomes. The threshold is 0.715. **B)** SkewI colored by chromosome. **C)** GC skew plots for all three chromosomes (chr) for *Burkholderia contaminans* strains MS14 (left) and SK875 (right). **D)** Alignments between MS14 and SK875 chr 1 and 2. MS14 is shown on the y axis of each plot. **E)** Cross-chromosome alignments between MS14 and SK875 chr 1 and 2 reveal that a 1.7Mbp region of MS14 chromosome 1 belongs to chr 2 and two regions in MS14 chr 2 belong in chr 1. **F)** We rearranged the sequences of MS14 chr 1 and 2 based on the alignments and GC-Skew plots. **G)** The final MS14 chr alignments with those of *B. contaminans* SK875.

and the GC skew plots of *B. contaminans MS14*, it appears that the 1.7Mbp region of *B. contaminans MS14* chromosome 1 from 812,522 to 2,579,632 belongs to chromosome 2. Similarly, two regions from *B. contaminans MS14* chromosome 2 belong to chromosome 1. (I note that it is possible that a very recent set of translocations and re-arrangements explains the anomalous SkewI value; however, the available data does not support that hypothesis.)

Based on the chromosome alignments and GC skew plots, I rearranged and inverted the individual *B. contaminans MS14* sequences as illustrated in **Figure 2.6F**. The final SkewI for these corrected chromosome 1 and chromosome 2 sequences were 0.774 and 0.946 respectively, both within the expected range. Additionally, realigning the new MS14 sequences against those of SK875 a far higher degree of synteny between the two genomes (**Figure 2.6G**).

### 2.3.3 SkewI versus GC Content and Mycobacterium

Analysis of the *Mycobacterium* SkewI distribution revealed a main peak at 0.85 and a smaller peak centered around 0.4 (**Figure 2.7A**). Due to the large standard deviation, the SkewI threshold was calculated to be 0.413, with 20 genomes falling below the threshold. However, upon investigation into the individual genomes, it appeared that all 20 of these genomes come from *Mycobacterium avium* and *M. avium* subspecies, suggesting that the SkewI values are not reflective of a mis-assembly but rather reflective of a different degree of skew in *M. avium* and possibly other species within the Mycobacteria.

I explored this hypothesis by re-plotting SkewI using different colors for each of the 12 species, as shown in **Figure 2.7B**. As the plot shows, the large peak centered around 0.85 mainly consists of the 179 *M. tuberculosis* genomes while the smaller peak mainly consists of the 27 *M. avium* genomes. Because *Mycobacterium* genomes

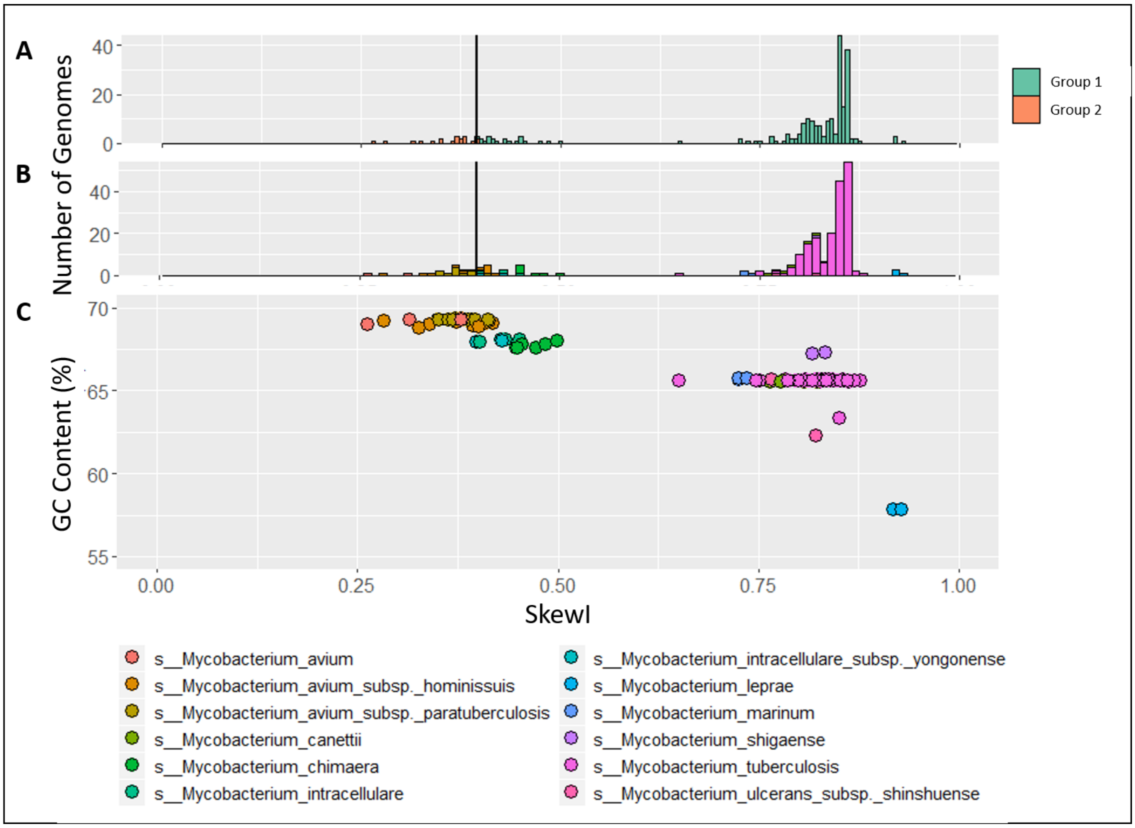


Figure 2.7: **Mycobacterium skew index values. A)** SkewI for 236 *Mycobacterium* genomes from 12 *Mycobacterium* species, all of which have multiple strains available in RefSeq. The threshold (vertical line) is at 0.413. **B)** SkewI colored by species. **C)** Plot comparing GC Content (%) to SkewI, where each dot represents a different genome colored by species.

have a high GC-content (%), we then plotted GC-content vs. SkewI for these same genomes (**Figure 2.7C**), revealing that for the *Mycobacterium* genus, higher GC-content results in a lower SkewI.

Although higher GC-content species within the *Mycobacterium* genus tend towards lower SkewI values, this evolutionary-based relationship [33] is not true across all bacterial clades. Upon analysis of the 12 bacterial genera with the greatest number of complete genomes, higher average GC-content does not necessarily reflect a low mean SkewI value (and vice versa, **Table 2.1**). For example, genomes in the *Mycobacterium*, *Burkholderia*, and *Bordetella* genera all have high GC-content (66%, 67% 68% respectively). However, while the average SkewI for *Mycobacterium* and *Bordetella* are relatively low (0.7589 and 0.5152), the average SkewI for *Burkholderia* genomes is at the higher end of the SkewI spectrum (0.9323). Similarly, the low GC-content genera of *Acinetobacter* and *Campylobacter*, (GC-content values of 39%, 31% respectively) have different mean SkewI values; *Campylobacter* genomes have an average SkewI of 0.77 while *Acinetobacter* genomes have an average SkewI of 0.97.

For a more in-depth analysis, I compared SkewI versus GC-content across all bacterial genomes (**Figure 2.8**). **Figure 2.8A** displays SkewI and GC-content for all 15,000+ RefSeq bacterial complete genomes while **Figure 2.8B** plots the mean SkewI and mean GC-content for every bacterial genus. However, analysis of both figures revealed no relationship between SkewI values and GC-content.

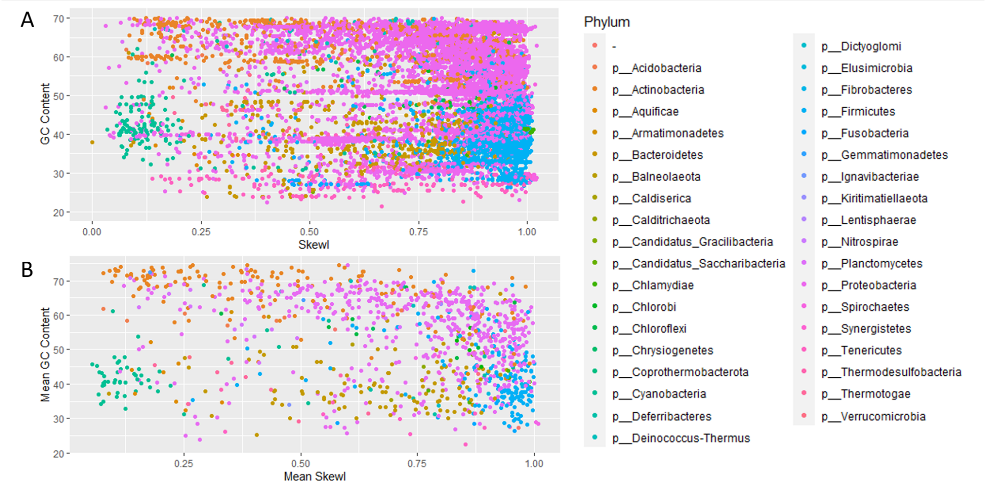


Figure 2.8: **SkewI vs. GC Content for bacterial RefSeq genomes.** This figure compares SkewI to GC-content of each bacterial genome. **A)** displays each individual genome as a separate point, while **B)** displays the average SkewI vs. average GC-content for each bacterial genus. Points in both plots are colored by phylum.

I then generated the same SkewI vs. GC-content figures for genomes in specific genera. **Figure 2.9** shows the SkewI and GC-content distributions for genomes in the *Bacillus*, *Escherichia*, *Salmonella*, and *Burkhoderia* genera. While there is evidence that GC-content is conserved within species, there is no relationship between SkewI and GC-content for these genera. By comparison, **Figure 2.10** shows similar SkewI/GC-content plots for *Mycobacterium* and *Bordetella*. For these two genera, there is some evidence that certain low GC-content species have higher SkewI values. However, while the patterns are more pronounced for *Mycobacterium*, there are some *Bordetella* species that follow this pattern (e.g. *Bordetella pertussis* and *Bordetella parapertussis*), there are also some *Bordetella* species that do not (e.g. *Bordetella flabilis*)

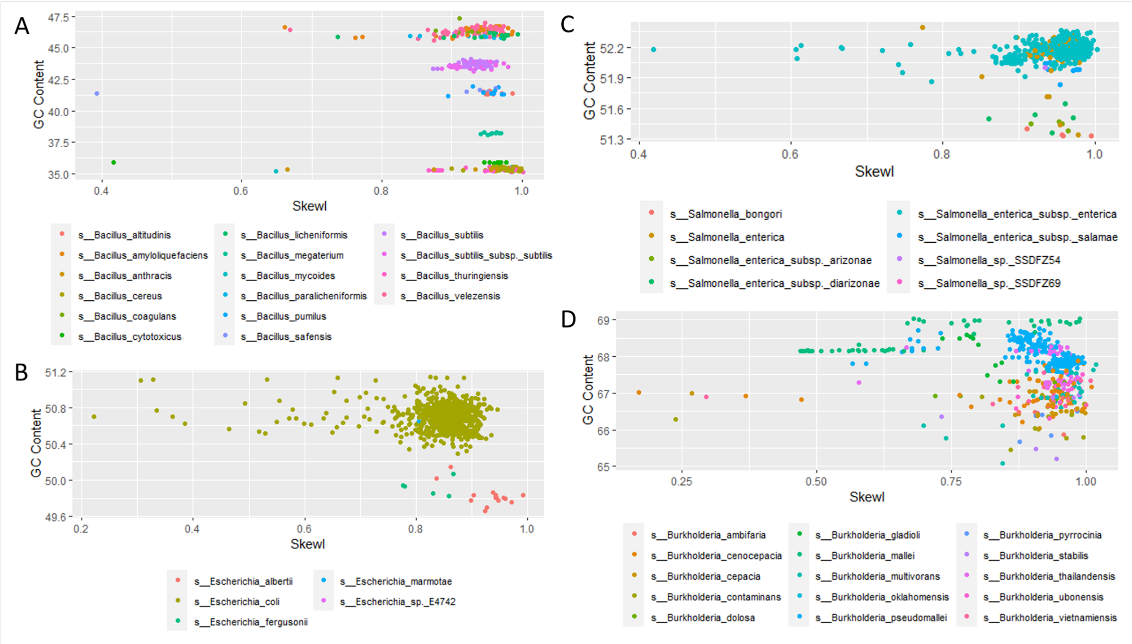


Figure 2.9: **SkewI vs. GC Content for *Bacillus*, *Escherichia*, *Salmonella*, and *Burkholderia* genera.** This figure compares SkewI to GC-content for four bacterial genera where no relationship between SkewI and GC-content is present. Axes in each plot are specific to the range of SkewI and GC-content values for genomes within that genus. Points are colored by species.

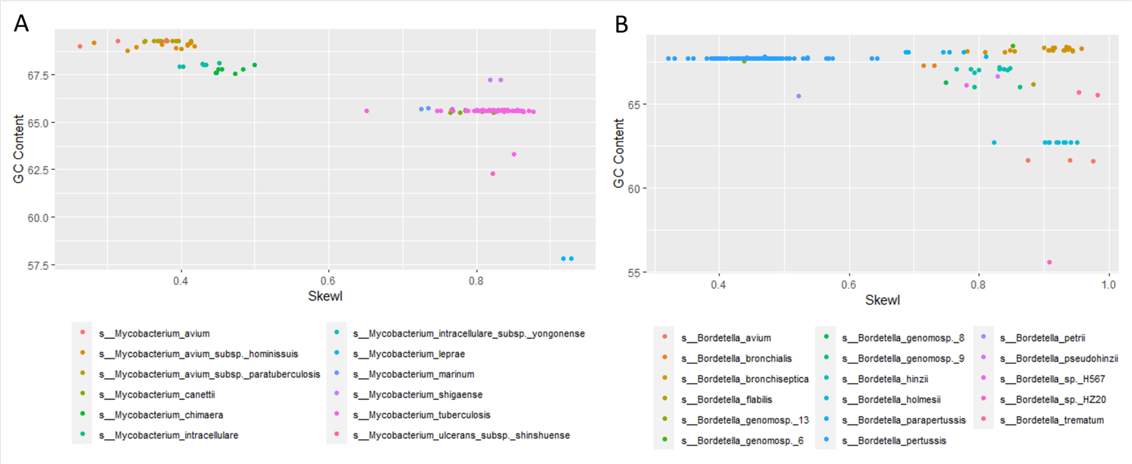


Figure 2.10: **SkewI vs. GC Content for *Mycobacterium* and *Bordetella*** This figure compares SkewI to GC-content for two bacterial genera where higher GC-content genomes tend towards lower SkewI values. Axes in each plot are specific to the range of SkewI and GC-content values for genomes within that genus. Points are colored by species.

### 2.3.4 Simulated Mutations

Following analysis of existing genomes and their SkewI values, I performed the following simulation experiment to measure the sensitivity of the SkewIT method for detecting misassemblies. First, I randomly selected 10 genomes belonging to each of the following species: *Bacillus thuringiensis* (SkewI threshold 0.896), *Salmonella enterica* (SkewI threshold 0.890), *Staphylococcus aureus* (SkewI threshold 0.853), *Escherichia coli* (SkewI threshold 0.759), and *Pseudomonas aeruginosa* (SkewI threshold 0.617). All selected genomes had SkewI values above the SkewI threshold for that genus.

For each genome, I simulated a misassembly error where a random subsequence, of length $k\%$ of the full genome length, is moved to another random location in the genome. I tested 12 different values of $k = 2.5, 5, 7.5, 10, 12.5, 15, 17.5, 20, 22.5, 25, 27.5, 30$ and for each value of $k$, I generated 100 randomly misassembled genomes and subsequently calculated the SkewI value of the misassembled genome. I then calculated the average number (across all 10 genomes for a given species) of misassembled genomes whose new SkewI values fell below the SkewI threshold for that genus.

**Figure 2.11** summarizes the results of this translocation experiment. **Figure 2.11A** shows the different SkewI thresholds for each of the tested species. **Figure 2.11B** displays the average number of misassemblies detected (with SkewI values falling below the threshold) for each value of $k$. As the length of the moved sequence increases,

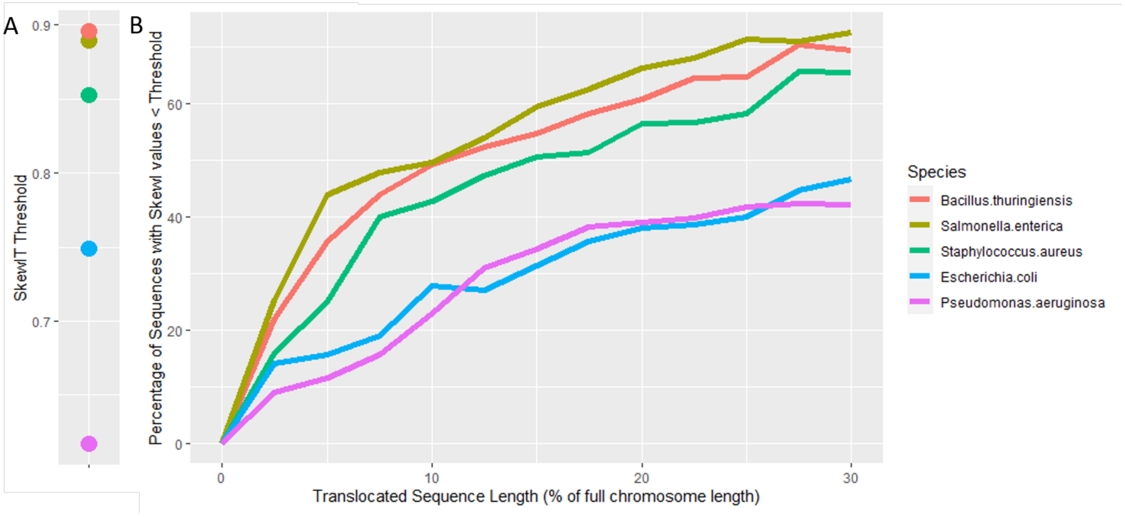


Figure 2.11: **SkewIT Sensitivity to Misassemblies**. In order to evaluate the sensitivity of the SkewIT method for detecting misassemblies, I first randomly selected 10 genomes from these species: *Bacillus thuringiensis*, *Salmonella enterica*, *Staphylococcus aureus*, *Escherichia coli*, and *Pseudomonas aeruginosa*. A) displays the SkewI threshold for each species. For each genome, I simulated 100 misassembled genomes by moving a random subsequence of length $k$% of the full genome length to another random location. This was repeated for 12 values of $k$ ranging from 0 to 30, with 100 random misassemblies for each value of $k$. B) shows the average percentage of the misassembled genomes that had SkewI values below the threshold.

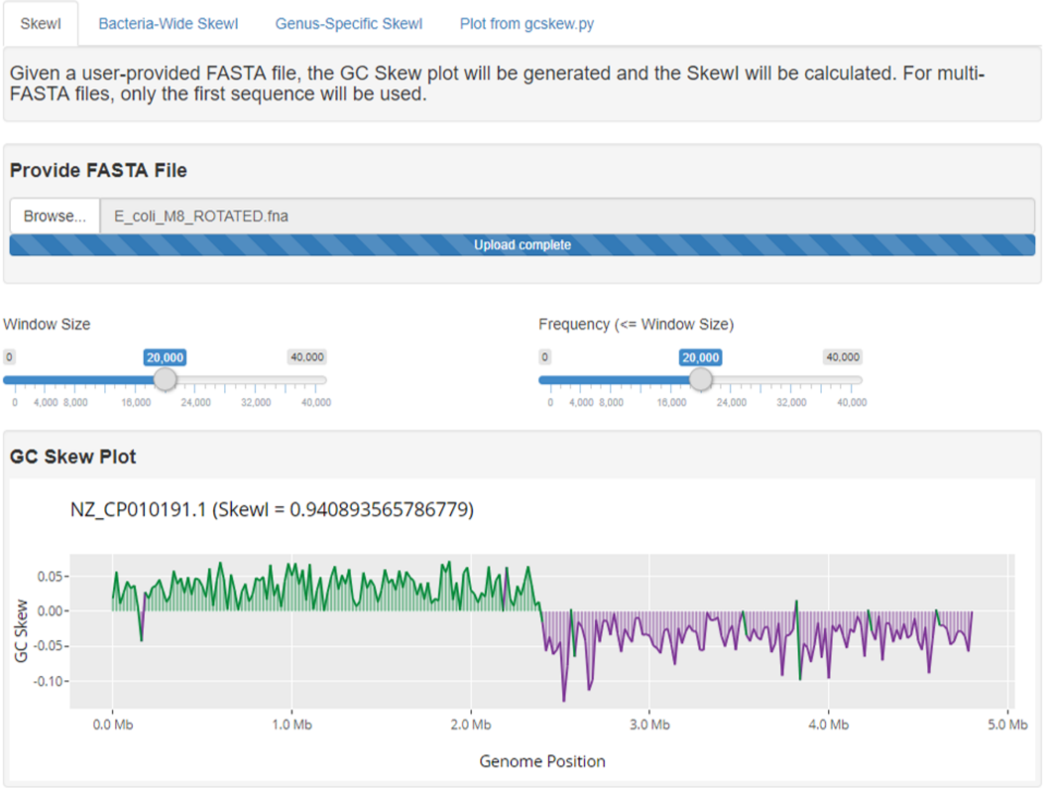
the number of misassemblies detected increases. Moving a short subsequence of only 5% of the full genome length yields a very small change in GC skew. Approximately 20% of these misassemblies caused low enough SkewI values for the SkewIT method to detect the change. However, when long subsequences are displaced, the GC skew pattern of the genome as a whole is disrupted more, decreasing the SkewI value. For example, the SkewIT method detected 60% of misassemblies when 20% of a *Bacillus thuringiensis* genome is randomly moved to the incorrect locations. However, if only 5% of the same genome is moved, then the SkewIT method only detects the misassembly 35.5% of the time. Comparisons between the various species also shows that species with higher thresholds, such as *Bacillus thuringiensis* and *Salmonella enterica*, are more sensitive to genome modifications/misassemblies.

### 2.3.5 SkewIT Runtime and Computational Resources

Execution of the SkewIT code for all 15,000+ NCBI RefSeq bacterial genomes required  30 minutes, using 112Mb of RAM. For a single genome, the SkewIT code calculated SkewI within 1 second, using only 50Mb of RAM. All code is single-threaded and can process multi-FASTA files.

## 2.4 Software Availability

For this project, I developed the SkewIT Application (available at `https://jenniferlu717.shinyapps.io/SkewIT/`) as an interactive web app which calculates SkewI and plots GC skew from a user-provided bacterial genome FASTA file. **Figure 2.12** displays the interface after a user has uploaded a FASTA file for sequence NZ_CP010191.1. The app displays the GC skew plot and the SkewI value based on the user-selected window size and frequency. Frequency is the distance between the start of each window for which GC skew is calculated. Users can regenerate the GC skew plot and recalculate a new SkewI value by choosing new window size/frequency parameters. The default window size and frequency is 20kb. The app also allows users to scroll



Given a user-provided FASTA file, the GC Skew plot will be generated and the SkewI will be calculated. For multi-FASTA files, only the first sequence will be used.

Provide FASTA File

E_coli_M8_ROTATED.fna

Window Size: 20,000

Frequency (<= Window Size): 20,000

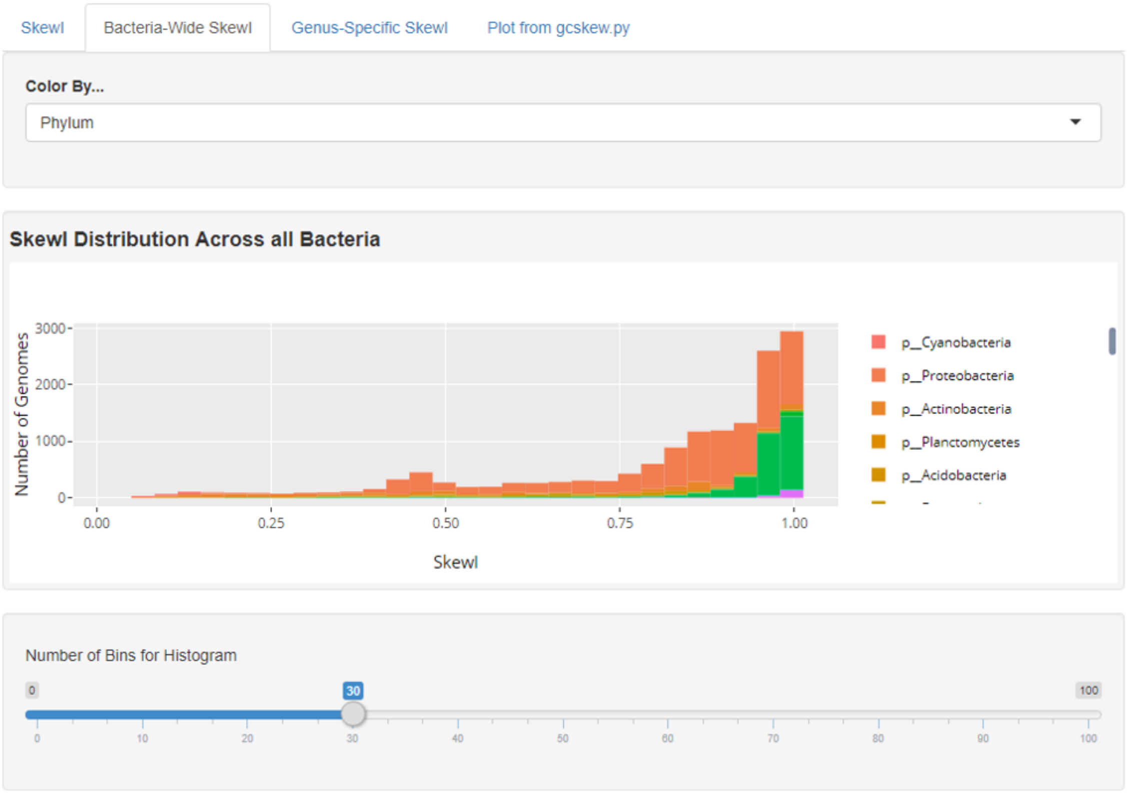
GC Skew Plot

NZ_CP010191.1 (SkewI = 0.940893565786779)

Figure 2.12: **SkewIT App: SkewI Calculation and GC skew Plot.** The main panel in the application allows users to upload any FASTA file from which the program will generate a GC skew plot and calculate the SkewI value for the FASTA sequence.

over the GC skew plot to identify individual genome positions of interest.

The SkewIT app provides additional tabs for investigating the existing SkewI data generated for Refseq Release 97 Bacterial complete chromosomes. First, the ”Bacteria-Wide SkewI” tab, seen in **Figure 2.13**, shows the full range of SkewI values as colored by Phylum or other taxonomic levels of interest. The bins used in generating the SkewI histogram can be adjusted to visualize the number of genomes in each SkewI range.



Color By...

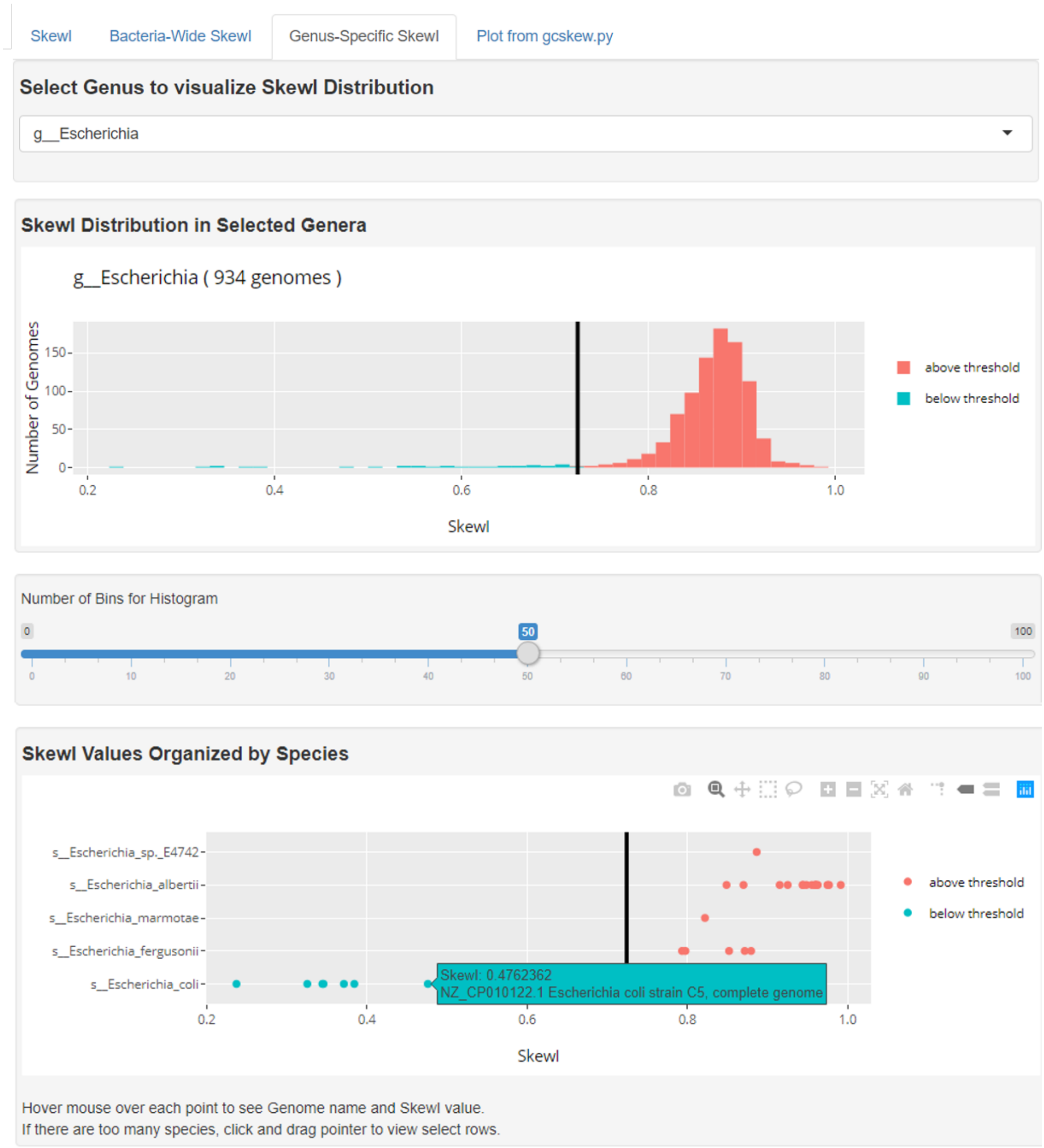
Phylum

Number of Bins for Histogram

30

Figure 2.13: **SkewIT App: Refseq Release 97 Bacterial SkewI Distribution** The SkewIT App allows users to explore the SkewI values across all bacteria in this tab, coloring the plot based on Phylum, Class, or other taxonomic groupings.

Finally, the"Genus-Specific SkewI" tab shows SkewI values for individual genera (**Figure 2.14**). Following selection of a genus, the SkewI values will be displayed in two separate plots: as a histogram and as a dot plot. The histogram shows the distribution of SkewI values for complete chromosomes within that genus. The dot plot displays each genome as a single point, grouped by species within that genus. Scrolling over individual points will show the SkewI value for that genome and the sequence ID associated with that SkewI value. This tab also shows the SkewI threshold for the genus as a black vertical line on both plots.



Select Genus to visualize SkewI Distribution

g__Escherichia

SkewI Distribution in Selected Genera

g__Escherichia ( 934 genomes )

Number of Bins for Histogram

SkewI Values Organized by Species

SkewI: 0.4762362
NZ_CP010122.1 Escherichia coli strain C5, complete genome

Hover mouse over each point to see Genome name and SkewI value.
If there are too many species, click and drag pointer to view select rows.

Figure 2.14: **SkewIT App: Refseq Release 97 Bacterial SkewI Distribution** The SkewIT App allows users to explore the SkewI values across all bacteria in this tab, coloring the plot based on Phylum, Class, or other taxonomic groupings.

## 2.5 Conclusion

The SkewIT (Skew Index Test) provides a fast method for identifying potentially mis-assembled genomes based on the well-known GC skew phenomenon for bacterial genomes. In this study, I described and implemented an algorithm that computes a new GC skew statistic, SkewI, and I computed this statistic across 15,067 genomes from RefSeq, discovering that GC skew varies considerably across genera.

Following the SkewIT method, I used anomalous values of SkewI to identify likely mis-assemblies in *Escherichia coli O121 strain RM8352* and in two chromosomes of *Burkholderia contaminans MS14*. For further analysis, I also used the SkewI values to investigate relationships between GC skew and GC-Content, discovering that certain genera do show a correlation between these two metrics, such as for *Mycobacterium* genomes. Finally, to determine the sensitivity of SkewIT for detecting misasssemblies, I performed an experiment with simulated mis-assemblies. In this experiment, I showed that when a longer portion of a genome is incorrectly placed, SkewIT is better able to detect the change in GC skew.

I suggest that researchers can validate future bacterial genome assemblies by running SkewIT and comparing the resulting SkewI value to the thresholds in **Table 2.1**. Genomes with SkewI values lower than the expected threshold should be further validated by comparison to closely-related genomes and by alignment of the original reads to the genome.

# Chapter 3

# Eukaryotic Pathogen Genomes

**Portions of this chapter originally appeared in:**

J. Lu, S. L. Salzberg (2018). “Removing contaminants from databases of draft genomes.” *PLoS Comput Biol*, 14(6): e1006277.

`https://doi.org/10.1371/journal.pcbi.1006277`

**Related data:** `http://ccb.jhu.edu/data/eupathDB/`

## 3.1 Introduction

### 3.1.1 Next-generation sequencing in pathogen discovery/diagnosis

Next-generation sequencing (NGS) over the last few years has emerged as a valuable tool for human pathogen discovery and diagnosis. In the case of human pathogen detection, traditional histological, immunological, or molecular techniques are limited and often yield an incorrect or incomplete diagnosis [34]. As sequencing has grown faster and cheaper, clinicians have begun to explore the possibility of replacing older methods with NGS, which provides a fast, specific, and relatively unbiased method of capturing the full spectrum of macro- and microorganisms in any sample.

A growing number of case studies illustrate the potential for NGS in diagnosis. For example, in 2013 Loman et al. conducted a retrospective investigation into the 2011 German outbreak of Shiga-toxigenic Escherichia coli (STEC) [35]. In this study, sequencing led to rapid and accurate identification of the bacterial infection in fecal specimens of the infected patients. In 2014, Hasman et al. analyzed 35 urine samples from patients with suspected urinary tract infections, confirming cultured bacterial infections using sequencing of isolated and cultured bacteria [36]. They also successfully identified polymicrobial bacterial infections by directly sequencing the urine samples. Later in 2014, Wilson et al. used next-generation sequencing of cerebrospinal fluid (CSF) to identify and treat a bacterial Leptospira infection in a 14-year old patient [37]. In 2016, Salzberg et al. tested the possibilities of detecting pathogens by sequencing brain or spinal cord biopsies from 10 patients presenting with neurologic symptoms with previously unidentified infections [38]. In that study, NGS identified both bacterial and viral infections in selected patients, diagnoses that were confirmed by traditional immunologic techniques.

A critical step in using NGS for diagnosis is in the bioinformatics analysis of the millions (or billions) of genomic reads that result from a sequencing experiment. The identification of the sequenced DNA provides the information about the potential pathogenic organisms causing the infection. Because the source of the sample is human tissue, all the studies mentioned above first filtered out human DNA, which is uninformative for pathogen discovery [35–38]. Following this step, the remaining sequencing reads are compared to reference genomic databases, such as RefSeq [5] or the NCBI nt database [4], using a variety of alignment and classification tools, including BLAST, Bowtie2, and Kraken [12,32,39].

### 3.1.2 Challenges in relying on reference databases

#### 3.1.2.1 Database Composition

Although databases of sequenced pathogens have grown dramatically larger over the past decade, the dependence on reference databases still presents challenges when used for diagnosis, for at least two reasons: (1) no database contains the full spectrum of all potential human pathogens, and (2) existing reference databases have been found to contain contamination.

Over the past two decades, microbial genome projects have predominantly focused on bacteria and viruses. Two of the most widely used genomic databases are the NCBI GenBank and NCBI RefSeq. The NCBI GenBank repository [4, 40] contains the majority of genome sequence data submitted by laboratories around the world while the NCBI RefSeq repository [5] is a curated subset of the GenBank genome sequences. As of January 2018, GenBank contained genome entries representing over 54,000 bacterial organisms but only 1,791 fungi and 389 protozoa (**Table 3.1**). RefSeq also reflected the focus on bacterial and viral genomes, with $\sim 37,000$ bacterial organisms and more than 7,500 viral organisms represented. By contrast, RefSeq contained genomes for only 251 fungi and 82 protozoa.

Between 2018 and 2020, the total number of organisms in RefSeq and Genbank increased by more than 45%, with significantly more bacterial and viral genomes. However, fungi and protozoa continue to represent only ∼ 3% of the Genbank genomes and ∼ 0.5% of Refseq genomes available. As of June 2020, GenBank contained genome entries representing over 72,000 bacterial organisms but only 3,018 fungi and 514 protozoa. The NCBI RefSeq project grew to contain 51,000 bacterial genomes and more than 9,700 viral genomes. By contrast, RefSeq contains genomes for only 329 fungi and 94 protozoa (**3.1**).

The composition of the reference databases is not representative of the species composition of the natural world, but rather reflects a focus on human pathogens, other

Table 3.1: **Organisms in GenBank and RefSeq: January 2018 vs. June 2020.** The number of vertebrates listed is the total number of organisms with either chromosome-level and complete assemblies. All other counts represent the number of complete genomes alone.

| | Draft and Complete Genomes | | | | Complete Genomes | | | |
|---|---|---|---|---|---|---|---|---|
| | GenBank | | RefSeq | | GenBank | | RefSeq | |
| | 2018 | 2020 | 2018 | 2020 | 2018 | 2020 | 2018 | 2020 |
| Bacteria | 54,153 | 72,182 | 37,399 | 51,439 | 5,372 | 8,152 | 5,121 | 7,851 |
| Viruses | 10,412 | 21,677 | 7,509 | 9,787 | 10,339 | 20,280 | 7,484 | 9,404 |
| Archaea | 1,861 | 2,514 | 533 | 818 | 272 | 346 | 235 | 320 |
| Vertebrates | 376 | 1,076 | 238 | 346 | 71* | 235 | 55* | 136 |
| Plants | 320 | 713 | 102 | 127 | 3 | 4 | 3 | 3 |
| Fungi | 1,791 | 3,018 | 251 | 329 | 26 | 53 | 8 | 11 |
| Protozoa | 389 | 514 | 82 | 94 | 3 | 14 | 2 | 3 |
| Total | 69,302 | 101,694 | 46,144 | 62,940 | 16,086 | 29,084 | 12,908 | 17,728 |
| Bacteria | 78.1% | 71.0% | 81.1% | 81.7% | 33.4% | 28.0% | 39.7% | 44.3% |
| Viruses | 15.0% | 21.3% | 16.3% | 15.6% | 64.3% | 69.6% | 58.0% | 53.0% |
| Archaea | 2.7% | 2.5% | 1.2% | 1.3% | 1.7% | 0.1% | 1.8% | 1.8% |
| Vertebrates | 0.5% | 1.0% | 0.5% | 0.6% | 0.4% | 0.1% | 0.4% | 0.8% |
| Plants | 0.5% | 0.7% | 0.2% | 0.2% | 0.0% | 0.0% | 0.0% | 0.0% |
| Fungi | 2.6% | 3.0% | 0.5% | 0.5% | 0.2% | 0.2% | 0.1% | 0.1% |
| Protozoa | 0.6% | 0.5% | 0.2% | 0.1% | 0.0% | 0.0% | 0.0% | 0.0% |

species of interest to humans, and the challenges of isolating and sequencing DNA from various species [41]. In many cases, microorganisms are difficult to isolate from their surrounding environments, living among thousands of other species in complex ecosystems [42, 43]. Some microorganisms live in extreme conditions and have gone undiscovered until recently [44]. Other microorganisms are difficult to grow in culture to provide sufficient DNA from which to derive a reference genome. As a result of these constraints, most early research into microorganisms focused on a few easily culturable bacteria [45]. However, studies over the last two decades suggest that culturable bacteria represent only a small fraction of the microorganisms on earth [41, 45–47].

Eukaryotic pathogens comprise an underrepresented group of microorganisms in genomic databases, although they are critically important for the diagnosis of human infections. This group includes a diverse group of species that infect multiple areas in the body; e.g., apicomplexans such as *Plasmodium falciparum*, which causes most cases of human malaria [48], and *Toxoplasma gondii* [49], which may cause neurological defects. Other examples include multiple fungal species belonging to the *Fusarium*, *Aspergillus*, *Curvularia*, and *Candida* genera, and amoebae species belonging to the *Acanthamoeba* genus, the latter of which causes a majority of human corneal infections [50, 51]. These are only a small sample of the hundreds of known eukaryotic pathogens of humans.

Table 3.2: **EuPathDB genome representation in RefSeq.** This table shows the number of genomes from the eukaryotic pathogen database that also exist in the NCBI RefSeq database along with the breakdown of their assembly status.

| | RefSeq 2018 | RefSeq 2020 |
|---|---|---|
| Complete Genome | 3 | 6 |
| Chromosome | 41 | 48 |
| Scaffold | 46 | 77 |
| Contig | 5 | 13 |
| *Not Represented* | *150* | *244* |
| Total | 245 | 388 |

EuPathDB is a database that, as of January 2018, represents 245 eukaryotic microorganisms [52], including both known pathogens and other closely related non-infectious eukaryotic species. Because no eukaryotic pathogen has yet been completely sequenced, this resource comprises primarily draft genomes at varying degrees of completeness, some of which have had little curation since their initial sequencing. However, EuPathDB is more comprehensive than the RefSeq database, containing more than 150 genomes that are absent from the RefSeq protozoa and fungi databases (see Table 3.2). By June 2020, EuPathDB-46 contained 388 eukaryotic genomes, 244 of which are absent in the RefSeq databases.

#### 3.1.2.2 Database Contamination

In recent years, multiple studies revealed contamination in the public genome sequences of many organisms, particularly for draft genomes. In 2011, Longo et al. identified 492 non-primate public databases from NCBI, UCSC, and Ensembl containing human genome sequences [8]. A 2014 study found that portions of the complete genome for *Neisseria gonorrhoeae* TCDC-NG08107 belonged to the cow and sheep genomes [53]. Another study in 2015 identified over 18,000 microbial isolate genome sequences that were contaminated with PhiX174, a bacteriophage used as a control in Illumina sequencing runs [9]. 10% of those 18,000 genomes were published in the literature. In 2016, Kryukov et al. identified 154 non-human genome assemblies containing human sequence fragments that were at least 100bp long [10]. As one example, they discovered that more than 330,000 bp in the reference genome of *Plasmodium gaboni*, a relative of *Plasmodium falciparum*, appears to be contaminating human sequence.

Contamination and incompleteness in reference databases causes bioinformatics analysis of sequencing reads to yield both false positive and false negative results, thereby decreasing the overall reliability of NGS in pathogen diagnostics. False positives, where the wrong pathogen is identified, might in turn lead to inaccurate treatments, with the potential to harm rather than help patients.

## 3.2 Methods for the Removal of Genomic Contamination

In this study, I present a new method for eliminating genomic contamination that can be used on both complete and draft reference genomes. I first test the method on

EuPathDB-28 (released in 2018), yielding a cleaned and filtered eukaryotic pathogen database ready for use in bioinformatics pipelines, including those intended for NGS diagnostics, with decreased false positive and false negative rates. I then repeat the method for EuPathDB-46 (released at the end of 2019) to generate an updated and improved eukaryotic pathogen database.

The eukaryotic pathogen genomes underwent a multi-step cleaning process to remove both contaminating and non-informative sequences (see **Figure 3.1**). Each genome was first split into 100bp overlapping pseudo-reads, with each pseudo-read beginning every 50bp along the genome. The pseudo-reads were then compared to three unique databases, using the Kraken [12] and Bowtie2 [32] classification and alignment programs.

Kraken labels reads only if they contain an exact 31 base-pair (31-mer) match to any 31-mer in the database sequences [12]. For this process, pseudo-reads were classified with Kraken against two unique Kraken databases. The first Kraken database contains 15,000 genomic sequences from the human, human CHM1, mouse, bacteria, archaea, viral, and plant RefSeq databases as of November 30th, 2017. I also included contaminating sequences such as the UniVec database, EmVec database, and phiX174 vector in the first Kraken database. The second Kraken database contains all complete and chromosomal-level assemblies of non-human and non-mouse vertebrate sequences (representing 56 vertebrate species). Kraken requires that the

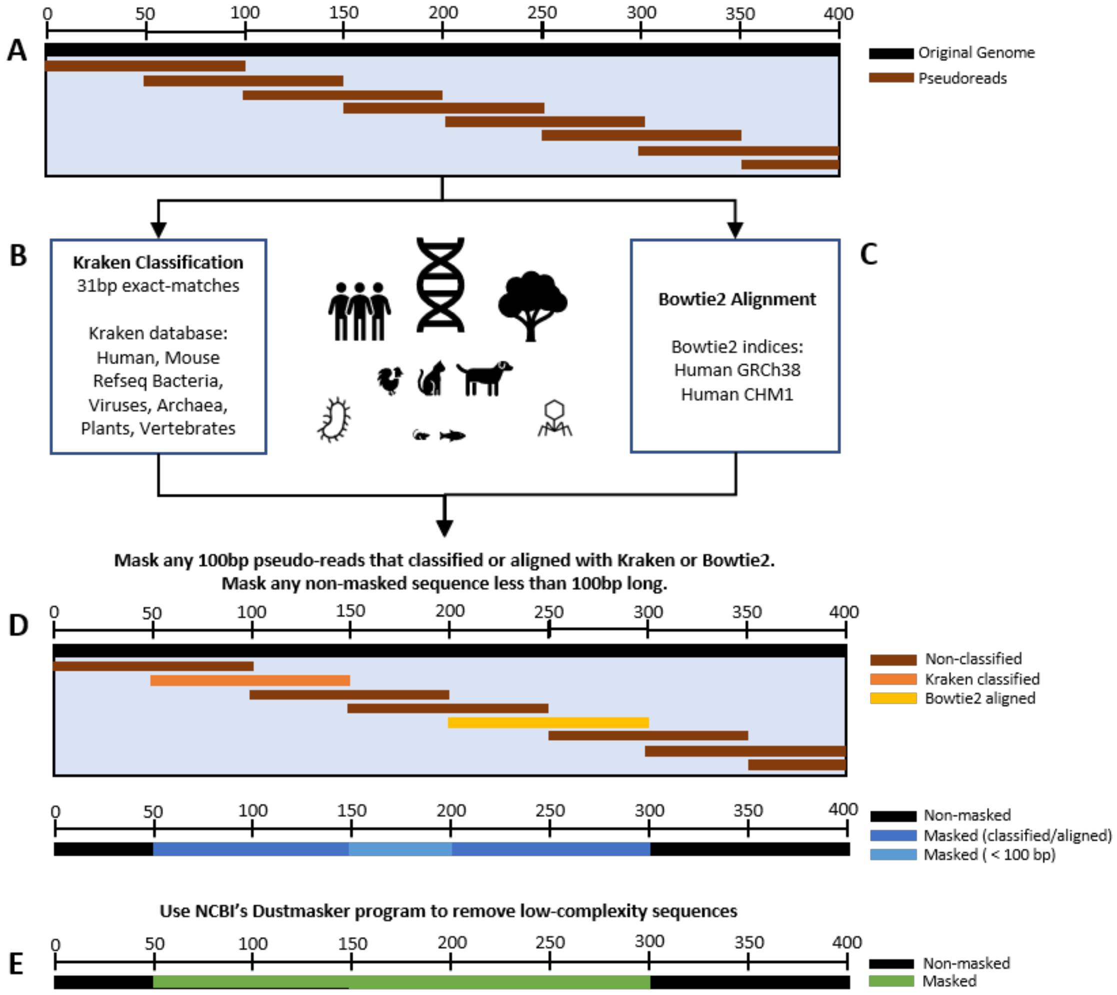


Figure 3.1: **Masking procedure A)** The original genome is split into 100bp overlapping pseudo-reads. **B)** The pseudo-reads are then classified using Kraken first against the common contaminating vector sequences and the plant, viral, bacterial, archaeal, human, and mouse RefSeq database. The pseudo-reads are also classified using Kraken against non-human and non-mouse vertebrate RefSeq genomes. **C)** Bowtie2 is then used to align all pseudo-reads against the human genome. **D)** All pseudo-reads that were classified in the previous steps are masked out of the original genomes. Any remaining non-masked sequence with less than 100p is also masked. **E)** Finally, Dustmasker is used to mask additional low-complexity sequences.

selected database is first loaded into RAM prior to classification. I used two databases in order to reduce RAM usage at a single time, allowing sequential classification of the pseudo-reads to each database.

Bowtie2 aligns sequencing reads against any reference sequence, allowing for gaps or mismatches [32]. I created a bowtie2 index of GRCh38.p11 (the human reference genome) and Human CHM1 (another haploid human genome) and aligned the pseudo-reads against it. Note that even though I include GRCh38.p11 in the Kraken database, which enables Kraken to find human reads, Bowtie2's more sensitive alignment algorithm can align some sequences that Kraken will miss.

Any pseudo-read that was classified in these steps represents either a contaminating sequence in the pathogen genome or a low-complexity sequence that matches a distant species only by chance. In either case, these sequences could lead to false positive identifications if they are used for metagenomics analysis. Therefore, I masked any portion of a database genome that corresponded to a pseudo-read that was classified or aligned in the previous steps. (Masking can be done in a variety of ways; I simply replaced the sequence with Ns to keep the overall genome length the same.) If, after this initial masking step, I created non-masked sequences that were ¡100 bp in length, I masked those sequences as well. I then used Dustmasker [54] to mask additional low-complexity sequences (**Figure 3.1**).

## 3.3 Contamination Removal Results

I tested my method for eliminating contamination on the draft genomes contained in EuPathDB release 28 (released in 2018) [52], which contains 245 genomes categorized into the following sub-databases: AmoebaDB (29 genomes), CryptoDB (11), FungiDB (87), GiardiaDB (6), MicrosporidiaDB (25), PiroplasmaDB (8), PlasmoDB (9), ToxoDB (30), TrichDB (1), and TriTrypDB (39).

**Figure 3.2** and **Table 3.3** show how much of each of the 245 genomes was masked in each step of the cleaning procedure and the final lengths of the cleaned pathogen genomes.

Genome lengths in EuPathDB ranged from 2Mbp to 186Mbp prior to our cleaning procedure. Post-cleaning genome lengths ranged from 1.7Mbp to 182Mbp, with an average of 11% of each genome identified as contaminating or low-complexity sequences. As **Figure 3.2** illustrates, a few genomes were outliers with over 50% of the genome being masked, but most genomes lost ¡10% of their length through this process.

In the first masking step, pseudo-reads across all EuPathDB genomes are classified against two Kraken databases containing bacterial, archaeal, viral, human, mouse, vertebrate, and contaminating vector genomes (**Figure 3.1**). Reads classified as vertebrates are further broken down into sub-classifications such as fish or bird species.

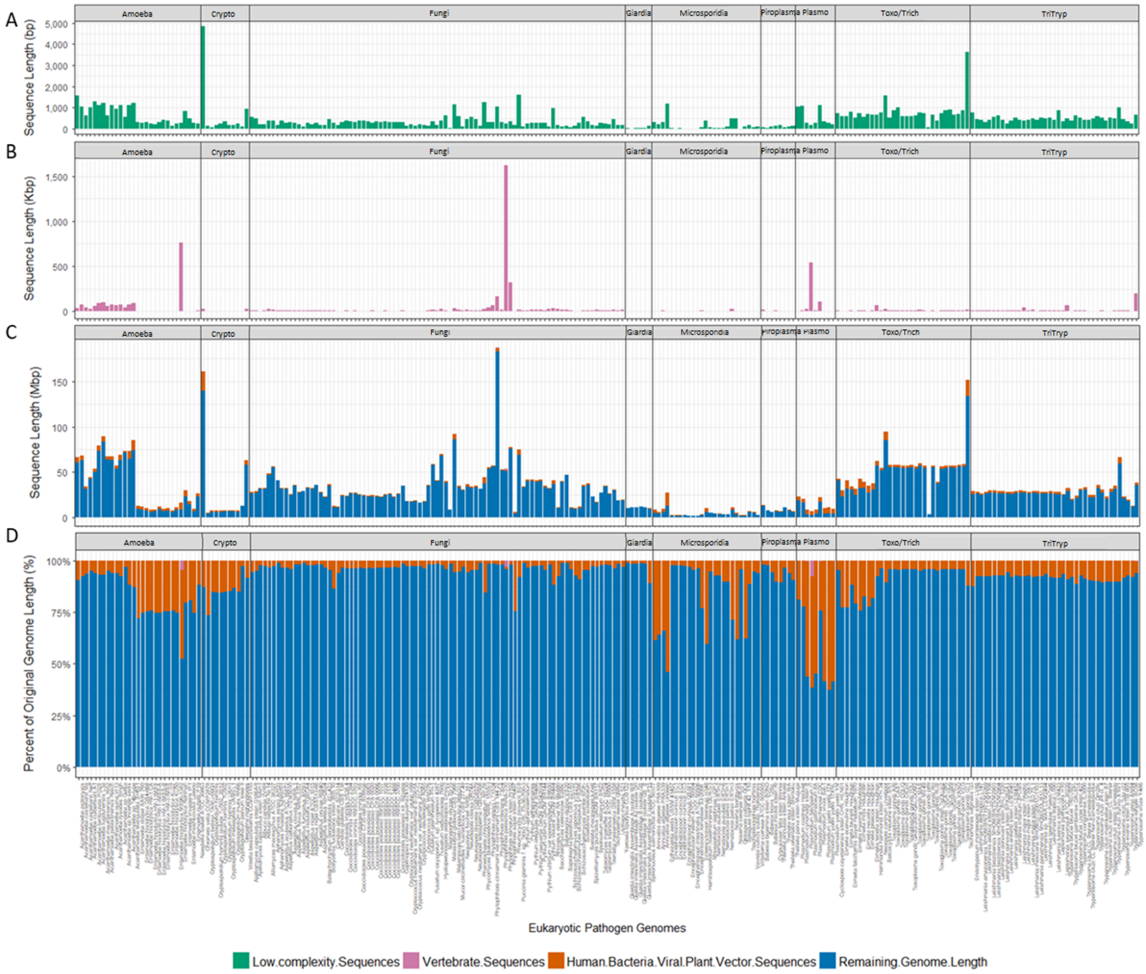


Figure 3.2: **Masking results.** **C)** provides an overview of sequence lengths for each eukaryotic pathogen genome masked in each step and the sequence lengths of the final cleaned genomes. As low-complexity sequences and vertebrate masked sequences are much smaller compared to the final genome length or human/bacterial/viral/plant/vector sequences, these were additionally plotted in **A)** and **B)** for each eukaryotic pathogen genome. Low-complexity sequences were masked as a final step as well. Masked sequence lengths are also presented as percentages of the original genome length to show the percent of each genome remaining and the percent masked in each step **D)**.

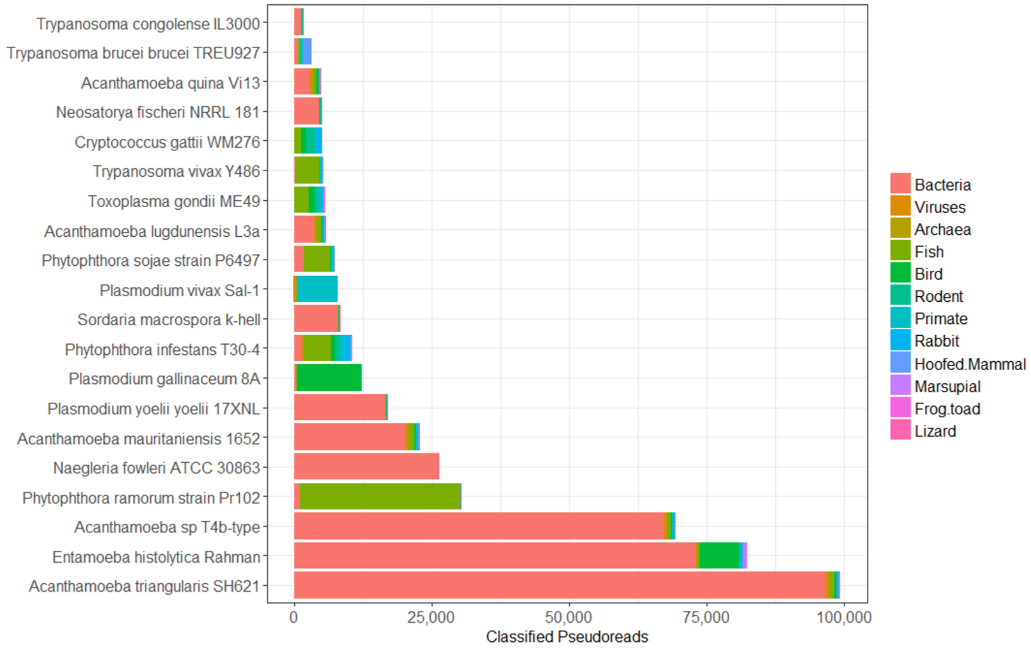


Figure 3.3: **Pseudo-read Kraken classifications.** This plot shows the 20 eukaryotic pathogen genomes with the greatest numbers of pseudo-reads that Kraken identified as matching foreign species when searching against database containing bacteria, viruses, archaea, and a limited set of vertebrate genomes. Vertebrate classifications are grouped by common categories, such as fish, birds, rodents, or primates. Primate and rodent numbers do not include human and mouse, which are counted and shown separately.

**Figure 3.3** shows the breakdown of these classifications for the 20 pathogen genomes with the largest numbers of classified pseudo-reads. **Figure 3.4** shows a similar breakdown focusing specifically on the 20 genomes with the most pseudo-reads labelled as mouse or human.

Most genome masking occurred after the first Kraken screen against the database of bacterial, archaeal, viral, human, mouse, and vector genomes. As a result of this step, I masked on average $\sim 10\%$ of each of the EuPathDB genomes. After classifying the

Table 3.3: **Genomes in EuPathDB-28**

| Type | Genome | Orig Length | Final Length | |
|---|---|---:|---:|---:|
| AmoebaDB | *Acanthamoeba astronyxis* Unknown | 74,102,518 | 60,248,454 | 81.3% |
| AmoebaDB | *Acanthamoeba castellanii* Ma | 77,971,411 | 63,240,657 | 81.1% |
| AmoebaDB | *Acanthamoeba castellanii* str. Neff | 39,443,455 | 31,934,200 | 81.0% |
| AmoebaDB | *Acanthamoeba culbertsoni* A1 | 48,610,170 | 42,492,065 | 87.4% |
| AmoebaDB | *Acanthamoeba lenticulata* PD2S | 59,106,818 | 50,456,191 | 85.4% |
| AmoebaDB | *Acanthamoeba lugdunensis* L3a | 90,244,278 | 73,682,923 | 81.6% |
| AmoebaDB | *Acanthamoeba mauritaniensis* 1652 | 99,275,674 | 83,231,758 | 83.8% |
| AmoebaDB | *Acanthamoeba palestinensis* Reich | 73,493,067 | 63,560,786 | 86.5% |
| AmoebaDB | *Acanthamoeba quina* Vil3 | 75,960,588 | 63,017,593 | 83.0% |
| AmoebaDB | *Acanthamoeba rhysodes* Singh | 64,616,007 | 53,929,813 | 83.5% |
| AmoebaDB | *Acanthamoeba* sp Galka | 78,249,914 | 63,414,386 | 81.0% |
| AmoebaDB | *Acanthamoeba* sp Incertae sedis | 77,638,095 | 71,166,671 | 91.7% |
| AmoebaDB | *Acanthamoeba* sp T4b-type | 83,119,178 | 64,574,355 | 77.7% |
| AmoebaDB | *Acanthamoeba triangularis* SH621 | 94,707,426 | 74,305,684 | 78.5% |
| AmoebaDB | *Entamoeba dispar* SAW760 | 22,825,791 | 9,112,384 | 39.9% |
| AmoebaDB | *Entamoeba histolytica* DS4-868 | 19,756,966 | 8,717,509 | 44.1% |
| AmoebaDB | *Entamoeba histolytica* HM-1:CA | 17,729,387 | 8,023,015 | 45.3% |
| AmoebaDB | *Entamoeba histolytica* HM-1:IMSS-A | 12,285,409 | 6,337,751 | 51.6% |
| AmoebaDB | *Entamoeba histolytica* HM-1:IMSS-B | 12,661,880 | 6,608,724 | 52.2% |
| AmoebaDB | *Entamoeba histolytica* HM-1:IMSS | 20,734,772 | 9,046,626 | 43.6% |
| AmoebaDB | *Entamoeba histolytica* HM-3:IMSS | 13,617,072 | 6,867,090 | 50.4% |
| AmoebaDB | *Entamoeba histolytica* KU27 | 15,171,051 | 7,490,305 | 49.4% |
| AmoebaDB | *Entamoeba histolytica* KU50 | 11,893,480 | 5,739,198 | 48.3% |
| AmoebaDB | *Entamoeba histolytica* MS96-3382 | 19,015,936 | 8,369,644 | 44.0% |
| AmoebaDB | *Entamoeba histolytica* Rahman | 23,309,729 | 8,688,150 | 37.3% |
| AmoebaDB | *Entamoeba invadens* IP1 | 40,506,505 | 23,410,257 | 57.8% |
| AmoebaDB | *Entamoeba moshkovskii* Laredo | 22,738,010 | 14,687,127 | 64.6% |
| AmoebaDB | *Entamoeba nuttalli* P19 | 14,351,590 | 7,063,189 | 49.2% |
| AmoebaDB | *Naegleria fowleri* ATCC 30863 | 28,634,883 | 23,333,454 | 81.5% |
| Type | Genome | Orig Length | Final Length | |
| CryptoDB | *Cryptosporidium baileyi* TAMU-09Q1 | 8,502,994 | 4,027,024 | 47.4% |
| CryptoDB | *Cryptosporidium hominis* | 9,050,842 | 6,395,187 | 70.7% |
| CryptoDB | *Cryptosporidium hominis* TU502 | 8,741,121 | 6,177,062 | 70.7% |
| CryptoDB | *Cryptosporidium hominis* TU502_2012 | 9,110,085 | 6,421,811 | 70.5% |
| CryptoDB | *Cryptosporidium hominis* UKH1 | 9,141,398 | 6,446,679 | 70.5% |
| CryptoDB | *Cryptosporidium meleagridis* UKMEL1 | 8,973,224 | 6,567,905 | 73.2% |
| CryptoDB | *Cryptosporidium muris* RN66 | 9,238,736 | 6,644,382 | 71.9% |
| CryptoDB | *Cryptosporidium parvum* Iowa II | 9,083,766 | 6,423,289 | 70.7% |
| CryptoDB | *Chromera velia* CCMP2878 | 193,306,556 | 140,041,267 | 72.4% |
| CryptoDB | *Gregarina niphandrodes* | 13,637,874 | 12,535,606 | 91.9% |
| CryptoDB | *Vitrella brassicaformis* CCMP3155 | 71,768,977 | 58,191,116 | 81.1% |
| Type | Genome | Orig Length | Final Length | |
| FungiDB | *Aspergillus aculeatus* ATCC 16872 | 35,185,919 | 31,227,227 | 88.7% |
| FungiDB | *Aphanomyces astaci* APO3 | 58,572,258 | 55,052,220 | 94.0% |
| FungiDB | *Albuco candida* 2VRR | 32,793,462 | 31,551,199 | 96.2% |

| | | | | |
|---|---|---|---|---|
| FungiDB | *Ajellomyces capsulatus* G186AR | 30,238,072 | 26,141,840 | 86.5% |
| FungiDB | *Ajellomyces capsulatus* NAm1 | 30,625,832 | 27,342,692 | 89.3% |
| FungiDB | *Aspergillus carbonarius* ITEM5010 | 34,247,686 | 31,339,808 | 91.5% |
| FungiDB | *Aspergillus clavatus* NRRL1 | 27,885,697 | 24,720,573 | 88.6% |
| FungiDB | *Aspergillus flavus* NRRL3357 | 36,829,644 | 35,106,178 | 95.3% |
| FungiDB | *Aspergillus fumigatus* Af293 | 28,841,706 | 27,625,228 | 95.8% |
| FungiDB | *Aphanomyces invadans* NJM9701 | 41,452,125 | 40,541,975 | 97.8% |
| FungiDB | *Albugo laibachii* Nc14 | 32,766,339 | 31,411,454 | 95.9% |
| FungiDB | *Allomyces macrogynus* ATCC38327 | 52,682,416 | 46,663,619 | 88.6% |
| FungiDB | *Aspergillus nidulans* FGSCA4 | 29,817,723 | 28,864,062 | 96.8% |
| FungiDB | *Aspergillus niger* ATCC1015 | 34,853,277 | 32,727,858 | 93.9% |
| FungiDB | *Aspergillus niger* CBS513-88 | 33,930,387 | 31,903,326 | 94% |
| FungiDB | *Aspergillus oryzae* RIB40 | 37,117,683 | 35,381,650 | 95.3% |
| FungiDB | *Aspergillus terreus* NIH2624 | 29,197,939 | 27,705,997 | 94.9% |
| FungiDB | *Batrachochytrium dendrobatidis* JEL423 | 23,403,618 | 22,029,972 | 94.1% |
| FungiDB | *Botryotinia fuckeliana* B05.10 | 38,867,192 | 34,959,939 | 89.9% |
| FungiDB | *Candida albicans* SC5314 | 14,320,608 | 10,725,103 | 74.9% |
| FungiDB | *Coprinopsis cinerea* okayama7#130 | 36,150,108 | 34,628,883 | 95.8% |
| FungiDB | *Cryptococcus deuterogattii* R265 | 17,161,958 | 16,157,647 | 94.1% |
| FungiDB | *Cryptococcus gattii* WM276 | 18,361,682 | 17,012,564 | 92.7% |
| FungiDB | *Candida glabrata* CBS 138 | 12,317,942 | 10,937,497 | 88.8% |
| FungiDB | *Coccidioides immitis* H538.4 | 25,599,272 | 23,699,859 | 92.6% |
| FungiDB | *Coccidioides immitis* RS | 28,438,790 | 26,105,184 | 91.8% |
| FungiDB | *Coccidioides immitis* RMSCC 2394 | 25,231,528 | 23,330,655 | 92.5% |
| FungiDB | *Coccidioides immitis* RMSCC 3703 | 29,015,619 | 26,657,761 | 91.9% |
| FungiDB | *Cryptoccocus neoformans* B-3501A | 18,519,479 | 17,393,527 | 93.9% |
| FungiDB | *Cryptoccocus neoformans* var. grubii H99 | 18,899,441 | 17,706,084 | 93.7% |
| FungiDB | *Cryptoccocus neoformans* JEC21 | 19,050,062 | 17,883,144 | 93.9% |
| FungiDB | *Coccidioides posadasii* C735 | 27,013,379 | 24,901,832 | 92.2% |
| FungiDB | *Coccidioides posadasii* CPA 0001 | 25,967,020 | 23,853,535 | 91.9% |
| FungiDB | *Coccidioides posadasii* CPA 0020 | 24,838,067 | 22,974,283 | 92.5% |
| FungiDB | *Coccidioides posadasii* CPA 0066 | 25,523,284 | 23,454,107 | 91.9% |
| FungiDB | *Coccidioides posadasii* RMSCC 1037 | 24,674,763 | 22,858,408 | 92.6% |
| FungiDB | *Coccidioides posadasii* RMSCC 1038 | 23,601,803 | 21,932,692 | 92.9% |
| FungiDB | *Coccidioides posadasii* RMSCC 2133 | 27,099,173 | 25,033,449 | 92.4% |
| FungiDB | *Coccidioides posadasii* RMSCC 3488 | 28,086,866 | 25,863,576 | 92.1% |
| FungiDB | *Coccidioides posadasii* RMSCC 3700 | 23,429,173 | 21,943,486 | 93.7% |
| FungiDB | *Coccidioides posadasii* str. Silveira | 27,427,344 | 25,330,651 | 92.4% |
| FungiDB | *Fusarium graminearum* PH-1 | 36,223,641 | 34,755,316 | 95.9% |
| FungiDB | *Fusarium oxysporum* sp. lycopersici 4287 | 59,936,783 | 57,585,630 | 96.1% |
| FungiDB | *Fusarium verticillioides* 7600 | 41,700,345 | 39,989,208 | 95.9% |
| FungiDB | *Hyaloperonospora arabidopsis* Emoy2 | 70,831,685 | 67,916,852 | 95.9% |
| FungiDB | *Mucor circinelloides* CBS 277.49 | 36,587,022 | 32,884,834 | 89.9% |
| FungiDB | *Malassezia globosa* CBS 7966 | 8,958,094 | 8,651,600 | 96.6% |
| FungiDB | *Melampsora larici-populina* 98AG31 | 97,682,699 | 86,308,373 | 88.4% |

| | | | | |
|---|---|---:|---:|---:|
| FungiDB | *Magnaporthe oryzae* 70-15 | 41,504,533 | 38,052,678 | 91.7% |
| FungiDB | *Neurospora crassa* OR74A | 41,061,603 | 34,500,349 | 84% |
| FungiDB | *Neurospora discreta* FGSC8579 | 37,145,397 | 32,331,446 | 87% |
| FungiDB | *Neosatorya fischeri* NRRL 181 | 31,760,917 | 29,820,025 | 93.9% |
| FungiDB | *Neurospora tetrasperma* FGSC2508 | 38,490,826 | 33,482,562 | 87% |
| FungiDB | *Pythium aphanidermatum* | 34,264,281 | 33,318,281 | 97.2% |
| FungiDB | *Pythium arrhenomanes* | 42,813,264 | 40,437,665 | 94.5% |
| FungiDB | *Phycomyces blakesleeanus* NRRL 1555 | 53,368,881 | 37,402,174 | 70.1% |
| FungiDB | *Phytophthora capsici* LT1534 | 56,042,007 | 54,100,541 | 96.5% |
| FungiDB | *Phanerochaete chrysosporium* RP-78 | 32,504,098 | 31,154,123 | 95.8% |
| FungiDB | *Phytophthora cinnamomi* CBS 144.22 | 58,248,003 | 55,862,880 | 95.9% |
| FungiDB | *Puccinia graminis* f.sp. Tritici CRL 7-536-700-3 | 81,521,292 | 69,040,482 | 84.7% |
| FungiDB | *Phytophthora infestans* T30-4 | 190,133,476 | 182,998,942 | 96.2% |
| FungiDB | *Pythium irregulare* DAOM BR486 | 42,676,619 | 40,260,460 | 94.3% |
| FungiDB | *Pythium iwayamai* DAOM BR22034 | 41,665,904 | 39,368,273 | 94.5% |
| FungiDB | *Pneumocystis jirovecii* SE8 | 8,152,511 | 4,527,610 | 55.5% |
| FungiDB | *Phytophthora parasitica* INRA-310 | 53,871,265 | 52,042,370 | 96.6% |
| FungiDB | *Phytophthora ramorum* strain Pr102 | 54,424,536 | 51,287,109 | 94.2% |
| FungiDB | *Phytophthora sojae* strain P6497 | 79,331,234 | 75,782,571 | 95.5% |
| FungiDB | *Pythium ultimum* BR650 | 35,611,117 | 33,254,115 | 93.4% |
| FungiDB | *Pythium ultimum* DAOM BR144 | 42,791,577 | 40,425,968 | 94.5% |
| FungiDB | *Pythium vexans* DAOM BR484 | 33,582,665 | 31,805,220 | 94.7% |
| FungiDB | *Rhizopus delemar* RA 99-880 | 45,303,457 | 36,279,280 | 80.1% |
| FungiDB | *Saccharomyces cerevisiae* S288c | 12,157,105 | 10,560,358 | 86.9% |
| FungiDB | *Saprolegnia diclina* VS20 | 40,460,948 | 39,216,331 | 96.9% |
| FungiDB | *Schizosaccharomyces japonicus* yFS275 | 11,216,055 | 10,320,906 | 92% |
| FungiDB | *Sordaria macrospora* k-hell | 39,814,042 | 36,162,566 | 90.8% |
| FungiDB | *Schizosaccharomyces octosporus* | 11,307,207 | 9,772,309 | 86.4% |
| FungiDB | *Saprolegnia parasitica* CVS 223.65 | 48,138,513 | 46,578,255 | 96.8% |
| FungiDB | *Schizosaccharomyces pombe* 972h | 12,630,977 | 10,649,585 | 84.3% |
| FungiDB | *Spizellomyces punctatus* DAOM BR117 | 23,906,001 | 22,722,876 | 95.1% |
| FungiDB | *Sporisorium reilianum* SRZ2 | 17,998,092 | 16,533,450 | 91.9% |
| FungiDB | *Sclerotinia sclerotiorum* | 38,001,451 | 33,827,555 | 89% |
| FungiDB | *Talaromyces marneffei* ATCC 18224 | 28,467,480 | 26,991,013 | 94.8% |
| FungiDB | *Tremella mesenterica* DSM 1558 | 27,987,508 | 25,965,761 | 92.8% |
| FungiDB | *Trichoderma reesei* QM6a | 33,348,438 | 29,762,710 | 89.2% |
| FungiDB | *Talaromyces stipitatus* ATCC 10500 | 35,558,430 | 34,098,032 | 95.9% |
| FungiDB | *Ustilago maydis* 521 | 19,641,656 | 18,814,259 | 95.8% |
| FungiDB | *Yarrowia lipolytica* CLIB122 | 20,501,810 | 18,893,458 | 92.2% |
| Type | Genome | Orig Length | Final Length | |
| GiardiaDB | *Giardia intestinalis* A isolate WB | 10,703,889 | 10,467,715 | 97.8 % |
| GiardiaDB | *Giardia intestinalis* A2 isolate DH | 11,192,174 | 10,918,617 | 97.6 % |
| GiardiaDB | *Giardia intestinalis* B isolate GS | 10,998,431 | 10,757,478 | 97.8 % |
| GiardiaDB | *Giardia intestinalis* B isolate GS_B | 12,009,619 | 11,767,306 | 98.0 % |
| GiardiaDB | *Giardia intestinalis* E isolate P15 | 11,521,527 | 11,237,712 | 97.5 % |
| GiardiaDB | *Spironucleus salmonicida* ATCC 50377 | 12,893,052 | 9,474,143 | 73.5 % |

| Type | Genome | Orig Length | Final Length | |
|---|---|---|---|---|
| MicrosporidiaDB | *Anncaliia algerae* PRA109 | 13,851,317 | 5,200,550 | 37.5% |
| MicrosporidiaDB | *Anncaliia algerae* PRA339 | 9,877,948 | 3,924,651 | 39.7% |
| MicrosporidiaDB | *Anncaliia algerae* Undeen | 13,803,782 | 6,147,817 | 44.5% |
| MicrosporidiaDB | *Edhazardia aedis* USNM 41457 | 46,603,333 | 12,448,222 | 26.7% |
| MicrosporidiaDB | *Enterocytozoon bieneusi* H348 | 3,859,221 | 2,375,928 | 61.6% |
| MicrosporidiaDB | *Encephalitozoon cuniculi* EC1 | 2,240,501 | 2,149,920 | 96.0% |
| MicrosporidiaDB | *Encephalitozoon cuniculi* EC2 | 2,241,415 | 2,152,041 | 96.0% |
| MicrosporidiaDB | *Encephalitozoon cuniculi* EC3 | 2,235,625 | 2,146,050 | 96.0% |
| MicrosporidiaDB | *Encephalitozoon cuniculi* GB-M1 | 2,496,714 | 2,357,956 | 94.4% |
| MicrosporidiaDB | *Encephalitozoon hellem* Swiss | 2,182,433 | 2,065,044 | 94.6% |
| MicrosporidiaDB | *Encephalitozoon intestinalis* ATCC 50506 | 2,216,798 | 2,050,267 | 92.5% |
| MicrosporidiaDB | *Encephalitozoon romaleae* SJ-2008 | 2,187,587 | 2,047,833 | 93.6% |
| MicrosporidiaDB | *Hamiltosporidium tvaerminnensis* OER-3-3 | 13,270,809 | 6,120,415 | 46.1% |
| MicrosporidiaDB | *Mitosporidium daphniae* UGP3 | 5,636,645 | 5,133,874 | 91.1% |
| MicrosporidiaDB | *Nosema bombycis* CQ1 | 14,356,492 | 8,163,893 | 56.9% |
| MicrosporidiaDB | *Nosema ceranae* BRL01 | 7,860,219 | 3,313,417 | 42.2% |
| MicrosporidiaDB | *Nematocida ausubeli* | 4,649,639 | 3,948,126 | 84.9% |
| MicrosporidiaDB | *Nematocida parisii* ERTm1 | 4,029,056 | 3,056,154 | 75.9% |
| MicrosporidiaDB | *Nematocida parisii* ERTm3 | 4,121,387 | 3,132,651 | 76% |
| MicrosporidiaDB | *Nematocida ausubeli* | 4,228,442 | 3,624,837 | 85.7% |
| MicrosporidiaDB | *Ordospora colligata* OC4 | 2,290,527 | 2,098,057 | 91.6% |
| MicrosporidiaDB | *Spraguea lophii* 42_110 | 4,979,932 | 1,720,891 | 34.6% |
| MicrosporidiaDB | *Trachipleistophora hominis* | 7,706,555 | 6,131,376 | 79.6% |
| MicrosporidiaDB | *Vittaforma corneae* ATCC 50505 | 3,148,732 | 2,754,939 | 87.5% |
| MicrosporidiaDB | *Vavraia culicis* subsp. floridensis | 6,033,822 | 5,448,931 | 90.3% |
| Type | Genome | Orig Length | Final Length | |
| PiroplasmaDB | *Babesia bigemina* strain BOND | 13,840,936 | 13,152,126 | 95.0% |
| PiroplasmaDB | *Babesia bovis* T2Bo | 8,179,705 | 7,793,821 | 95.3% |
| PiroplasmaDB | *Babesia microti* strain RI | 6,392,336 | 5,626,076 | 88% |
| PiroplasmaDB | *Cytauxzoon felis* strain Winnie | 9,110,257 | 6,621,491 | 72.7% |
| PiroplasmaDB | *Theileria annulata* strain Ankara | 8,357,924 | 6,054,564 | 72.4% |
| PiroplasmaDB | *Theileria equi* strain WA | 11,674,476 | 11,027,791 | 94.5% |
| PiroplasmaDB | *Theileria orientalis* strain Shintoku | 9,006,764 | 7,652,828 | 85.0% |
| PiroplasmaDB | *Theileria parva* strain Muguga | 8,353,489 | 6,313,179 | 75.6% |
| Type | Genome | Orig Length | Final Length | |
| PlasmoDB | *Plasmodium coatneyi* Hackeri | 27,691,932 | 18,780,353 | 67.8% |
| PlasmoDB | *Plasmodium cynomolgi* strain B | 25,338,238 | 15,993,945 | 63.1% |
| PlasmoDB | *Plasmodium falciparum* IT | 22,608,450 | 3,623,247 | 16% |
| PlasmoDB | *Plasmodium gallinaceum* 8A | 16,919,478 | 2,725,441 | 16.1% |
| PlasmoDB | *Plasmodium reichenowi* CDC | 23,777,383 | 3,956,350 | 16.6% |
| PlasmoDB | *Plasmodium vivax*-like Pvl01 | 26,960,177 | 16,696,343 | 61.9% |
| PlasmoDB | *Plasmodium yoelii yoelii* 17X | 22,573,807 | 4,256,123 | 18.9% |
| PlasmoDB | *Plasmodium yoelii yoelii* 17XNL | 22,923,632 | 4,268,426 | 18.6% |
| PlasmoDB | *Plasmodium yoelii yoelii* YM | 21,430,035 | 4,047,021 | 18.9% |

| Type | Genome | Orig Length | Final Length | |
|---|---|---:|---:|---:|
| ToxoDB | *Cyclospora cayetanensis* strain CHN_HEN01 | 46,076,418 | 40,885,607 | 88.7% |
| ToxoDB | *Eimeria acervulina* Houghton | 45,677,409 | 23,000,742 | 50.4% |
| ToxoDB | *Eimeria brunetti* Houghton | 65,020,649 | 31,842,885 | 49.0% |
| ToxoDB | *Eimeria falciformis* Bayer Haberkorn | 41,633,439 | 30,420,208 | 73.1% |
| ToxoDB | *Eimeria maxima* Weybridge | 45,874,462 | 25,082,861 | 54.7% |
| ToxoDB | *Eimeria mitis* Houghton | 66,895,571 | 32,426,613 | 48.5% |
| ToxoDB | *Eimeria necatrix* Houghton | 54,911,932 | 32,308,343 | 58.8% |
| ToxoDB | *Eimeria praecox* Houghton | 55,968,490 | 26,973,281 | 48.2% |
| ToxoDB | *Eimeria tenella* strain Houghton | 51,173,562 | 30,699,534 | 60.0% |
| ToxoDB | *Hammondia hammondi* strain H.H.34 | 67,460,985 | 57,295,789 | 84.9% |
| ToxoDB | *Neospora caninum* Liverpool | 59,079,711 | 52,342,634 | 88.6% |
| ToxoDB | *Sarcocystis neurona* | 117,871,271 | 84,838,398 | 72.0% |
| ToxoDB | *Toxoplasma gondii* ARI | 63,082,652 | 55,370,610 | 87.8% |
| ToxoDB | *Toxoplasma gondii* CAST | 63,046,868 | 55,340,458 | 87.8% |
| ToxoDB | *Toxoplasma gondii* COUG | 63,695,689 | 55,558,767 | 87.2% |
| ToxoDB | *Toxoplasma gondii* CtCo5 | 62,620,635 | 54,910,445 | 87.7% |
| ToxoDB | *Toxoplasma gondii* FOU | 61,897,817 | 54,334,863 | 87.8% |
| ToxoDB | *Toxoplasma gondii* GAB2-2007-GAL-DOM2 | 62,977,178 | 55,266,195 | 87.8% |
| ToxoDB | *Toxoplasma gondii* GT1 | 63,916,432 | 56,172,934 | 87.9% |
| ToxoDB | *Toxoplasma gondii* MAS | 61,483,819 | 54,038,607 | 87.9% |
| ToxoDB | *Toxoplasma gondii* ME49 | 65,463,023 | 57,092,239 | 87.2% |
| ToxoDB | *Toxoplasma gondii* p89 | 61,877,399 | 54,370,919 | 87.9% |
| ToxoDB | *Toxoplasma gondii* RH | 4,034,738 | 3,539,244 | 87.7% |
| ToxoDB | *Toxoplasma gondii* RUB | 62,609,256 | 55,003,468 | 87.9% |
| ToxoDB | *Toxoplasma gondii* TgCATBr5 | 61,636,148 | 54,079,463 | 87.7% |
| ToxoDB | *Toxoplasma gondii* TgCATBr9 | 61,824,191 | 54,377,809 | 88.0% |
| ToxoDB | *Toxoplasma gondii* TgCatPRC2 | 62,981,717 | 55,005,780 | 87.3% |
| ToxoDB | *Toxoplasma gondii* TgCkUg2 | 41,929,350 | 37,103,249 | 88.5% |
| ToxoDB | *Toxoplasma gondii* VAND | 62,334,980 | 54,725,606 | 87.8% |
| ToxoDB | *Toxoplasma gondii* VEG | 63,535,336 | 55,816,152 | 87.9% |
| Type | Genome | Orig Length | Final Length | |
| TrichDB | *Trichomonas vaginalis* G3 | 175,592,576 | 133,413,618 | 76.0% |

| Type | Genome | Orig Length | Final Length | |
|---|---|---:|---:|---:|
| TriTrypDB | *Endotrypanum monterogeii* LV88 | 32,006,254 | 25,442,985 | 79.5% |
| TriTrypDB | *Leishmania aethiopica* L147 | 30,987,107 | 26,122,578 | 84.3% |
| TriTrypDB | *Leishmania amazonensis* MHOM/BR/71973/M2269 | 29,003,854 | 24,729,684 | 85.3% |
| TriTrypDB | *Leishmania arabica* LEM1108 | 30,769,451 | 26,094,063 | 84.8% |
| TriTrypDB | *Leishmania braziliensis* MHOM/BR/75/M2903 | 32,474,516 | 27,565,352 | 84.9% |
| TriTrypDB | *Leishmania braziliensis* MHOM/BR/75/M2904 | 31,996,772 | 27,334,656 | 85.4% |
| TriTrypDB | *Leishmania donovani* strain BHU 1220 | 31,201,868 | 26,747,829 | 85.7% |
| TriTrypDB | *Leishmania donovani* strain BPK282A1 | 31,252,135 | 26,776,266 | 85.7% |
| TriTrypDB | *Leishmania enriettii* strain LEM3045 | 30,426,963 | 26,787,339 | 88.0% |
| TriTrypDB | *Leishmania gerbilli* strain LEM452 | 30,817,898 | 25,943,276 | 84.2% |
| TriTrypDB | *Leishmania infantum* JPCM5 | 32,101,728 | 27,259,756 | 84.9% |
| TriTrypDB | *Leishmania major* strain Friedlin | 32,855,082 | 27,609,648 | 84.0% |
| TriTrypDB | *Leishmania major* strain LV39c5 | 31,923,298 | 26,995,213 | 84.6% |
| TriTrypDB | *Leishmania major* strain SD 75.1 | 31,157,115 | 26,458,979 | 84.9% |
| TriTrypDB | *Leishmania mexicana* MHOM/GT/2001/U1103 | 32,074,503 | 26,912,815 | 83.9% |
| TriTrypDB | *Leishmania panamensis* MHOM/COL/81/L13 | 30,964,489 | 26,462,045 | 85.5% |
| TriTrypDB | *Leptomonas pyrrhocoris* H10 | 30,266,257 | 25,517,685 | 84.3% |
| TriTrypDB | *Leptomonas seyomouri* ATCC 30220 | 27,617,457 | 24,322,244 | 88.1% |
| TriTrypDB | *Leishmania* sp. MAR LEM2494 | 30,528,139 | 26,182,127 | 85.8% |
| TriTrypDB | *Leishmania tarentolae* Parrot-TarII | 30,440,719 | 26,681,639 | 87.7% |
| TriTrypDB | *Leishmania tropica* L590 | 31,322,741 | 26,386,117 | 84.2% |
| TriTrypDB | *Leishmania turanica* strain LEM423 | 30,870,945 | 25,680,606 | 83.2% |
| TriTrypDB | *Trypanosoma brucei* gambiense DAL972 | 22,110,721 | 18,709,859 | 84.6% |
| TriTrypDB | *Trypanosoma brucei* Lister strain 427 | 25,703,564 | 21,093,540 | 82.1% |
| TriTrypDB | *Trypanosoma brucei* brucei TREU927 | 35,816,880 | 29,307,744 | 81.8% |
| TriTrypDB | *Trypanosoma congolense* IL3000 | 34,080,344 | 29,545,872 | 86.7% |
| TriTrypDB | *Trypanosoma cruzi* strain CL Brener | 25,827,529 | 20,606,810 | 79.8% |
| TriTrypDB | *Trypanosoma cruzi* CL Brener Esmeraldo-like | 36,032,823 | 30,297,678 | 84.1% |
| TriTrypDB | *Trypanosoma cruzi* CL Brener Non-Esmeraldo-like | 27,751,085 | 22,461,132 | 80.9% |
| TriTrypDB | *Trypanosoma cruzi* DM28c | 27,304,309 | 22,335,463 | 81.8% |
| TriTrypDB | *Trypanosoma cruzi* strain Esmeraldo | 34,967,951 | 28,007,865 | 80.1% |
| TriTrypDB | *Trypanosoma cruzi* JR cl. 4 | 40,114,148 | 32,504,101 | 81.0% |
| TriTrypDB | *Trypanosoma cruzi* marinkellei strain B7 | 34,233,090 | 27,383,012 | 80.0% |
| TriTrypDB | *Trypanosoma cruzi* Sylvio X10//1 | 38,589,511 | 31,510,663 | 81.7% |
| TriTrypDB | *Trypanosoma cruzi* Tula cl2 | 74,005,907 | 60,022,805 | 81.1% |
| TriTrypDB | *Trypanosoma evansi* strain STIB 805 | 25,432,062 | 21,367,606 | 84.0% |
| TriTrypDB | *Trypanosoma grayi* ANR4 | 20,809,961 | 17,856,524 | 85.8% |
| TriTrypDB | *Trypanosoma rangeli* SC58 | 14,016,406 | 11,878,107 | 84.7% |
| TriTrypDB | *Trypanosoma vivax* Y486 | 41,775,787 | 36,041,565 | 86.3% |

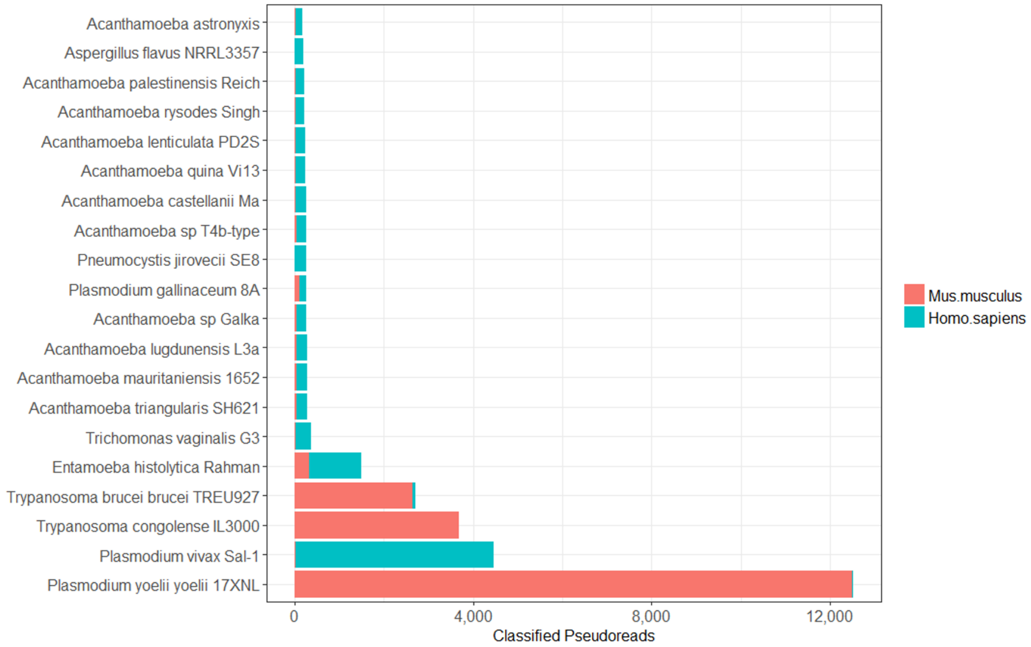


Figure 3.4: **Human/Mouse classified pseudo-reads.** This plot shows the 20 genomes with the most number of pseudo-reads classified as either human or mouse. Perhaps not surprisingly, the mouse strain of malaria, *P. yoelii*, contains a substantial number of contaminant reads from mouse.

remaining pseudo-reads against the vertebrate database, I masked a much smaller amount of sequence, with only 0.1% of each genome matching vertebrate sequences in this step.

The most contaminated eukaryotic pathogen genomes are the three *Plasmodium yoelii* genomes (strains 17XNL, YM, and 17X), with approximately 60% of the genomes identified as human/bacterial/viral/archaeal (Figures 3.3 and 3.4). The primary sources of contamination in these three genomes were *Methylococcus capsulatus* (16,000 pseudo-reads) and the mouse genome ($12,000$ pseudo-reads). The genome for *Plasmodium vivax Sal-1*, which causes malaria in humans, contained the greatest amount of human contamination, with more than 4, 000 pseudo-reads classified as Homo sapiens. *Entamoeba histolytica Rahman*, a human intestinal parasite, is also notably contaminated, with nearly 50% of its genome identified as either human or bacteria (**Figures 3.3 and 3.4**).

Other eukaryotic pathogens that underwent significant masking due to contamination include *Plasmodium gallinaceum 8A* (62% masked), *Plasmodium falciparum IT* (57% masked), *Plasmodium reichenowi CDC* (55% masked). Each of these pathogens contained significant contamination likely due to host DNA, as the masked pseudo-reads were identified as matching their original host. For example, *Plasmodium gallinaceum* causes malaria in poultry and 11,700 pseudo-reads were identified as chicken DNA [55]. Although *Plasmodium falciparum* is a human malarial parasite, it originated from the gorilla malarial parasite [56]. More than 450 pseudo-reads for *Plasmodium falciparum* were identified as gorilla. Similarly, *Plasmodium reichenowi* is a malarial parasite in chimpanzees and was one of only two Plasmodium genomes to have chimpanzee pseudo-reads [56]. Interestingly, *Edhazardia aedis* had 55% of its genome length masked, but had very few classified pseudo-reads. Instead, the majority of its non-masked sequences to begin with were stretches of DNA less than 100bp. Over 358, 000 individual sequences were very small contigs, shorter than 100bp which are masked due to length.

## 3.4 Eukaryotic Pathogen Detection in Human Cornea Samples

To measure the effectiveness of this database cleaning method for NGS diagnosis of human infections, I evaluated a set of 20 human cornea samples recently described by Li et. al 2018 [57] against our EuPathDB-clean. The 20 corneal samples include 4 bacterial infections, 9 eukaryotic pathogen infections, 3 herpes virus infections, and 4 controls. Details about these samples and the true positive pathogens in each sample are listed in **Table 3.4**.

I first used Bowtie2 to align all corneal sample reads against the human genome reference, GRCh38.p7, and extracted any unaligned reads for each sample (**Table 3.4**). The non-human reads from each sample were then classified against Kraken databases generated from 1) the original eukaryotic pathogen genomes, 2) the eukaryotic pathogen genomes after removal of bacterial, viral, archaeal, and plant contamination and 3) the final cleaned eukaryotic pathogen genomes. The second set of genomes did not yet undergo removal of human and vertebrate contamination.

**Figure 3.5** summarizes the results when using each of databases to identify the pathogens in these samples. The classifications differed greatly depending on the database used, demonstrating the importance of database selection prior to the computational analysis of any NGS sample. However, in the case of diagnostics, the

Table 3.4: **Cornea sample true positives.** This table summarizes the pathogens present in each of the corneal samples. Metagenomic shotgun sequencing was performed on all samples as described in [31] generating from 20–46 million pairs of 75-bp reads per sample. Sequencing was done in two batches of 10 samples each, where the 10 samples were multiplexed.

| Case # | True Positives | Total Reads | Non-Human Reads |
|---|---|---:|---:|
| Case 1 | *Staphyloccoccus aureus* | 35,947,243 | 8,166 |
| Case 2 | *Streptococcus agalactiae* | 42,281,022 | 2,354,821 |
| Case 3 | *Mycobacterium* | 32,321,057 | 1,440,343 |
| Case 4 | *Mycobacterium chelonae* | 31,259,428 | 2,927,088 |
| Case 5 | *Candida parapsilosis* | 22,572,576 | 3,615,840 |
| Case 6 | *Fusarium solani* | 43,187,311 | 3,048,256 |
| Case 7 | *Candida albicans/dubliensis* | 45,410,366 | 1,993,853 |
| Case 8 | *Curvularia* | 42,359,755 | 3,181,901 |
| Case 9 | *Aspergillus flavus* | 46,033,752 | 2,875,199 |
| Case 10 | *Anncaliia algerae* | 20,060,037 | 2,756,229 |
| Case 11 | *Acanthamoeba* | 43,742,352 | 2,880,293 |
| Case 12 | *Acanthamoeba* | 46,648,496 | 3,602,638 |
| Case 13 | *Acanthamoeba* | 44,554,101 | 3,472,961 |
| Case 14 | *Herpes simplex type 1* | 22,460,961 | 1,470,059 |
| Case 15 | *Herpes simplex type 1* | 25,512,845 | 1,411,580 |
| Case 16 | *Herpes simplex type 1* | 23,749,398 | 3,874,558 |
| Case 17 | None | 43,643,461 | 2,637,693 |
| Case 18 | None | 45,824,224 | 2,341,716 |
| Case 19 | None | 25,623,975 | 1,071,939 |
| Case 20 | None | 25,202,226 | 1,823,615 |

contamination in the raw (unprocessed) genome databases creates false positive signals that overwhelm the true pathogen of the samples. For example, classification with the original EuPathDB presents *Toxoplasma gondii* as one of the primary infections in all but one of the corneal samples (**Figure 3.5A**). However, none of the cornea samples had infections by *Toxoplasma gondii* [57], making this classification a false positive.

The contamination removal process masked on average 5% of each *Toxoplasma gondii*

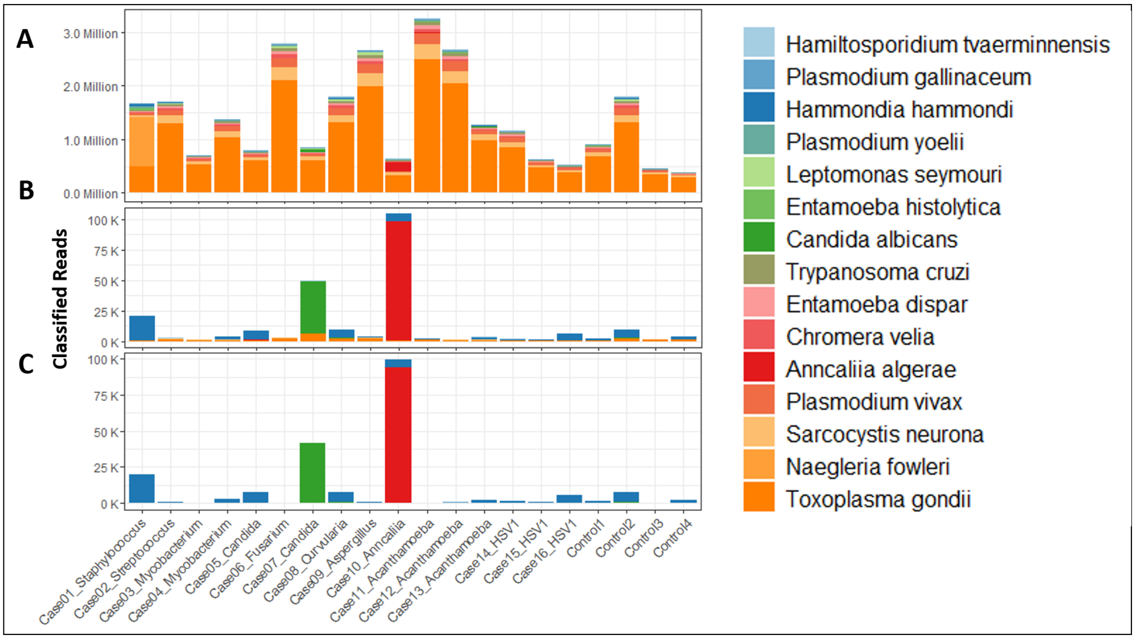


Figure 3.5: **Top 15 species identified in corneal samples when classified with the original EuPathDB-28 database.** The non-human reads from the 20 corneal samples were classified against three Kraken databases: the original EuPathDB **(A)**, EuPathDB without bacterial/archaeal/viral/plant contamination **(B)**, and the final EuPathDB-28 with additional removal of human/vertebrate contamination **(C)**. The plot above focuses on the 15 species with the most classified reads when classifying the corneal samples against the original EuPathDB. The plot compares how the number of classified reads changed when contaminating sequences were removed from the eukaryotic pathogen genomes.

genome. For example, the initial *Toxoplasma gondii ME49* genome is $\sim$ 60 Mb long and the final masked genome is 57 Mb. Fortunately, removing this relatively small proportion of the genome produced a cleaned database with a far better classification profile for the corneal samples. As shown in **Figure 3.5C**, the correct eukaryotic infections for Cases 7, 9, 10, 11, and 12 are immediately evident with the new database. Instead of thousands of reads identified as *Toxoplasma gondii*, the new database shows very high (and correct) read counts for *Anncaliia algerae* in Case 10, *Candida albi-*

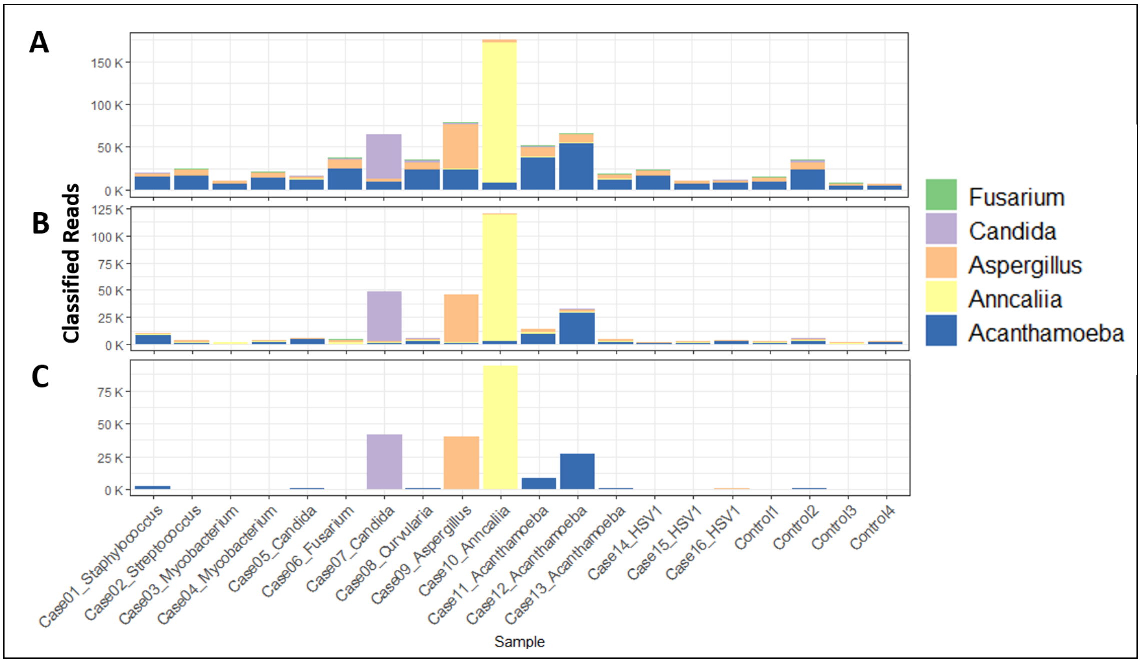


Figure 3.6: **Classified reads for the true positive genera in the corneal samples for the EuPathDB-28 databases.** The above plot compares the number of classified reads for the true eukaryotic pathogens in the infected samples when classifying the samples against: the original EuPathDB **(A)**, EuPathDB without bacterial/archaeal/viral/plant contamination **(B)**, and the final EuPathDB-28 with additional removal of human/vertebrate contamination **(C)**. The true pathogens are *Fusarium* (Case 6), *Candida* (Case 7), *Aspergilllus* (Case 9), *Anncaliia* (Case 10), and *Acanthamoeba* (Cases 11-13).

*cans* in Case 7, *Aspergillus* in Case 9, and *Acanthamoeba* in Cases 11 and 12, all true positive infections. With EuPathDB-clean, the maximum number of reads labeled as *Toxoplasma gondii* in any single sample was 24.

Another way to look at the data is to examine the read counts for the true positive genera only, as shown in **Figure 3.6**. Here I show the number of reads in each sample that were assigned to the 5 eukaryotic pathogens known to be present in at least one of the samples. With the original EuPathDB, the non-infected samples, alongside the truly infected samples, all appear to have numerous reads classified as *Acanthamoeba* or *Aspergillus* (**Figure 3.6A**). Upon removal of bacterial, archaeal, and viral contamination, *Acanthamoeba* reads were mainly identified in the Case 12 and 13 corneal sample while *Aspergillus* reads were mainly identified in the Case 9 corneal sample (**Figure 3.6B**). However, the corneal samples without eukaryotic pathogen infections continued to have a few thousand false positives. By comparison, the final EuPathDB-clean (where human and vertebrate contamination was also removed), identified less than 10 *Aspergillus flavus* reads in all non-*Aspergillus*-infected samples while maintaining a strong signal for *Aspergillus flavus* in Case 9.

## 3.5 EuPathDB-46

As of November 6, 2019, the most up to date Eukaryotic Pathogen Database is EuPathDB release 46 (released on November 6, 2019), which now contains 388 genomes [52]. **Table 3.5** displays the increased number of genomes between 2018 and 2020 for each sub-database. The increase in the number of genomes is largely due to the addition of genomes representing various strain of a single species. For example, EuPathDB-28 only contained one *Plasmodium falciparum* genome representing *Plasmodium falciparum IT* while EuPathDB-46 now contains 16 different *Plasmodium falciparum* strains. Similarly, in FungiDB-28, there is only one *Fusarium oxysporum* genome; in FungiDB-46, there are seven *Fusarium oxysporum* genomes, each

Table 3.5: **Composition of EuPathDB-28 and EuPathDB-46.**

| Sub-Database | EuPathDB-28 | EuPathDB-46 |
|---|---:|---:|
| AmoebaDB | 29 | 30 |
| CryptoDB | 11 | 18 |
| FungiDB | 87 | 164 |
| GiardiaDB | 6 | 10 |
| MicrosporidiaDB | 25 | 35 |
| PiroplasmaDB | 8 | 10 |
| PlasmoDB | 9 | 45 |
| ToxoDB | 30 | 33 |
| TrichDB | 1 | 1 |
| TriTrypDB | 39 | 42 |
| Total | 245 | 388 |

representing a different strain. In addition to the increased representation of some species, there are 94 new species introduced in EuPathDB-46 that were not present in EuPathDB-28 (**Table 3.6**).

Following the release of the new database version, I repeated the previously described contamination removal process for the EuPathDB-46 genomes. **Figure 3.7** summarizes the masked sequence length for each step. Following the removal of low-complexity, bacterial, viral, plant, vector, and vertebrate sequences, the majority of the eukaryotic pathogen genomes retained at least 80% of their original genome length. On average, the remaining genome length after cleaning is 87% of the original genome length.

However, while the majority of sequences retained a significant portion of their genome length, *Plasmodium* genomes on average had 50% of their genome length remaining. A significant portion of *Plasmodium* genome lengths were masked due to low complexity

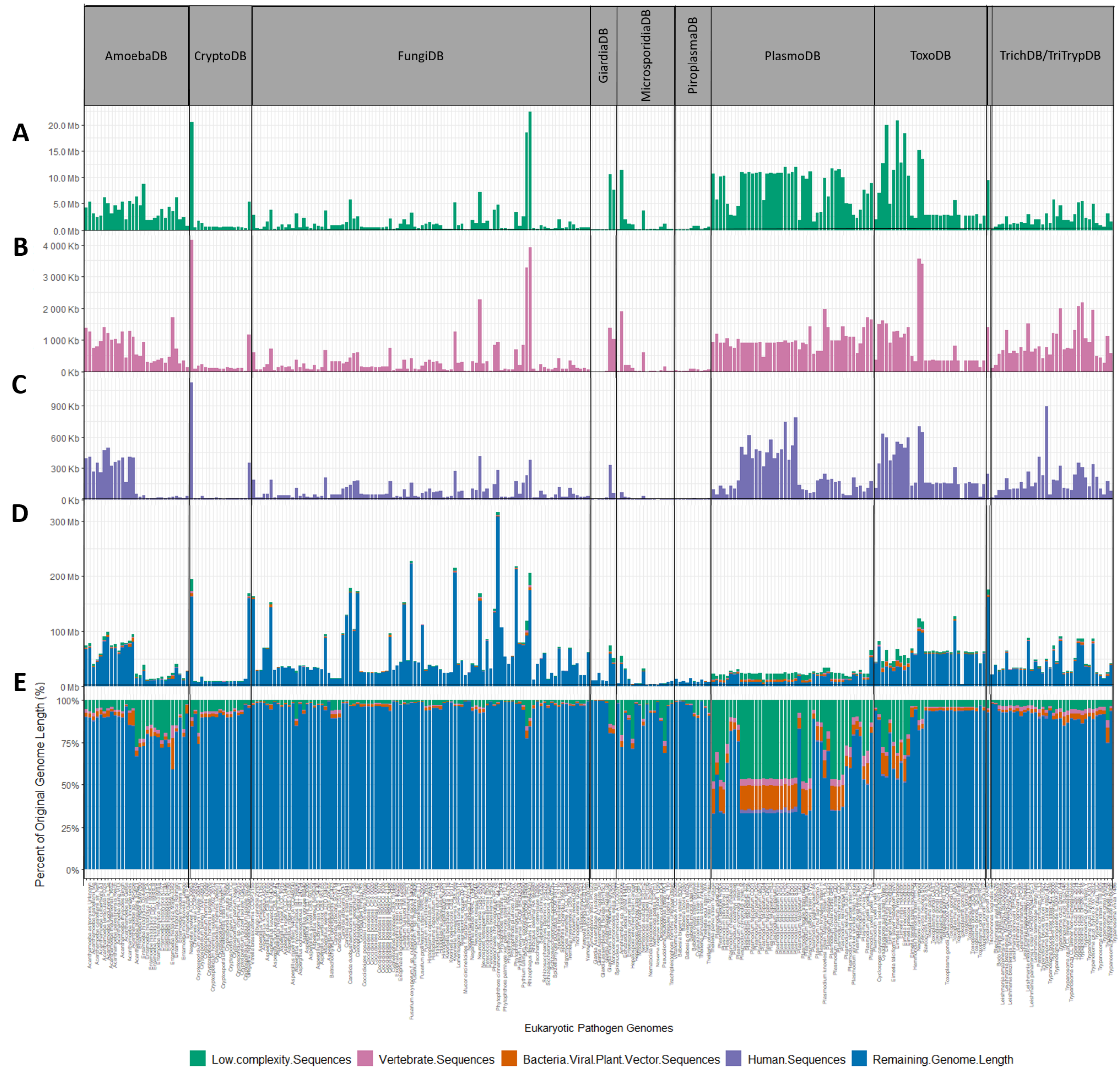


Figure 3.7: **EuPathDB-46 Masking results D)** provides an overview of sequence lengths masked in each step and the sequence lengths of the final cleaned genomes. As low-complexity sequences and vertebrate masked sequences are much smaller compared to the final genome length or bacterial/viral/plant/vector sequences, these were additionally plotted in **A)** and **B)** for each eukaryotic pathogen genome. Human masked sequences are plotted in **C)**. Masked sequence lengths are also presented as percentages of the original genome length to show the percent of each genome remaining and the percent masked in each step **E)**.

Table 3.6: **New Species in EuPathDB-46**. This table lists, for each sub-database, the new species represented in EuPathDB-46 that were not previously represented in EuPathDB-28.

| | | | | | |
|---|---|---|---|---|---|
| Crypto | *Cryptosporidium andersoni* | Fungi | *Clavispora lusitaniae* | Giardia | *Giardia Assemblage* |
| | *Cryptosporidium sp.* | | *Cryptococcus gattii* | | *Giardia muris* |
| | *Cryptosporidium tyzzeri* | | *Cryptococcus neoformans* | | *Monocercomonoides exilis* |
| | *Cryptosporidium ubiquitum* | | *Cyphellophora europaea* | Microsporidia | *Enterocytozoon hepatopenaei* |
| Fungi | *Amauroascus mutatus* | | *Exophiala mesophila* | | *Enterospora canceri* |
| | *Amauroascus niger* | | *Exophiala oligosperma* | | *Hepatospora eriocheir* |
| | *Aspergillus brasiliensis* | | *Fonsecaea pedrosoi* | | *Nematocida ausubeli* |
| | *Aspergillus campestris* | | *Fusarium fujikuroi* | | *Nematocida displodere* |
| | *Aspergillus fischeri* | | *Fusarium proliferatum* | | *Pseudoloma neurophilia* |
| | *Aspergillus glaucus* | | *Histoplasma capsulatum* | Piroplasma/Plasmo | *Babesia divergens* |
| | *Aspergillus kawachii* | | *Hyaloperonospora arabidopsidis* | | *Babesia ovata* |
| | *Aspergillus luchuensis* | | *Kwoniella bestiolae* | | *Plasmodium adleri* |
| | *Aspergillus novofumigatus* | | *Kwoniella dejecticola* | | *Plasmodium berghei* |
| | *Aspergillus ochraceoroseus* | | *Kwoniella heveanensis* | | *Plasmodium billcollinsi* |
| | *Aspergillus steynii* | | *Lomentospora prolificans* | | *Plasmodium blacklocki* |
| | *Aspergillus sydowii* | | *Malassezia restricta* | | *Plasmodium chabaudi* |
| | *Aspergillus tubingensis* | | *Naganishia albida* | | *Plasmodium fragile* |
| | *Aspergillus versicolor* | | *Paracoccidioides brasiliensis* | | *Plasmodium gaboni* |
| | *Aspergillus wentii* | | *Paracoccidioides lutzii* | | *Plasmodium inui* |
| | *Aspergillus zonatus* | | *Penicillium rubens* | | *Plasmodium knowlesi* |
| | *Botrytis cinerea* | | *Phytophthora palmivora* | | *Plasmodium malariae* |
| | *Byssoonygena ceratinophila* | | *Phytophthora plurivora* | | *Plasmodium ovale curtisi* |
| | *Candida auris* | | *Puccinia triticina* | | *Plasmodium praefalciparum* |
| | *Candida duobushaemulonis* | | *Phytopythium vexans* | | *Plasmodium relictum* |
| | *Candida haemulonis* | | *Rhizophagus irregularis* | | *Plasmodium vinckei* |
| | *Candida parapsilosis* | | *Scedosporium apiospermum* | Toxo | *Cystoisospora suis* |
| | *Candida tropicalis* | | *Sporothrix brasiliensis* | Trich | *Blechomonas ayalai* |
| | *Cenococcum geophilum* | | *Sporothrix schenckii* | | *Bodo saltans* |
| | *Chrysosporium queenslandicum* | | *Trichoderma virens* | | *Leptomonas seymouri* |
| | *Cladophialophora carrionii* | | *Uncinocarpus reesii* | | *Paratrypanosoma confusum* |
| | *Cladophialophora immunda* | | *Zymoseptoria tritici* | | *Trypanosoma theileri* |

sequences, in accordance with existing literature on the prevalence of low complexity regions in *Plasmodium* genomes [58, 59].

In addition to the low-complexity sequences, vertebrate, bacterial, viral, plant, vector, and human sequences were detected across all *Plasmodium* genomes. For additional insight into these contaminating sequences, I analyzed the *Plasmodium* pseudo-read classifications (**Figure 3.8**). Notably, the masked pseudo-reads indicated that host DNA continues to be present in these draft genomes as discovered previously; chicken pseudo-reads were mainly found in the poultry malarial parasite *Plasmodium galli-*

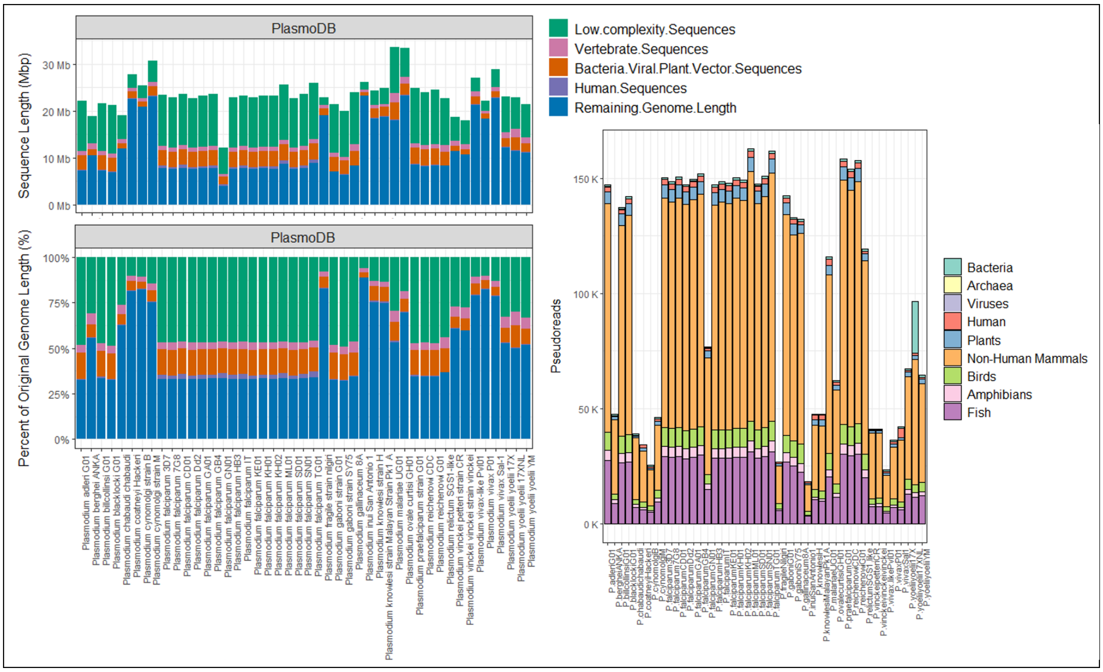


Figure 3.8: **Pseudo-read Classifications in EuPathDB-46 *Plasmodium* genomes.** The top left panel displays the lengths of genome sequences masked in each masking step (human sequences in purple, bacterial/viral/plant/vector sequences in orange, vertebrate sequences in pink, and low complexity sequences in green) along with the length of the remaining genome sequence length. The bottom left panel shows the percentage of the original sequence length removed in each step and the percentage of genome sequence length remaining. The plot on the right displays the pseudo-reads classifications of each of the *Plasmodium* genomes.

*naceum* [55], mouse pseudo-reads found in the mouse malarial parasite *Plasmodium yoellii*, and primate pseudo-reads found in the human/chimpanzee malarial parasite *Plasmodium vivax* [56].

The ToxoDB genomes in EuPathDB-46 also exhibited higher levels of contamination as compared to the remaining eukaryotic pathogen genomes. **Figure 3.9** shows the breakdown of the pseudo-read classifications for all ToxoDB genomes. Specifically, the 8 *Eimeria* genomes revealed high levels of contamination, resulting in approximately

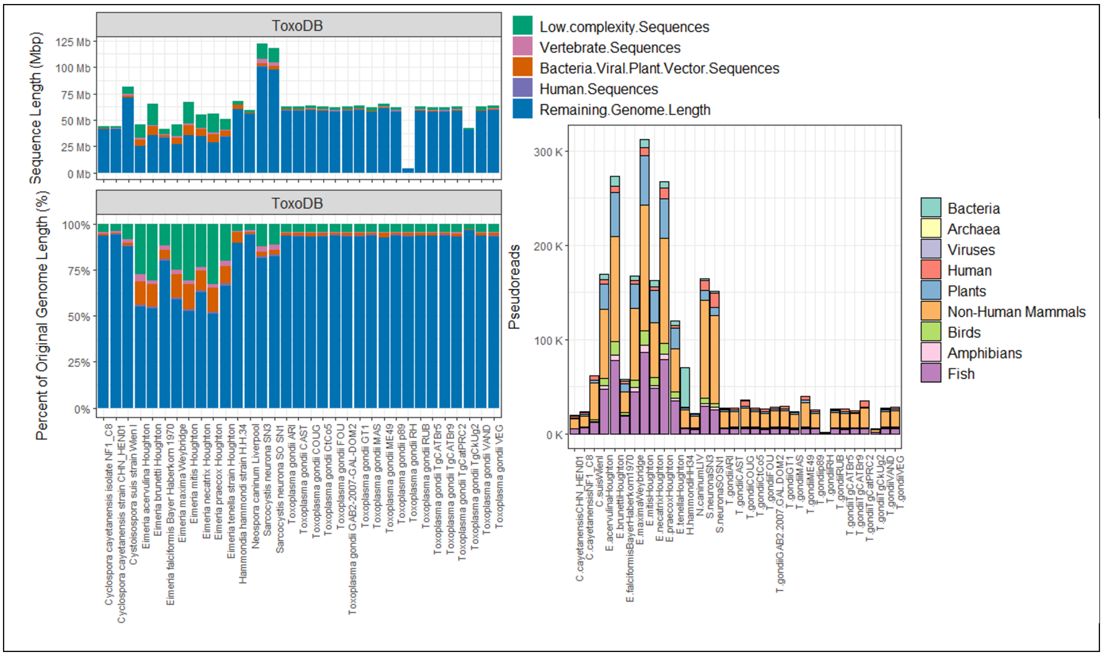


Figure 3.9: **Pseudo-read Classifications in EuPathDB-46 *Toxoplasma* genomes.** The top left panel displays the lengths of genome sequences masked in each masking step (human sequences in purple, bacterial/viral/plant/vector sequences in orange, vertebrate sequences in pink, and low complexity sequences in green) along with the length of the remaining genome sequence length. The bottom left panel shows the percentage of the original sequence length removed in each step and the percentage of genome sequence length remaining. The plot on the right displays the pseudo-reads classifications of each of the *Toxoplasma* genomes.

40% of each *Eimeria* genome being masked, mainly due to host DNA. The *Eimeria* genera, while not infectious to humans, is closely related to the human intestinal pathogen *Cyclospora* [60] which is also represented in the ToxoDB sub-database. The *Eimeria* organisms instead infect a variety of various animals including fish, poultry, rodents, and bats. [61]. For example, *Eimeria mitis* is a known infectious agent of chickens [62] and in our analysis, more than 1 Mb of the *Eimeria mitis* genome was masked due to approximately 14,500 of its pseudo-reads classified as bird DNA.

The *Eimeria* genomes were also significantly contaminated with fish DNA, as several *Eimeria* species are known fish parasites [63].

### 3.5.1 Human Cornea Samples vs. EuPathDB-46

As previously described, I evaluated the effectiveness of the contamination removal process by classifying the human cornea samples from Li et. al 2018 [57] against EuPathDB-46. Details about these samples are listed in **Table 3.4**. For a thorough investigation into the importance of each masking step, I generated multiple Kraken databases from the eukaryotic pathogen genomes at each step of the cleaning process. I then classified the non-human reads from each of the corneal samples against each of the Kraken databases.

**Figure 3.10** summarizes the change in classified reads for the top 15 species originally identified in the corneal samples when using the original EuPathDB-46 genomes. **Figure 3.10A** shows how *Toxoplasma gondii* and *Naegleria fowleri* are found across all 20 corneal samples, despite each sample being infected by different pathogens. Additionally, the distribution of the 15 species is identical across each sample. After removing bacterial, viral, and archaeal DNA from EuPathDB-46, **Figure 3.10B** reveals significantly lower levels of *Toxoplasma gondii* across all samples. Additionally, the removal of these contamination sources also revealed clear true positive signals for *Anncaliia algerae* in Case 10 and *Candida albicans* in Case 7. Removal of non-human vertebrate DNA from the EuPathDB-46 genomes further reduced the *Toxoplasma gondii* signal in all samples and reduced the number of *Plasmodium falciparum* reads in the Case 1 corneal sample (**Figure 3.10C**). Finally, additional removal of human DNA resulted in a much cleaner EuPathDB-46 (**Figure 3.10D**), significantly reduc-

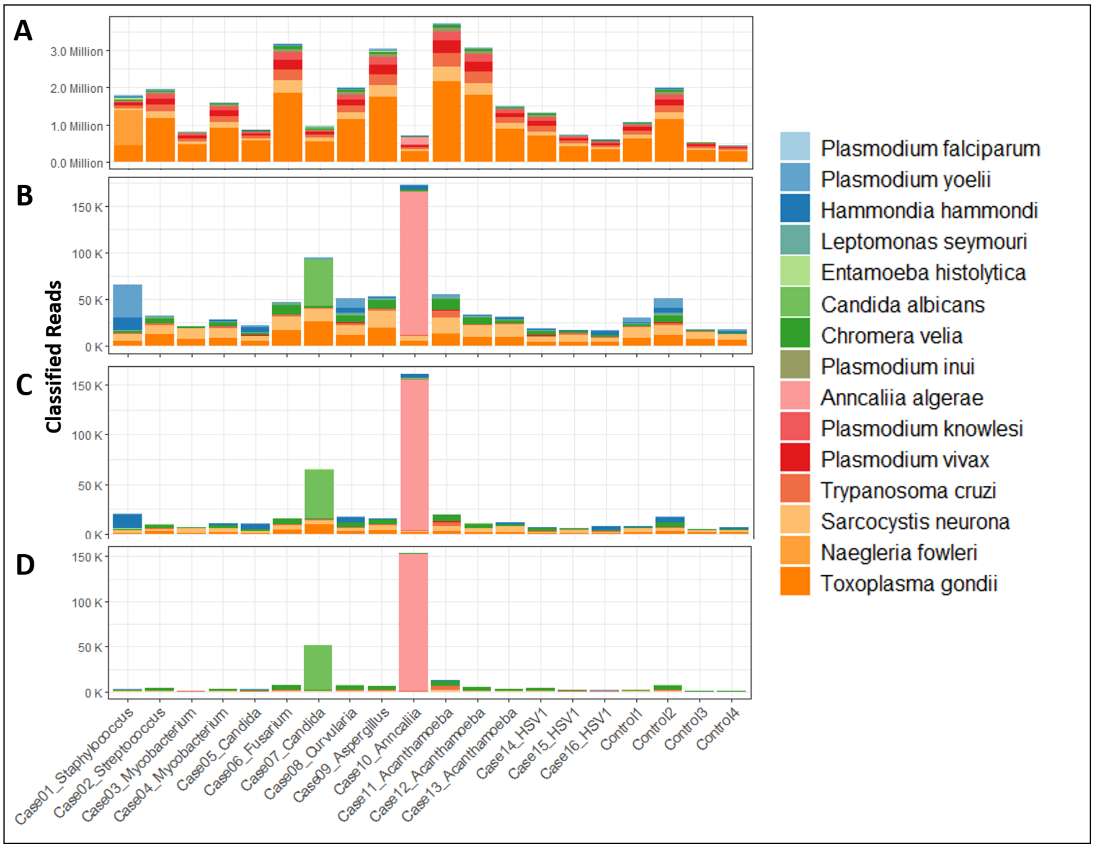


Figure 3.10: **Top 15 species identified in corneal samples when classified with the original EuPathDB-46 database.** The non-human reads from the 20 corneal samples were classified against four Kraken databases: the original EuPathDB **(A)**, EuPathDB without bacterial/archaeal/viral/plant contamination **(B)**, EuPathDB with additional removal of vertebrate contamination **(C)** and the final EuPathDB-46 with additional removal of human contamination **(D)**. The plot above focuses on the 15 species with the most classified reads when classifying the corneal samples against the original EuPathDB, comparing the number of classified corneal reads for each database.

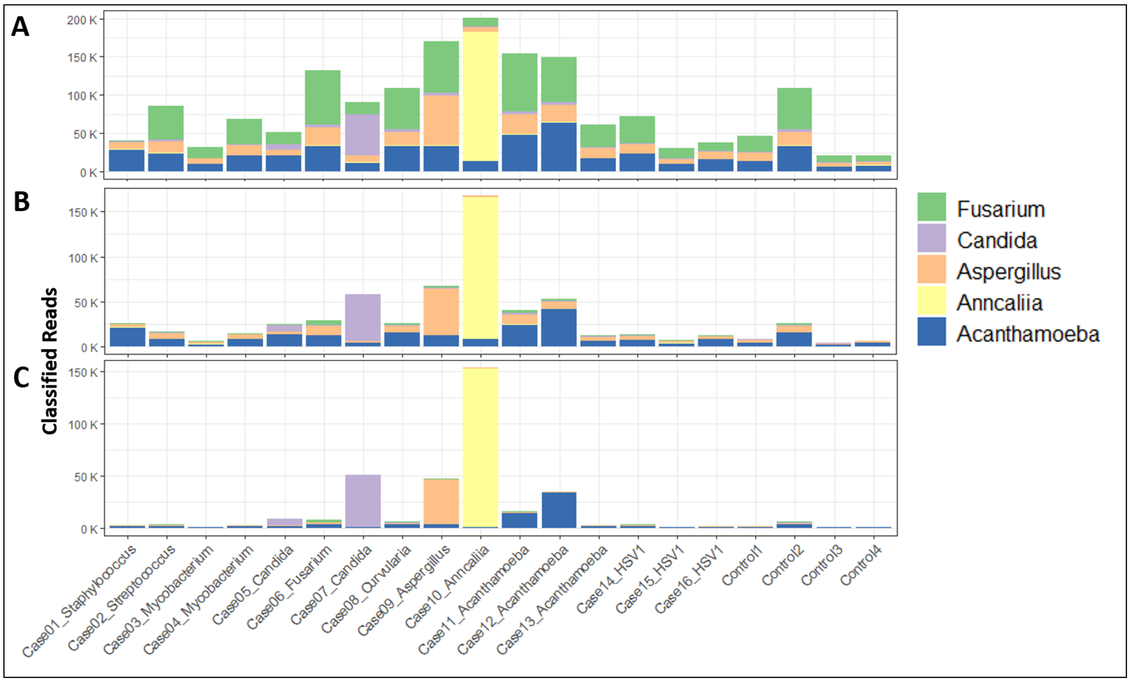


Figure 3.11: **Classified reads for the true positive genera in the corneal samples for the EuPathDB-46 databases.** The above plot compares the number of classified reads for the true eukaryotic pathogens in the infected samples when classifying the samples against: the original EuPathDB **(A)**, EuPathDB without low-complexity sequences **(B)**, and the final EuPathDB-46 with removal of human/bacterial/archaeal/viral/plant/vertebrate contamination **(C)**. The true pathogens are *Fusarium* (Case 6), *Candida* (Case 7), *Aspergilllus* (Case 9), *Anncaliia* (Case 10), and *Acanthamoeba* (Cases 11-13).

ing the number of false positive *Hammondia hammondi* reads across all samples.

I then compared the read counts across databases for the true positives in the corneal samples: *Fusarium* (Case 6), *Candida* (Case 7), *Aspergillus* (Case 9), *Anncaliia* (Case 10), and *Acanthamoeba* (Cases 11-13). Notably, *Curvularia* in Case 8 is also a eukaryotic pathogen, but the *Curvularia* genome does not exist in EuPathDB-46. **Figure 3.11A** shows the read counts for these genera across all samples when classifying the corneal samples against the original EuPathDB-46. As with EuPathDB-28, the original database limits the visibility of the truly infected samples, with signals for all five eukaryotic pathogens showing up in all 20 corneal samples, including the 4 control samples. However, after low-complexity sequences are removed, the resulting EuPathDB-46 already shows improvement, with the number of *Fusarium* reads reduced across all samples (**Figure 3.11B**). The remaining cleaning steps yielded a final EuPathDB-46 that allows for correct identification of the true positive genera in their respective samples.

## 3.6 Conclusion

In principle, next-generation sequencing can identify all microbial organisms within any sample, making it a potentially a revolutionary method for the diagnosis of human infections. However, this method relies heavily on the computational analysis that compares sequencing reads against reference databases, such as RefSeq and GenBank. Although new genomes are being sequenced daily, the reference databases remain incomplete and, because most new genomes are in draft form, inaccurate. Recent studies have identified contamination in many published genomes, hindering our ability to use them for accurate diagnosis.

I therefore developed a comprehensive contamination removal process, identifying human, vertebrate, bacterial, viral, archaeal, and vector contamination in the 245 eukaryotic pathogen draft genomes of EuPathDB-28 and then again in the 388 genomes of EuPathDB-46. By removing contamination and low-complexity sequences, I have created a much cleaner database that minimizes false positives and provides better identification of true positive pathogens in NGS diagnostic samples.

# Chapter 4

# Bracken: Bayesian Reestimation of Abundance after Classification with KrakEN

**Portions of this chapter originally appeared in:**

J. Lu, F. P. Breitwieser, P. Thielen, S. L. Salzberg, (2017). "Bracken: estimating species abundance in metagenomics data." *PeerJ Computer Science*, 3:e104 `https://doi.org/10.7717/peerj-cs.104`

**Related Software:** `https://github.com/jenniferlu717/Bracken/`

## 4.1 Introduction

When it was first published in 2014, the Kraken metagenomics classifier provided an extremely fast and accurate method for classifying sequencing reads by comparing exact-match kmers [12]. As compared to existing tools, the Kraken classifier provided a major enhancement in speed for analyzing large metagenomics sequencing data, running over 900 times faster than MegaBlast [54], the closest competitor at the time. Kraken's success and accuracy rely on its use of a very large, efficient index of short sequences of length $k$, which it builds into a specialized database. If $k$ is chosen appropriately, then most sequences of length $k$ in the database will be unique to a single species, and many will also be unique to a particular strain or genome. Larger values of $k$ will yield a database in which even more of each genome is uniquely covered by $k$-mers; obviously, though, $k$ should not be longer than the length of a sequencing read, and metagenomics projects currently generate reads as short as 75–100 base pairs (bp). Longer $k$-mers are also more likely to contain errors, meaning that more reads will be left unclassified if k is too long. Smaller $k$-mers, in contrast, will yield higher sensitivity because the minimum match length is shorter.

When used to identify the taxonomic label of metagenomics sequences, the Kraken system for classification of metagenomics sequences is extremely fast and accurate [12]. When classifying raw sequence reads, though, many reads correspond to identical regions between two or more genomes. (The number of such ambiguous reads decreases as reads get longer.) Kraken solves this problem by labeling the sequence with the lowest common ancestor (LCA) of all species that share that sequence, as discussed further below.

### 4.1.1 Ambiguity among microbial species and strains

As the database of bacterial genomes has grown, an increasing number of genomes share large portions of their sequence with other genomes. In many cases, these genomes are nearly identical; indeed, sequencing has revealed to scientists that many formerly distinct species and genera are far closer than were known prior to sequencing. Many species have been renamed as a result, in a process that is continual and ongoing, but many other species have retained their old names, often for historical or other reasons.

For example, the species Mycobacterium bovis is over 99.95% identical to *Mycobacterium tuberculosis* [64], and many cases of human tuberculosis are caused by *M. bovis* (which also infects cows) rather than *M. tuberculosis* [65]. Their high sequence identity indicates that they should be considered as two strains of a single species, but they retain different species names. As a compromise, taxonomists have created the category *Mycobacterium tuberculosis complex* [66] to represent a collection of taxa that now includes more than 100 strains of five different species. This category sits above the species level but below the genus level in the current microbial taxonomy, but it can best be described as a species.

Other examples are numerous and still growing. The three species *Bacillus anthracis* (the causative agent of anthrax), *Bacillus cereus*, and *Bacillus thuringiensis* are well over 99% identical and should all be designated as a single species [67], although their names have not been changed despite their near-identity revealed by sequencing. As a compromise, taxonomists created the category *Bacillus cereus group*, between the level of species and genus, to include these three species and at least five others [68], all of which are extremely similar to one another. In some cases, two organisms that should be called the same species may even have different genus names. For example, *Escherichia coli* and *Shigella flexneri* are classified in different genera, but we know from sequence analysis that they represent the same species [69].

Failure to recognize the mutability of the bacterial taxonomy can lead to erroneous conclusions about the performance of metagenomic classifiers. For example, one recent study [70] created a mock community of 11 species, one of which was *Anabaena variabilis* ATCC 29413, not realizing that this genome had been renamed and was synonymous with species in the genus *Nostoc* [71]. When *Anabaena* was removed from the database, Kraken correctly identified the reads as *Nostoc*, but Peabody et al. erroneously considered all these reads to be misclassified.

### 4.1.2 Classification versus abundance estimation

Kraken attempts to assign a taxonomy label to every read in a metagenomics sample using a custom-built database that may contain any species the user chooses. Among the current set of finished bacterial and archaeal genomes, hundreds of species can be found for which large fractions of their sequence are identical to other genomes belonging to distinct strains, species, or even genera. The reads arising from common regions in these species result in a tie when analyzed with Kraken's classification algorithm, so Kraken correctly reports only the lowest common ancestor (LCA) [12]. It follows that for well-populated clades with low genome diversity, Kraken only reports species-level assignments for reads from unique regions, and a true indication of total abundance can only be made by taking both species and genus (or higher) level assignments into account. This implies that for some species, the majority of reads might be classified at a higher level of the taxonomy. Kraken thus leaves many reads "stranded" above the species level, meaning that the number of reads classified directly to a species may be far lower than the actual number present.

Therefore, any assumption that Kraken's raw read assignments can be directly translated into species- or strain-level abundance estimates (e.g., [72]) is flawed, as ignoring reads at higher levels of the taxonomy will grossly underestimate some species, and creates the erroneous impression that Kraken's assignments themselves were incorrect.

Nonetheless, metagenomics analysis often involves estimating the abundance of the species in a particular sample. Although Kraken cannot unambiguously assign each read to a species, I sought to estimate how much of each species is present, specifically by estimating the number or percentage of reads in the sample. Several software tools have been developed to estimate species abundances in metagenomics samples [MetaPhlAn, ConStrains, GAAS, GASiC, TAEC, GRAMMy] [73–78]. These tools, however, employ different strategies for read-level classification which are not always as accurate and efficient as Kraken's k-mer approach [79]. Rather than re-engineer Kraken to address the ambiguous read classification issue and to provide abundance estimates directly, I implemented the new species-level abundance estimation method described here as a separate program. This preserves both backwards compatibility for existing Kraken users, and offers the ability to generate more accurate species abundance estimates for datasets already processed by Kraken. Note that if Kraken fails to identify a species (e.g., if the species was missing from the Kraken database), Bracken too will not identify that species.

## 4.2 Materials and Methods

My new method, Bracken (Bayesian Reestimation of Abundance after Classification with KrakEN), estimates species abundances in metagenomics samples by probabilistically re-distributing reads in the taxonomic tree. Reads assigned to nodes above

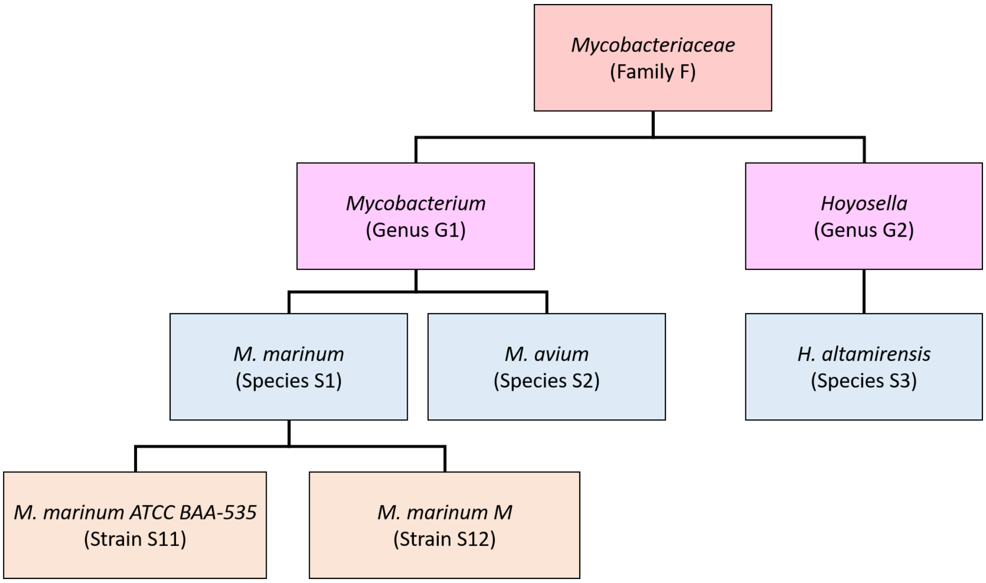


Figure 4.1: **Schematic showing a partial taxonomic tree for the Mycobacteriaceae family.**

the species level are distributed down to the species nodes, while reads assigned at the strain level are re-distributed upward to their parent species. For example, in **Figure 4.1** we would distribute reads assigned to the *Mycobacteriaceae* family and the Mycobacterium genus down to *M. marinum* and *M. avium*, and reads assigned to each *M. marinum* strain would be reassigned to the *M. marinum* species. As I show below, Bracken can easily reestimate abundances at other taxonomic levels (e.g., genus or phylum) using the same algorithm.

In order to re-assign reads classified at higher-level nodes in the taxonomy, I need to compute a probabilistic estimate of the number of reads that should be distributed to the species below that node. To illustrate using the nodes in **Figure 4.1**, I need to allocate all reads assigned to *Mycobacterium* (G1) to *M. marinum* (S1) and *M.*

*avium* (S2) below it, and reads assigned to the *Mycobacteriaceae* would have to be allocated to M. marinum (S1), M. avium (S2), and *Hoyosella altamirensis* (S3).

Reallocating reads from a genus-level node in the taxonomy to each genome below it can be accomplished using Bayes' theorem, if the appropriate probabilities can be computed. Let $P(S_i)$ be the probability that a read in the sample belongs to genome $S_i$, $P(G_j)$ be the probability that a read is classified by Kraken at the genus level $G_j$, and $P(G_j|S_i)$ be the probability that a read from genome $S_i$ is classified by Kraken as the parent genus $G_j$. Then the probability that a read classified at genus $G_j$ belongs to the genome $S_i$ can be expressed as Eq.4.1:

$$P(S_i|G_j) = \frac{P(G_j|S_i)P(S_i)}{P(G_j)} \tag{4.1}$$

Note that because I began by assuming that a read was classified at node $G_j$, $P(G_j) = 1$.

Next I consider how to compute $P(G_j|S_i)$, the probability that a read from genome $S_i$ will be classified by Kraken at the parent genus $G_j$. I estimate this probability for reads of length $r$ by classifying the sequences (genomes) that we used to build the database using that same database, as follows. For each $k$-mer in the sequences, Kraken assigns it a taxonomy ID by a fast lookup in its database. To assign a taxonomy ID for a read of length $r$, Kraken examines all $k$-mer classifications in that read. For example, for $k = 31$ and $r = 75$, the read will contain 45 $k$-mers. My procedure examines, for each genome in the database, a sliding window of length $r$ across the entire genome.

To find the taxonomy ID Kraken would assign to each window, I simply find the deepest taxonomy node in the set of $k$-mers in that window. Since each $k$-mer in a database sequence is assigned to a taxonomy ID somewhere along the path from the genome's taxonomy ID to the root, the highest-weighted root-to-leaf path (and thus the Kraken classification) corresponds to the deepest node.

For each genome $S_i$ of length $L_i$ I thus generate $(L_i - r + 1)$ mappings to taxonomical IDs. For node $G_j$, I then count the number of reads from $S_i$ that are assigned to it, $N_{G_j(S_i)}$. $P(G_j|S_i)$ is then the proportion of reads from $S_i$ that were assigned to the genus node $G_j$; i.e., $P(G_j|S_i) = N_{G_j(S_i)}/(L_i - r + 1)$. I also calculate the proportion of reads from $S_i$ that were assigned to every node from genome Si to the root node of the taxonomy tree.

The final term that I must calculate from Eq. 4.1 is $P(S_i)$, the probability that a read in the sample belongs to genome $S_i$, which is computed in relation to other genomes from the same genus. For example, if the sample contains three genomes in the same genus, and if 30% of all reads from those three genomes belong to $S_i$, then $P(S_i) = 0.3$. I estimate this probability using the reads that are uniquely assigned by Kraken to genome $S_i$, as follows.

If I let $U_{S_i}$ be the proportion of genome Si that is unique, then

$$U_{S_i} = \frac{N_{S_i}}{L_i - r + 1} \tag{4.2}$$

where $N_{S_i}$ is the number of $k$-mers of length $r$ that are uniquely assigned to genome $S_i$ by Kraken, and $L_i$ is the genome length. For example, if $L_i = 1$ Mbp and only $250,000$ $k$-mers are unique to genome $S_i$, then $U_{S_i} = 0.25$.

Then, using the number of reads $K_{S_i}$ from a sample that Kraken actually assigns to $S_i$, I can estimate the number of reads that likely derive from $S_i$ as:

$$\hat{K}_{S_i} = \frac{K_{S_i}}{U_{S_i}} \tag{4.3}$$

For example, if Kraken classifies $1,000$ reads as genome $S_i$ and 25% of the reads from $S_i$ are unique, then I would estimate that $4,000$ reads $(1,000/0.25)$ from

If genus $G_j$ contains $n$ genomes, I estimate the number of reads $\hat{K}_S$ for each of the $n$ genomes and then calculate $P(S_i)$ by:

$$P(S_i) = \frac{\hat{K}_{S_i}}{\sum_{a=1}^{n} \hat{K}_{S_a}} \tag{4.4}$$

Using this result in Eq.4.1 above allows me to compute $P(S_i|G_j)$ for each genome $S_i$. Each probability $P(S_i|G_j)$ is then used to estimate the proportion of the reads assigned to genus $G_j$ that belong to each of the genomes below it.

These calculations are repeated for each taxonomic level above the genus level (family, class, etc.), with read distribution at each level going to all genomes classified within that taxonomic subtree.

To compute species abundance, any genome-level (strain-level) reads are simply added together at the species level. In cases where only one genome from a given species is detected by Kraken in the dataset, I simply add the reads distributed downward from the genus level (and above) to the reads already assigned by Kraken to the species level. In cases where multiple genomes exist for a given species, the reads distributed to each genome are combined and added to the Kraken-assigned species level reads. The added reads give the final species-level abundance estimates.

This method can also estimate abundance for other taxonomic levels. In such cases, only higher nodes within the taxonomy tree undergo read distribution. After distributing reads downward, I estimate abundance for a node at the level specified by combining the distributed reads across all genomes within that node's subtree.

### 4.2.1 Software and data availability

Bracken is written in Perl and Python and is freely available for download at `http://ccb.jhu.edu/software/bracken/`. The reads from the skin microbiome experiment are freely available from NCBI under BioProject PRJNA316735.

## 4.3 Kraken/Bracken Results and Discussion

I applied the statistical re-assignment method described here to create species-level abundance estimates for several metagenomics data sets. The overall procedure works as follows. First, I compute a set of probabilities from the Kraken database by computing, for every sequence of length R in every genome, where it will be assigned in the taxonomy (see 'Methods'). For my experiments, I set $R = 75$ as our datasets contain 75-bp reads. Bracken can use these probabilities for any metagenomics data set, including data with different read lengths, although the estimates might be slightly improved by re-computing with a read length that matches the experimental data.

Second, I run Kraken on the dataset to produce read-level taxonomic classifications. I then apply our abundance estimator, Bracken, which uses the numbers of reads assigned by Kraken at every level of the taxonomy to estimate the abundances at a single level (e.g., species). Note that to exclude × positives, Bracken ignores species with counts below a user-adjustable threshold. In my experiments, I selected a threshold of 10 reads.

### 4.3.1 Experiments on a 100-genome metagenomics data set

For my first experiments, I used a data set containing simulated Illumina reads from 100 genomes. This data, which I call here the i100 dataset, was used previously in a comparison of metagenomic assembly algorithms [80]. The data contains 53.3 million paired reads (26.7M pairs) from 100 genomes representing 85 species. The reads have error profiles based on quality values found in real Illumina reads [80]. The i100 dataset includes several very challenging genomes for this task, including multiple strains and species in the genera *Bacillus* and *Mycobacteria*, some of which are nearly identical to one another. The i100 data are freely available at `http://www.bork.embl.de/~mende/simulated_data`.

The difficulty of estimating species abundance increases as the database itself contains more species. For example, it would clearly be easier to estimate abundances in the i100 dataset if I used a Kraken database containing only the 100 genomes in that dataset. To make the problem more realistic, I built two different databases and

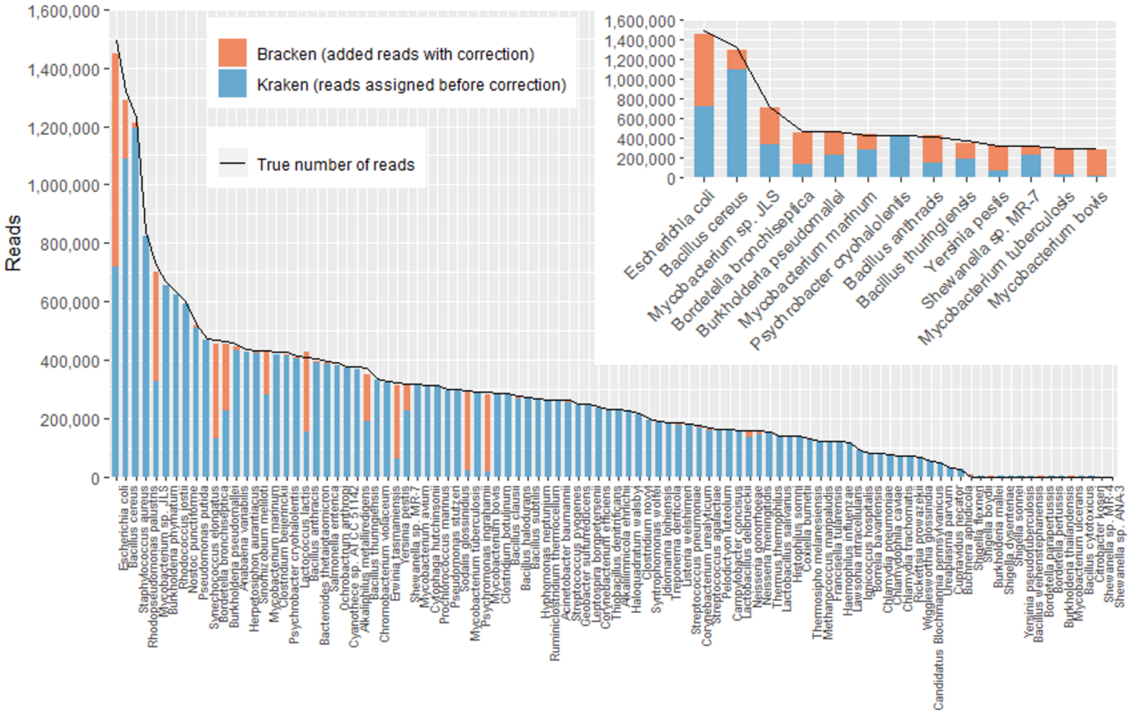


Figure 4.2: **Estimates of species abundance in the i100 metagenomics dataset computed by Kraken (blue) and Bracken (blue + orange).** For this result, the Kraken database contained 693 genomes that included the i100 genomes. The smaller graph displays results for the subset of species for which Bracken made the largest adjustments. The black line shows the true number of reads from each species.

estimated abundance using both. The first (“small”) database contains 693 genomes including the i100 genomes; this is the full database from the simulation study by Mende et al. [80]. The results when using the small database for classification are shown in **Figure 4.2**. For several species, the initial Kraken numbers (reads assigned to a particular species) are far too low, because many of the reads (for some genomes, a large majority) were assigned labels at the genus level or above. After reestimation with Bracken, these reads were redistributed to the species level, with the result that

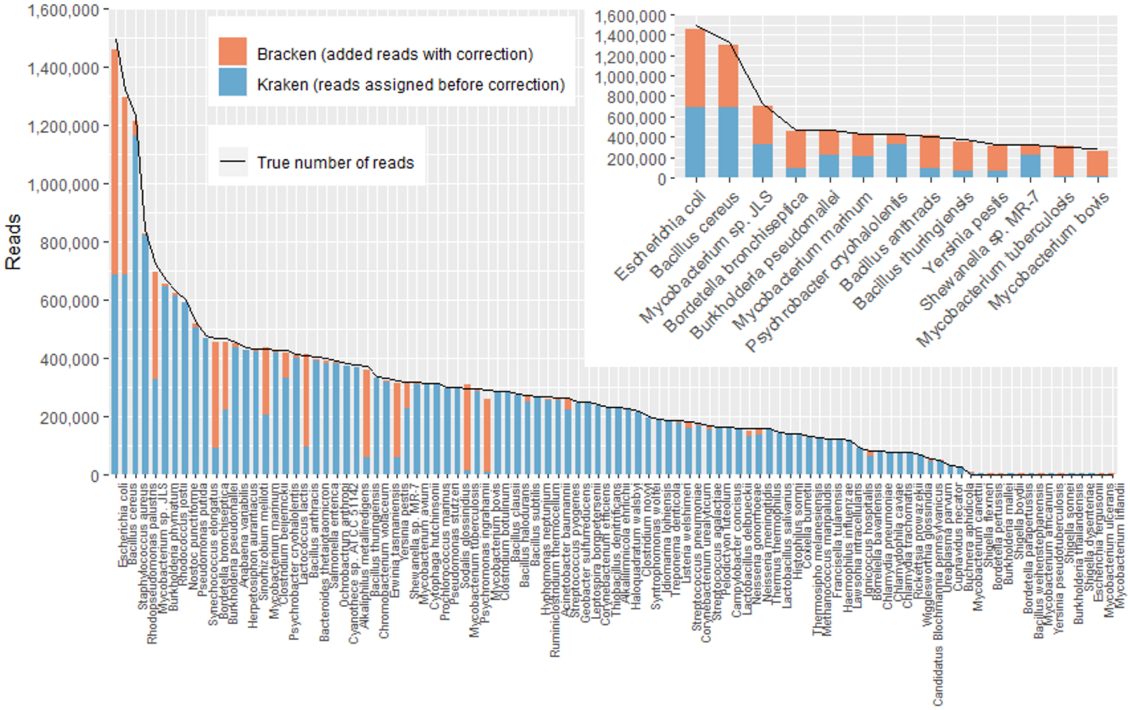


Figure 4.3: **Estimates of species abundance computed by Kraken (blue) and Bracken (blue + orange) for the i100 metagenomics data.** For this result, the Kraken database contained 2,635 distinct bacterial and archaeal taxa. The black line shows the true number of reads from each species. The smaller graph displays results for the subset of species for which Bracken made the largest adjustments.

almost all the abundance estimates were 98–99% correct, as shown in the figure.

The second ("large") database contains all genomes used in the synthetic and spike-in experiments, as well as a broad background of bacterial genomes. In particular, it includes all complete bacterial and archaeal genomes from RefSeq as of 25 July 2014 (archived at `ftp://ftp.ncbi.nlm.nih.gov/genomes/archive/old_refseq`), which total 2596 distinct taxa, plus those i100 genomes that were not present in the RefSeq data. (We excluded draft genomes because they often contain vector sequences or other contaminants.) We also added the nine genomes used in our skin bacteria spike-in experiment (described below) resulting in a total of 2635 distinct taxa. The resulting Kraken database has a size of 74 GB.

**Figure 4.3** shows results when using the large database to estimate abundance for the i100 genomes. This test is much more difficult because of the large number of similar and near-identical genomes in the database. Many more reads are ambiguous, mapping identically to two or more species, which means that Kraken assigns them to the LCA of those species. Nonetheless, Bracken brings the estimated abundance of all species within 4% of the true abundance, and most fall within 1%. Note that when the re-estimation procedure distributes reads from higher nodes in the taxonomy down to multiple species within a single genus, it may over-estimate one species and underestimate its sister species if the re-allocation is imperfect.

For a quantitative evaluation of Bracken's ability to capture the true species abundances, I evaluated the modified Mean Absolute Percentage Error (MAPE) which compares the difference between the true read counts ($T_g$) for a given genus and the measured read counts ($A_g$) for that same genus.

$$MAPE = \sum_{g=1}^{n} \frac{T_g}{\sum_{g=1}^{n} T_g} \times \frac{|A_g - T_g|}{T_g} \tag{4.5}$$

where $n$ is the number of species in the i100 data. When using the small database, the MAPE of Bracken is 1.90% across all 85 species in the i100 data. For the larger database, the average relative error is 1.97%. I also calculated the false positive rates for the i100 data as the percentage of total reads incorrectly classified after Bracken abundance estimation. For the small database, the false positive rate is 0.13% and for the large database, the false positive rate is 0.24%.

Within the i100 genomes, the five species belonging to the *Mycobacterium* genus (*M. tuberculosis, M. bovis, M. avium, M. marinum*, and *M. sp. JLS*) pose a particular challenge for abundance estimation due to the similarities among their individual genomes. For example, Kraken classified only 9,733 *M. tuberculosis* reads at the species level, and classified the remaining 285,414 reads as either *Mycobacterium* (a genus) or *M. tuberculosis complex* (a taxonomic class intermediate between genus and species), as shown in **Figure 4.4** and **Table 4.1**. For these *Mycobacteria* genomes, Bracken reallocated the reads from higher-level nodes to yield species abundance estimates within 4% of the true abundance. **Figure 4.4** and **Table 4.1** show the number

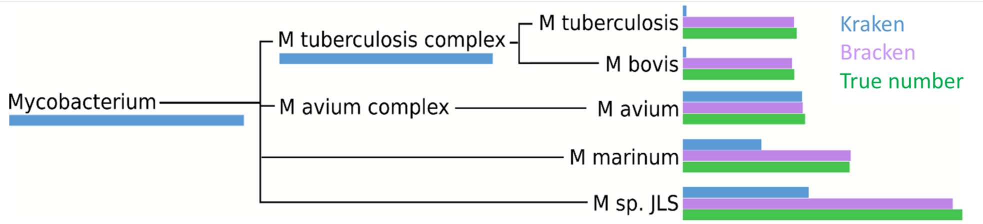


Figure 4.4: **Number of reads within the *Mycobacterium* genus as assigned by Kraken (blue), estimated by Bracken (purple) and compared to the true read counts (green).** Initially, Kraken assigned only 325,073 reads to *Mycobacterium sp. JLS* although 722,880 reads originated from this species. Bracken reassigned 370,601 reads from the *Mycobacterium* genus to *M. sp. JLS*. Bracken's re-estimated abundance for *M. sp. JLS* is much closer to the true read count. **Table 4.1** contains precise numbers for all species shown here.

Table 4.1: **Mycobacterium Bracken Re-estimates**. This table lists the Kraken assigned reads for various *Mycobacterium* species and for the *Mycobacterium* genus and the *Mycobacterium tuberculosis complex* (*MTB complex*). The table demonstrates how Bracken redistributed the reads and compares the final read estimates to the true read counts.

| Name | TaxID | Reads assigned by Kraken | Added from *Mycobacterium* | Added from *MTB complex* | Added from Higher Up |
|---|---|---|---|---|---|
| *M. sp. JLS* | 164757 | 325073 | 370601 | 0 | 519 |
| *M. avium* | 1764 | 308864 | 3652 | 0 | 841 |
| *M. marinum* | 1781 | 203097 | 229780 | 0 | 83 |
| *M. TB* | 1773 | 9733 | 386 | 296463 | 209 |
| *M. bovis* | 1765 | 8965 | 329 | 247453 | 181 |
| *Mycobacterium* | 1763 | 606334 | | | |
| *MTB complex* | 77643 | 550752 | | | |
| Total | | | 604747 | 543916 | |

| Name | TaxID | Reads assigned by Kraken | Total reads added by Bracken | Final Bracken read estimate | True read count |
|---|---|---|---|---|---|
| *M. sp. JLS* | 164757 | 325073 | 371120 | 696193 | 722880 |
| *M. avium* | 1764 | 308864 | 4493 | 313357 | 316134 |
| *M. marinum* | 1781 | 203097 | 229863 | 432960 | 431254 |
| *M. TB* | 1773 | 9733 | 297059 | 306792 | 295147 |
| *M. bovis* | 1765 | 8965 | 247962 | 256927 | 288400 |

of reads assigned to each species by Kraken, the true number of reads, and the number of reads assigned to each species by Bracken after abundance reestimation.

The five species of the *Mycobacterium* genus also provide an example of potential overestimation by Bracken. Bracken apportions all ambiguous reads classified by Kraken at the genus level (and above) to the existing species identified by Kraken. Because Bracken uses a probabilistic method in distributing the reads, one species may receive too many reads while another may receive too few. For example, Kraken assigned 543,916 reads to *M. tuberculosis complex*. Bracken re-allocated 296,543 of these reads to *M. tuberculosis* and the remaining 247,453 reads to *M. bovis.* When added to Kraken's original assignments, Bracken estimated that 306,792 reads belonged to *M. tuberculosis* (11,645 reads more than the true number) that 256,927 reads belonged to *M. bovis* (31,473 reads less than the true number). It is likely that some of the additional reads Bracken allocated to *M. tuberculosis* originated from *M. bovis* instead. However, despite the over- and under-estimation, Bracken's estimates fell within 4% of the true number of reads for both species.

If *M. bovis* were excluded from the database, the 8,965 reads unique to *M. bovis*, as identified by Kraken, would be unclassified, while all 543,916 reads assigned to the *M. tuberculosis complex* would assigned to M. tuberculosis by Kraken. These reads would no longer be ambiguous because no other *Mycobacterium* species from the *M. tuberculosis complex* would be present in the database. In general, reads belonging to species excluded from the database will either be assigned to species with very high similarity to the missing species or will remain unclassified.

### 4.3.2 Experiments on a real metagenomics sample created from known species

For a more realistic test, I evaluated the performance of Bracken using a mock community of bacteria commonly found on healthy human skin. This mock community was assembled by **Peter Thielen** by combining purified DNA from nine isolates that were identified and sequenced during the initial phase of the Human Microbiome Project [81]: *Acinetobacter radioresistens* strain SK82, *Corynebacterium amycolatum* strain SK46, *Micrococcus luteus* strain SK58, *Rhodococcus erythropolis* strain SK121, *Staphylococcus capitis* strain SK14, *Staphylococcus epidermidis* strain SK135, *Staphylococcus hominis* strain SK119, *Staphylococcus warneri* strain SK66, and *Propionibacterium acnes* strain SK137. To generate the skin microbiome community, purified DNA was obtained from the Biodefense and Emerging Infections Research Resources Repository (BEI Resources). Each of the nine bacterial isolates was grown under conditions recommended by BEI Resources, collected by centrifugation during log growth phase at a 600nm optical density (OD600) of 0.8–1.2, and genomic DNA was isolated using MasterPure DNA isolation reagents (Epicentre). Purified genomic DNA was quantified using the high sensitivity picogreen assay (Invitrogen), pooled in equal amounts by mass, and prepared for sequencing using Nextera XT library preparation reagents (Illumina). The resulting mock community was sequenced on a HiSeq sequencer, generating a total of 78,439,985 million read pairs (157 million reads), all of them 100 bp in length. I then classified the sample using Kraken, which concatenates the two reads from each pair and assigns them to a single taxonomic category.

I then used Bracken to estimate both species and genus-level abundance in the skin microbiome community. In the Bracken results, the nine true species comprise over

99% of the species-level abundance estimates. The mixture was created with approximately equal amounts of each of the nine genomes, so the expectation was that each species would account for $\sim 11\%$ of the total. However, as shown in **Figure 4.5**, the estimates varied from 7.3% to 14.8%. Details for the exact number of reads assigned by Kraken and the abundance estimates by Bracken are shown in **Table 4.2**.

Deviations from the expected abundance could arise from a variety of factors. The process of quantifying DNA and mixing in equal amounts can be influenced by pipetting consistency. Second, library amplification by PCR, an integral step in the Nextera library preparation process, can exaggerate small differences in quantities and lead to significant biases in abundance [82]. I examined a sample of the classified reads by hand, and could find no evidence that Kraken mis-classified reads from *M.*

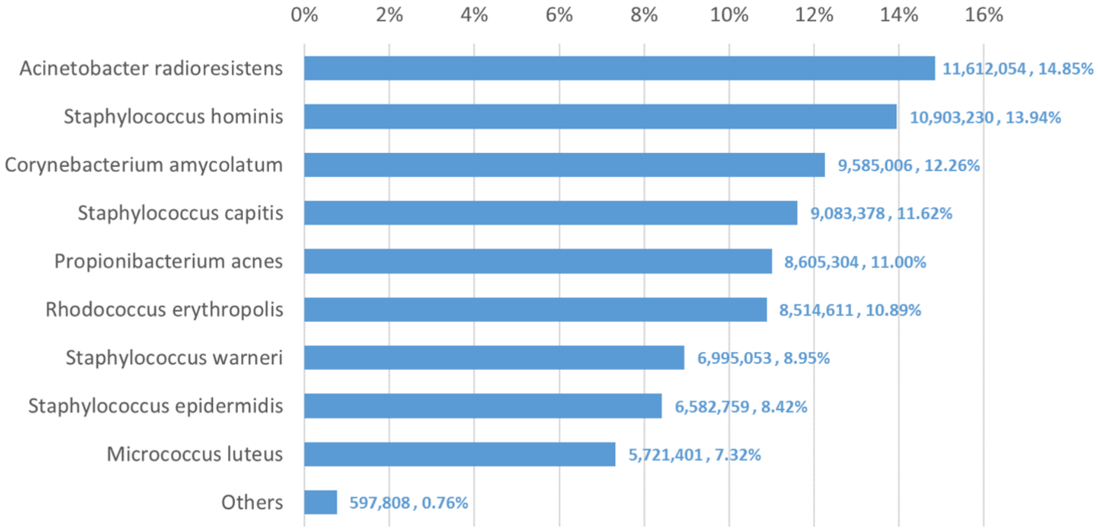


Figure 4.5: **Estimates of species abundance made by Bracken for the metagenomics community containing isolates of nine bacterial species commonly found on human skin.** Exact numbers are listed in **Table 4.2**

Table 4.2: **Bracken results when classifying the data from the skin microbiome community using the large Kraken database**

| Species | True Positive | Kraken Reads | Bracken | | |
|---|---|---|---|---|---|
| | | | Added | Final Reads | Fraction |
| *Acinetobacter radioresistens* | ✓ | 11,574,480 | 37,574 | 11,612,054 | 0.148 |
| *Staphylococcus hominis* | ✓ | 10,675,573 | 227,657 | 10,903,230 | 0.139 |
| *Corynebacterium amycolatum* | ✓ | 9,566,003 | 19,003 | 9,585,006 | 0.122 |
| *Staphylococcus capitis* | ✓ | 8,992,784 | 90,594 | 9,083,378 | 0.116 |
| *Propionibacterium acnes* | ✓ | 8,585,085 | 20,219 | 8,605,304 | 0.110 |
| *Rhodococcus erythropolis* | ✓ | 8,493,987 | 20,624 | 8,514,611 | 0.109 |
| *Staphylococcus warneri* | ✓ | 6,577,061 | 417,992 | 6,995,053 | 0.089 |
| *Staphylococcus epidermidis* | ✓ | 6,271,832 | 310,927 | 6,582,759 | 0.084 |
| *Micrococcus luteus* | ✓ | 5,678,927 | 42,474 | 5,721,401 | 0.073 |
| synthetic construct | × | 359,702 | 25,301 | 385,003 | 0.005 |
| *Staphylococcus aureus* | × | 25,611 | 59,955 | 85,566 | 0.001 |
| *Staphylococcus haemolyticus* | × | 29,658 | 1,237 | 30,895 | 0.00 |
| *Acinetobacter baumannii* | × | 26,939 | 1,103 | 28,042 | 0.000 |
| *Staphylococcus pasteuri* | × | 20,643 | 3,596 | 24,239 | 0.000 |

| Genus | True Positive | Kraken Reads | Bracken | | |
|---|---|---|---|---|---|
| | | | Added | Final Reads | Fraction |
| *Staphylococcus* | ✓ | 33,646,053 | 88,782 | 33,734,835 | 0.433 |
| *Acinetobacter* | ✓ | 11,635,903 | 4,426 | 11,640,329 | 0.150 |
| *Corynebacterium* | ✓ | 9,600,993 | 6,870 | 9,607,863 | 0.123 |
| *Propionibacterium* | ✓ | 8,601,649 | 3,739 | 8,605,388 | 0.110 |
| *Rhodococcus* | ✓ | 8,500,082 | 15786 | 8,515,868 | 0.109 |
| *Micrococcus* | ✓ | 5,678,927 | 48439 | 5,727,366 | 0.074 |
| *Delftia* | × | 1,655 | 25 | 1,680 | 0.000 |
| *Lactobacillus* | × | 1,604 | 12 | 1,616 | 0.000 |
| *Bacillus* | × | 1,036 | 67 | 1,103 | 0.000 |
| *Candidatus Hamiltonella* | × | 1,016 | 19 | 1,035 | 0.000 |
| *Enterococcus* | × | 993 | 18 | 1,011 | 0.000 |
| *Listeria* | × | 900 | 4 | 904 | 0.000 |
| *Mycoplasma* | × | 661 | 5 | 666 | 0.000 |

*luteus* (the smallest portion of the community, estimated at 7.3%) to any of the other species or genera. The abundances found in this data, therefore, may correspond fairly closely with the true abundances.

The genus-level abundance estimates computed by Bracken also correspond closely to the expected abundances for the six genera included in the sample. Four of the nine species belong to the genus *Staphylococcus*, which was thus expected to comprise 44% (4 x 11%) of the sample. The Bracken estimate was 43.3%. Each of the other genus classifications has only one species present, and their abundance estimates are the same for both genus and species.

The comparison between the Kraken classification of reads and Bracken's reassignment revealed that the nine species are sufficiently distinct to allow Kraken to classify a large majority of reads at the species level, with very few reads being classified at higher levels of the taxonomy. Specifically, Kraken classified 76.4 million reads to the nine species included in the sample. Only 1.3 million reads out of the 78.2 million total (1.6%) were classified by Kraken at the genus level or above. (The remaining reads were unclassified.) In this case Bracken does not provide a substantial benefit, because reassignment of the 1.3 million reads could yield at most a 1.6% change in the estimated composition of the sample.

### 4.3.3 Bracken timing and resource requirements

Execution of Bracken requires two main steps 1) building of the Kraken/Bracken database followed by 2) running a sample through Kraken and Bracken. In the initial i100 data experiment with the large database, the Kraken build time with 10 threads required 7 hours and 22 minutes, using 94.1 gigabytes (GB) of RAM and generating a database requiring 70.6 GB of space. The subsequent Bracken build took ¡ 45 minutes, using 75 GB of RAM to generate database files requiring 1.5 GB of space.

Kraken classification of the i100 dataset (53.3 million paired reads) took 10 minutes, using 10 threads and 73.6 GB of RAM. This step is limited by the size of the database, which is loaded into RAM during classification. Bracken alone runs in under a second, using 13 MB of RAM. The Kraken classification file for the i100 data is 1.9 GB, while Bracken abundance estimation files require $\sim$ 20 KB of space.

## 4.4 Kraken 2 and Bracken

In 2018, Kraken 2 was released to improve upon the memory usage and speed of Kraken while utilizing the same classification algorithm and maintaining the same high accuracy [13]. In a direct comparison, classification with Kraken 2 required 85% less memory than Kraken 1 by utilizing a probabilistic, compact hash table to save the *k*-mer information.

### 4.4.1 Kraken 2/Bracken: i100 metagenomics experiment

In order to test the continued compatibility of Kraken 2 with Bracken, I first repeated the i100 metagenomics experiment using the large database. The Kraken 2

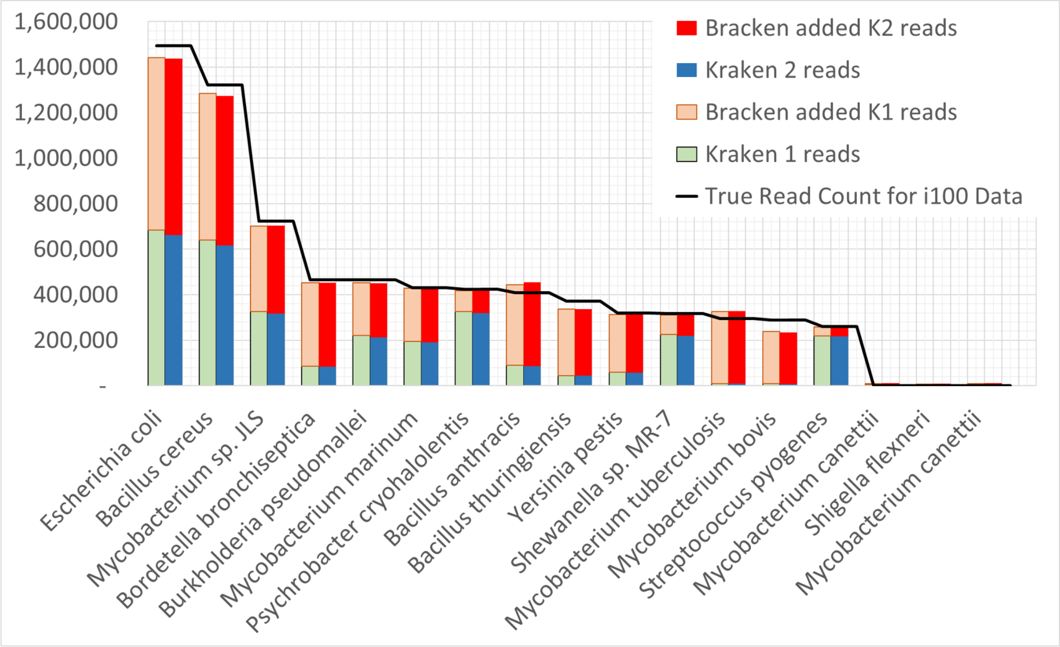


Figure 4.6: **Comparison of Bracken (red/light red) i100 species abundance when used alongside Kraken 1 (light green) vs. Kraken 2 (blue) for select species with the large database**. The black line shows the true number of reads from each species.

and Bracken results differed only slightly from the original Kraken 1 and Bracken abundance estimation results, with less than 1% change in MAPE. The final read counts for Kraken 2/Bracken were on average ¡ 1% different from the read counts with Kraken 1/Bracken. **Figure 4.6** shows the direct comparison between the two abundance estimation experiments for the species where Bracken performed the largest readjustment in read counts. In all cases, the final species read estimate was nearly identical to the true abundances.

While the accuracy of Bracken is maintained with Kraken 2, the build times and runtimes improved significantly. Originally, building of the Kraken 1 and Bracken database files using 10 threads required more than 8 hours in total. However, building of the Kraken 2 and Bracken database files required only 1 hour on the same computing system. Similarly, classifying and performing abundance estimation on the i100 data using Kraken 1/Bracken required 12 minutes while Kraken 2/Bracken performed the same steps in under 5 minutes. **Table 4.3** lists detailed timing, RAM, and space requirements for each file and step of the Bracken abundance estimation algorithm.

Table 4.3: **Kraken 1, Kraken 2, and Bracken timing and resource requirements for classifying the i100 data using the small Kraken database.** This table describes the time, RAM, and disk space required required for each abundance estimation step when Bracken is used either with Kraken 1 or Kraken 2. Disk space is measured for the generated files of each abundance estimation step. Notably, the timing, RAM, and space requirements differ between Kraken 1 and Kraken 2.

| Kraken 1 + Bracken (Large Database) | | | | |
|---|---|---|---|---|
| Step | Threads | Timing (H:MM:SS) | RAM | Space |
| 1. Build Kraken Database | 10 | 7:22:27 | 94.14 Gb | 70.6 Gb |
| 2. Generate database.kraken | 10 | 0:30:38 | 74.5 Gb | 1.0 Gb |
| 3. Generate database75mers.kraken | 10 | 0:11:05 | 1.7 Gb | 1.1 Mb |
| 4. Generate database75mers.distrib | 1 | 0:00:00 | 12.5 Mb | 466.8 Kb |
| 5. Classify data | 10 | 0:10:01 | 73.6 Gb | 1.9 Gb |
| 6. Generate report file | 1 | 0:02:07 | 655.9 Mb | 246 Kb |
| 7. Run Bracken | 1 | 0:00:00 | 12.6 Mb | 16.8 Kb |
| Steps 1-4 Total (once per database) | | 8:16:19 | 94.14 Gb | 72 Gb |
| Steps 5-7 Total (once per dataset) | | 0:12:08 | 73.6 Gb | 2 Gb |

| Kraken 2 + Bracken (Large Database) | | | | |
|---|---|---|---|---|
| Step | Threads | Timing (H:MM:SS) | RAM | Memory |
| 1. Build Kraken Database | 10 | 0:34:28 | 11.3 Gb | 10.7 Gb |
| 2. Generate database.kraken | 10 | 0:07:32 | 12.2 Gb | 2.4 Gb |
| 3. Generate database75mers.kraken | 10 | 0:13:48 | 3.1 Gb | 1.2 Mb |
| 4. Generate database75mers.distrib | 1 | 0:00:01 | 12.8 Mb | 504.8 Kb |
| 5. Classify data | 10 | 0:02:28 | 73.6 Mb | 2.2 Gb |
| 6. Generate report file | 1 | 0:02:17 | 656.5 Mb | 273 Kb |
| 7. Run Bracken | 1 | 0:00:00 | 13.2 Mb | 20.4 Kb |
| Steps 1-4 Total (once per database) | | 1:00:32 | 12.2 Gb | 13 Gb |
| Steps 5-7 Total (once per dataset) | | 0:04:45 | 656.5 Mb | 2.4 Gb |

### 4.4.2 Kraken 2/Bracken: Clade Exclusion Experiment

For additional testing, Derrick Wood and I executed the following strain exclusion experiment as described in [13]. Derrick Wood selected 40 prokaryotic genomes and simulated 1000 paired-end reads from each genome. Each of these genomes was then removed from the reference genome set used to generate the Kraken 1 and Kraken 2 databases and the corresponding Bracken database files. I then used Bracken to estimate both genus and species abundance estimation. These results are summarized in **Figure 4.7**

Although the true strain-level taxa are excluded from the database, Bracken recaptured most of the true genus-level and species-level sequence abundances using both Kraken 2 and Kraken 1 classification results. Comparing the results, the Bracken estimates were more accurate with Kraken 2 than with Kraken 1 at both the genus and species levels, likely owing to Kraken 2's higher sensitivity. For these experiments, Kraken 2 provided substantial increases in processing speed, performing classification over 5 times faster than Kraken 1. Bracken ran in less than 1 s.

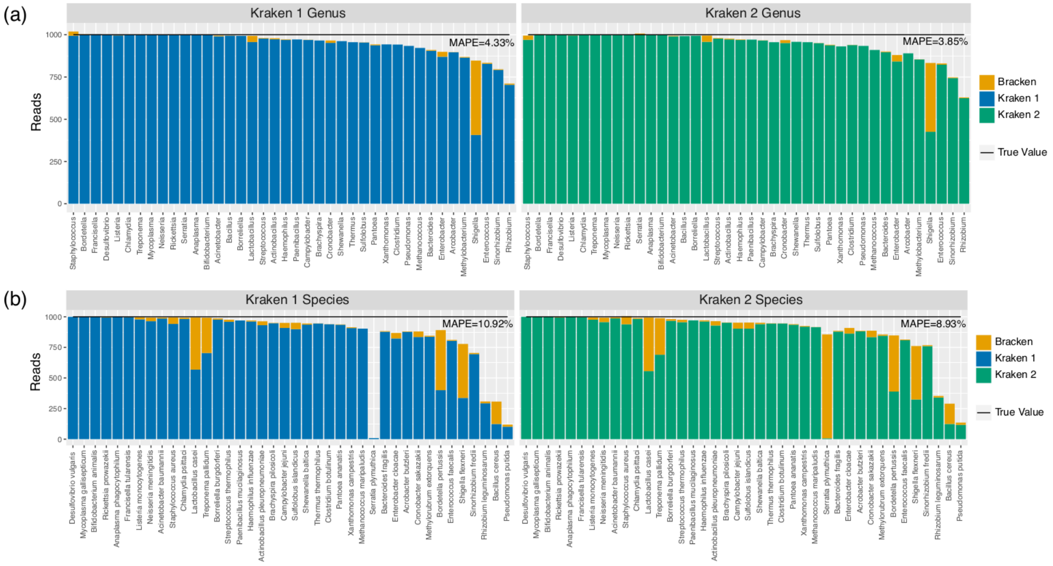


Figure 4.7: **Bracken performance on strain exclusion simulated prokaryotic data.** Each of the 40 genomes examined was removed from the reference genome set and each used to simulate 1000 paired-end reads. The same reference set was then used to build Kraken 1 and Kraken 2 databases and to classify each simulated fragment. I then used Bracken with both programs to estimate (a) genus sequence abundance estimation and (b) species sequence abundance estimation. “MAPE” is mean absolute percentage error.

## 4.5 Conclusion

Estimating the abundance of species, genera, phyla, or other taxonomic groups is a central step in the analysis of many metagenomics datasets. Metagenomics classifiers like Kraken provide a very fast and accurate way to label individual reads, and at higher taxonomic levels such as phyla, these assignments can be directly translated to abundance estimates. However, many reads cannot be unambiguously assigned to a single strain or species, for at least two reasons. First, many bacterial species are nearly identical, meaning that a read can match identically to two or more distinct species. Second, the bacterial taxonomy itself is undergoing constant revisions and updates, as genome sequencing reveals the need to re-assign species to new names. These revisions sometimes create new taxa that share near-identical sequence with a distinct species. In these situations, Kraken correctly assigns the read to a higher-level taxonomic category such as genus or family. This creates a problem in that Kraken's classifications cannot be used directly for species abundance estimation.

Bracken addresses this problem by probabilistically re-assigning reads from intermediate taxonomic nodes to the species level or above. As I have shown here, these re-assignments produce species-level abundance estimates that are very accurate, typically 98% correct or higher. For genus-level abundance, accuracy is even higher because fewer reads have ambiguous assignments at that level.

The release of Kraken 2 in 2018 contributed to improved performance and usability of Bracken. Kraken 2 allows for faster generation of Kraken and Bracken database files and reduced memory requirements for both programs. Additionally, Bracken maintains its accuracy in species and genus-level abundance estimation with Kraken 2.

# Chapter 5

# Ultrafast and accurate 16S microbial community analysis using Kraken 2 and Bracken

**Portions of this chapter originally appeared in:**

J. Lu, S. L. Salzberg "Ultrafast and accurate 16S microbial community analysis using Kraken 2." Microbiome 8, 124 (2020).

https://doi.org/10.1186/s40168-020-00900-2.

## 5.1 Introduction

Since the 1970s, sequencing of the 16S ribosomal RNA gene has been used for analyzing and identifying bacterial communities [83,84]. This technology targets the 16S rRNA gene, which has regions that are both highly conserved and highly variable (hypervariable) among bacterial species. The highly conserved regions allow for the design of "universal" PCR primers to target and amplify the 16S sequence, while the hypervariable regions allow for discrimination among different bacterial clades. These properties allow 16S sequencing experiments to capture nearly all of the bacteria in a microbial community, which can then be compared to large 16S databases to determine their identities.

Researchers have utilized 16S rRNA sequencing for a very broad range of environmental and clinical studies. For example, the Earth Microbiome Project [85] and other environmental studies have used 16S sequencing to reveal the bacterial diversity of soil [86,87], beach sand [88], and ocean environments [89]; while other studies targeted the microbiome of plants [90–92]. In the clinic, 16S rRNA has been used for diagnostic purposes to identify infectious bacterial species [93–95] and to characterize the role of bacterial diversity in diseases such as diabetes [96], Alzheimer's disease [97], cancer [98], and autism [99]. The international Human Microbiome Project has used 16S data to characterize the bacterial community present in the human gut, feces, skin, and other areas of the body [81, 100, 101].

### 5.1.1 16S Classification

Analysis of the bacterial community from a 16S rRNA sequencing experiment involves comparing the reads to reference database. The tool most widely used for 16S classification today is the Quantitative Insights into Microbial Ecology (QIIME) software package [14], which compares sequencing reads against a 16S reference database. The three standard 16S databases, each of which has somewhat different content, are Greengenes [102], SILVA [103], and RDP [104].

First released in 2011, QIIME 1 [14] provided 4 classification algorithms for 16S rRNA, respectively based on the RDP classifier [105], BLAST [39], UCLUST [106], and SortMeRNA [107]. In 2018, QIIME 2's q2-feature-classifier was released [15], adding 3 new classification algorithms based on scikit-learn's naïve Bayes algorithm [108], VSEARCH [109], and BLAST+ [110]. By default, QIIME 1 uses the UCLUST algorithm for classification while QIIME 2 suggests usage of the naïve Bayes algorithm.

In 2018, Almeida et al. performed benchmark tests comparing QIIME 2 to its predecessor, QIIME 1, and to two additional 16S classification tools, MAPseq [111] and mothur [112]. Almeida et al. evaluated the performance of each tool by classifying 16S rRNA reads that were simulated from bacteria known to be present in human gut, soil, and ocean microbiomes. That study concluded that QIIME 2 provides the best accuracy on the basis of recall and F-score. However, they also noted that QI-

IME 2 was the most computationally expensive, requiring substantially more CPU time and more memory than other tools.

### 5.1.2 Kraken, Kraken 2, and Bracken

The Kraken program uses an alignment-free algorithm that, when first released in 2014, was hundreds of times faster than any previously described program for shotgun metagenomics sequence analysis, with accuracy comparable to BLAST and superior to other tools [12]. Using a single thread, Kraken can classify metagenomics sequence data at a rate of >1 million reads per minute.

In 2016, Bracken was released as an extension to Kraken to estimate species abundance from Kraken's output [113]. As originally designed, Kraken attempts to classify each read as specifically as possible, allowing reads to be classified at any taxonomic level depending on how many genomes share the same sequence. For example, a read that has identical matches to two species will be classified at the genus level. Bracken adds the capability of abundance estimation to Kraken; i.e., using Kraken's read counts and prior knowledge of the database sequences, it estimates read counts for all species, genera, or higher-level taxa in a sample. For example, when Bracken is asked to estimate species counts, it will re-distribute all reads that Kraken assigns at the genus level (or higher) down to the species level.

Kraken 2, released in 2018, implemented significant changes to the database structure and classification steps to make databases smaller and classifications faster while maintaining compatibility with Bracken [13]. Because it uses the same classification algorithm, Kraken 2 has nearly the same precision and sensitivity as Kraken 1. However, Kraken 2 now also provides direct support for 16S classification with any of the three standard 16S databases: Greengenes, SILVA, and RDP. This new feature allowed direct comparison of Kraken 2 and Bracken to the current state-of-the-art programs for 16S classification, as described below.

### 5.1.3 Kraken 2 versus QIIME 2

In 2016, Lindgreen et al. evaluated 14 metagenomics classifiers, including Kraken 1 and QIIME 1 (UCLUST) [79]. That study showed that Kraken achieved the lowest false positive rate, 0%, while QIIME had a false positive rate of 0.28%. Kraken also had higher sensitivity than QIIME, correctly labeling 70% of the reads while QIIME was correct on 60%. Finally, Kraken obtained a Pearson correlation between the known and predicted abundances of phyla and genera of 0.99, versus 0.78 for QIIME. However, that study used different databases and different input data (reads produced by metagenomic shotgun sequencing) to evaluate these tools. For Kraken 1, Lindgreen et al. measured its performance on all input sequences from a shotgun metagenomics experiment, using a database containing all complete bacteria and archaeal genomes from RefSeq, while for QIIME 1, they analyzed its performance only on 16S rRNA sequences against the 16S Greengenes database.

Because QIIME is designed for 16S sequencing projects and Kraken has previously been used primarily for metagenomics shotgun sequencing projects, the tools have not been directly compared. Here, I compare QIIME 2 and Kraken 2 using the 16S rRNA reads generated in the Almeida et al. benchmark study [114], using both the Greengenes and SILVA 16S databases. I also show results for Kraken on the RDP database, which is not compatible with QIIME 2. Because I only tested the most recent version of each tool, I will henceforth refer to QIIME 2 as QIIME and Kraken 2 as Kraken.

## 5.2 Methods

### 5.2.1 Almeida Simulated Data

QIIME 2, Kraken 2, and Bracken were evaluated using the A500 synthetic microbiome samples generated by Almeida et al. [114] and available at `ftp://ftp.ebi.ac.uk/pub/databases/metagenomics/taxon_benchmarking/`. The A500 set contains 12 samples representing three different microbial environments: the human gut, ocean, and soil. For each of these environments, genomic sequences for their most abundant genera were extracted and randomly sampled. These human gut, ocean, and soil genomes then were sub-sampled four times to simulate 16S rRNA profiling using four different primer sets, generating $\sim 200,000$, 250-bp paired-end reads per primer sequence. The sub-sampling introduced a 2% random mutation to each sequencing read. Almeida et al. then performed pre-processing and quality control to filter sequences with ambiguous base calls. With three microbial environments and four primer sets, Almeida et. al. thereby generated 12 sets of synthetic communities for testing. Information about the software and primers used in dataset generation is further described in the Methods section of Almeida et al.

### 5.2.2 Software and Databases

The software packages tested are Kraken 2 (downloaded on 2020/03/05), Bracken v2.5 and QIIME 2 v2017.11. Kraken and Bracken database files were generated for Greengenes 13_8, SILVA 132, and RDP 11.5 database releases. QIIME 2 database files were generated for Greengenes 13_8 and SILVA 132.

### 5.2.3 Error Rate Calculations

For evaluating the accuracy of Kraken 2, Bracken, and QIIME 2, I calculated two different error metrics which compare the true genera distributions against those reported by each program. The first error metric is a modified mean absolute proportion error (MAPE) which compares the difference between the true read counts ($T_g$) for a given genus and the measured read counts ($A_g$) for that same genus.

$$MAPE = \sum_{g=1}^{n} \frac{T_g}{\sum_{g=1}^{n} T_g} \times \frac{|A_g - T_g|}{T_g} \tag{5.1}$$

Each difference is calculated as a fraction of the true counts and then weighted by the fraction of the total sample. $n$ is the total number of true genera in the sample.

The second metric, Bray-curtis dissimilarity [115], is a similar measurement of the dissimilarity between the true genera distribution and the measured genera distribution. The formula for Bray-curtis dissimilarity is:

$$BC_ij = 1 - \frac{2C_ij}{S_i + S_j} \tag{5.2}$$

where $C_ij$ is the sum of lesser reads for genera in common and $S_i = S_j$ is the total number of reads. In other words, for every true genus $g$ in the sample, if $T_g < A_g, C_ij = C_ij + T_g$. Otherwise if $T_g > A_g, C_ij = C_ij + A_g$.

$MAPE$ and $BC$ values both fall between 0 and 1, where larger values indicate a greater difference between samples and smaller values indicate a greater similarity.

### 5.2.4 Sensitivity and Precision (PPV) Calculations

As Kraken 2 provides taxonomic assignments for every read, I can use the taxonomic tree of each read to calculate sensitivity and precision at all taxonomic levels. For this example, I describe calculations of sensitivity and precision at the genus level. First, I calculate true positive ($TP$), vague positive ($VP$), false positive ($FP$), and false negative ($FN$) read counts. $TP$ is the number of reads correctly classified at the genus level. This includes reads that are classified as any species within the true genus. Vague positive ($VP$) reads account for the possibility that a read is classified as any ancestor of the true taxon. Therefore, $VP$ reads include $TP$ reads and reads assigned to ancestor taxa of the true genus. $FN$ reads are all classified reads that are not $VP$ reads. This includes reads classified at any taxa not within the direct lineage of the true genera. Finally, I define $FN$ as the number of unclassified reads. Notably, in all experiments, Kraken 2 did not label any read as unclassified ($FN = 0$).

From these values, I define sensitivity and precision (measured by positive-predictive-value, $PPV$) using the following two equations:

$$\begin{aligned} Sensitivity &= \frac{TP}{TP + VP + FN + FP} \\ &= \frac{TP}{TP + VP + FP} \end{aligned} \tag{5.3}$$

$$PPV = \frac{TP}{TP + FP} \tag{5.4}$$

## 5.3 Results

Prior to classification, Kraken requires users to first build a specialized database consisting of three files: taxo.k2d, opts.k2d, and hash.k2d. The user also can choose the value $k$ that determines the length of the sequences that Kraken uses for its index; every sequence (or $k$-mer) of length $k$ is associated with the species in which it occurs. $K$-mers that occur in two or more species are associated with the lowest common ancestor of those species. The database files contain the taxonomy and $k$-mer information for the specified database. Following generation of these files, Bracken requires users to generate a $k$-mer distribution file. Kraken and Bracken additionally allow the use of multiple threads to accelerate database construction. I tested building all files for the 16S Greengenes 13_8, SILVA 132, and RDP 11.5 databases using 1, 4, 8, and 16 threads. **Table 5.1** summarizes the contents of each of these databases.

Table 5.1: **16S Databases used for the metagenomics classifiers in this study.** For each of the most recently released versions of three 16S databases, this table describes the total number of sequences and the number of "traditional" nodes represented in their respective taxonomies. The Greengenes numbers refer to the 99% OTU database, and the SILVA values reflect the Ref NR 99 database.

| Database | Version | Release Date | Sequences |
|---|---|---|---|
| Greengenes | 13_8 | 08/15/2013 | 203,452 |
| SILVA | 132 | 12/13/2017 | 695,171 |
| RDP | 11.5 | 09/30/2016 | 3,356,808 |

| Database | Domains | Phyla | Classes | Orders | Family | Genera | Species |
|---|---|---|---|---|---|---|---|
| Greengenes | 2 | 89 | 248 | 404 | 513 | 2102 | 2952 |
| SILVA | 5 | 228 | 514 | 1277 | 1531 | 9379 | - |
| RDP | 2 | 60 | 99 | 154 | 384 | 2466 | - |

For QIIME, users generate the database (called a "classifier") by first converting sequence and taxonomy files into QIIME compatible .qza files. QIIME classifier generation is single-threaded. I built QIIME naïve-bayes classifiers for Greengenes 13_8 and SILVA 132.

**Figure 5.1A** compares the combined database building time for Kraken/Bracken against the classifier generation time of QIIME. Kraken was at least 9x faster than QIIME for database creation; e.g., it took 9 min to build the Greengenes database index, while QIIME required 78 minutes for the same database. For the SILVA database, Kraken required only 34 minutes while QIIME required more than 58 hours to build the same database.

To compare the accuracy of Kraken, Bracken, and QIIME, I classified 12 samples generated by Almeida et. al. [114]. These 12 samples, each containing just under 200,000 reads, represent 3 different metagenomes (human, ocean, and soil) and 4 different 16S primers (V12, V34, V4, and V45). The number of reads in each sample is shown in **Table 5.2**.

Table 5.2: **Sample Read Counts.** The read counts in each metagenome-primer sample. Each sample was generated as described in the Supplementary Methods.

| Read Counts | V12 | V34 | V4 | V45 | Total |
|---|---|---|---|---|---|
| Human microbiome | 186,689 | 189,972 | 193,787 | 192,319 | 762,767 |
| Soil microbiome | 196,254 | 193,564 | 196,226 | 194,325 | 780,369 |
| Ocean microbiome | 193,867 | 193,962 | 196,198 | 195,135 | 779,162 |

QIIME classifiers require one single file containing all de-multiplexed reads. Therefore, I provided QIIME with one file per metagenome, each containing reads from all

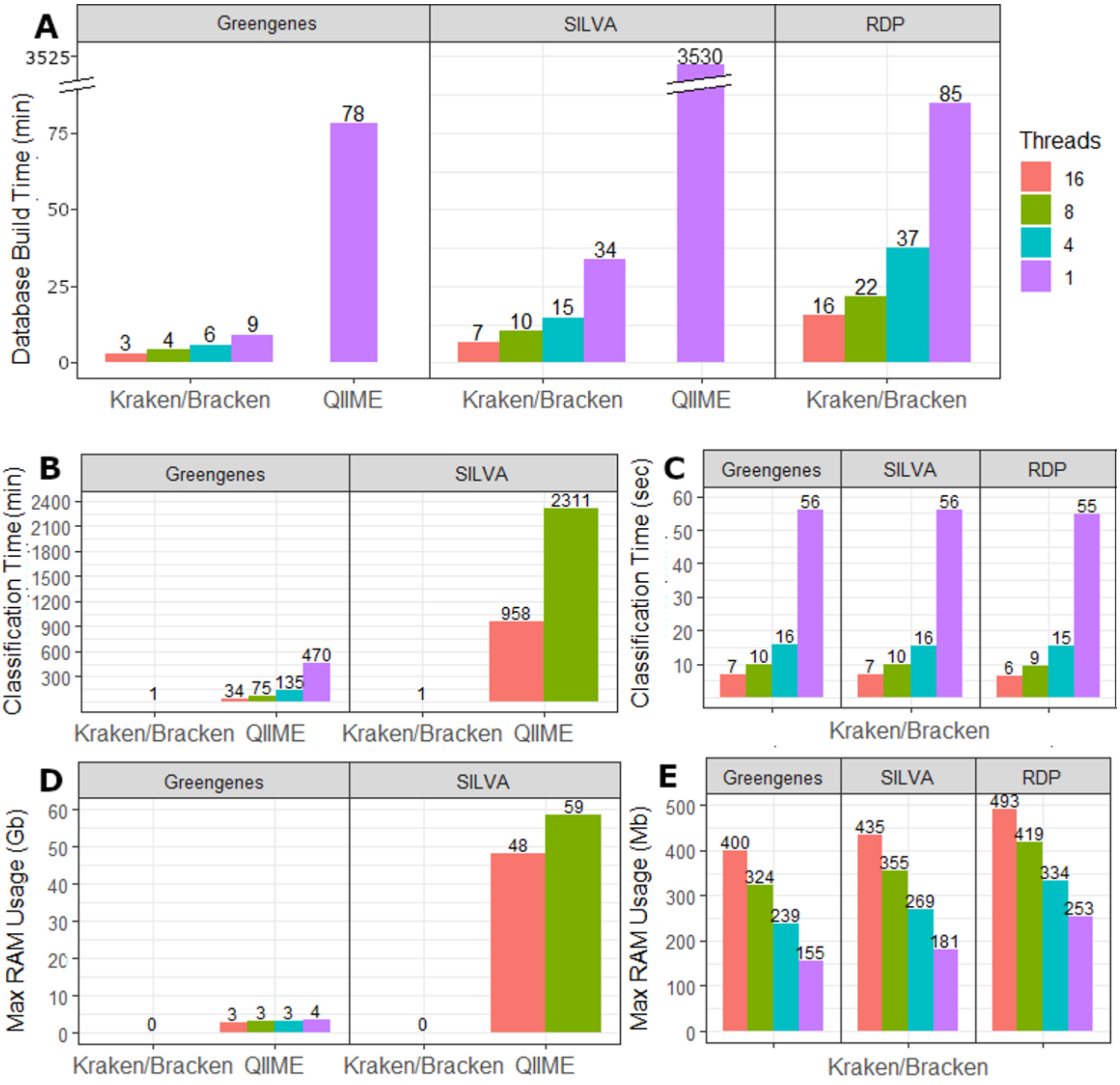


Figure 5.1: **Build and Classification Statistics A)** Required time to build each database for Kraken/Bracken and QIIME. Kraken and Bracken allow for multi-threading while QIIME is single-threaded. **B)** Average classification runtime in minutes for each database. Kraken/Bracken combined runtime is reported for only 1 thread as all runtimes are $<$ 1 min and bars are too small to be visible at this scale. QIIME was only run using 16 and 8 threads for SILVA. **C)** Classification runtime for Kraken and Bracken in seconds for all multi-threading options. **D)** Computational memory usage (RAM) for QIIME and Kraken/Bracken, shown in gigabytes (Gb). Kraken/Bracken RAM requirements reported only for 1 thread as Kraken and Bracken require $<$ 0.5Gb of RAM regardless of thread count. **E)** Computational memory usage (RAM) for Kraken/Bracken shown in megabytes (Mb).

4 primer sets. However, Kraken and Bracken classify samples one at a time, requiring each of the 12 samples to be processed individually.

Kraken and QIIME provide multi-threading options to speed up classification. I therefore tested Kraken and the QIIME Greengenes classifier using 1, 4, 8, and 16 threads. The QIIME SILVA classifier with 8 threads required approximately 1.5 days of run time, and for this reason I only tested it using 16 and 8 threads and did not evaluate the QIIME 2 SILVA classifier using 1 or 4 threads.

**Figure 5.1B** compares the average time in minutes required by QIIME as compared to Kraken/Bracken to classify a single metagenome using the 16S Greengenes and SILVA databases. Due to the very large difference in run time between tools, this figure compares the multi-threaded options of QIIME against the single-threaded classification time of Kraken/Bracken. **Figure 5.1C** reports the classification times of Kraken/Bracken in seconds.

Another important consideration for software selection is the computational memory resources required. I evaluated this by measuring the RAM in gigabytes (GB) required for both classifiers. **Figure 5.1D** compares the RAM required for the single-threaded runs of Kraken/Bracken against the multi-threaded runs using QIIME. Notably, all Kraken/Bracken runs used less than 0.5 GB of RAM, which appears in the figure as zero GB. To provide more detail on RAM usage, **Figure 5.1E** reports the RAM required by Kraken/Bracken in megabytes (MB) for all multi-threading options.

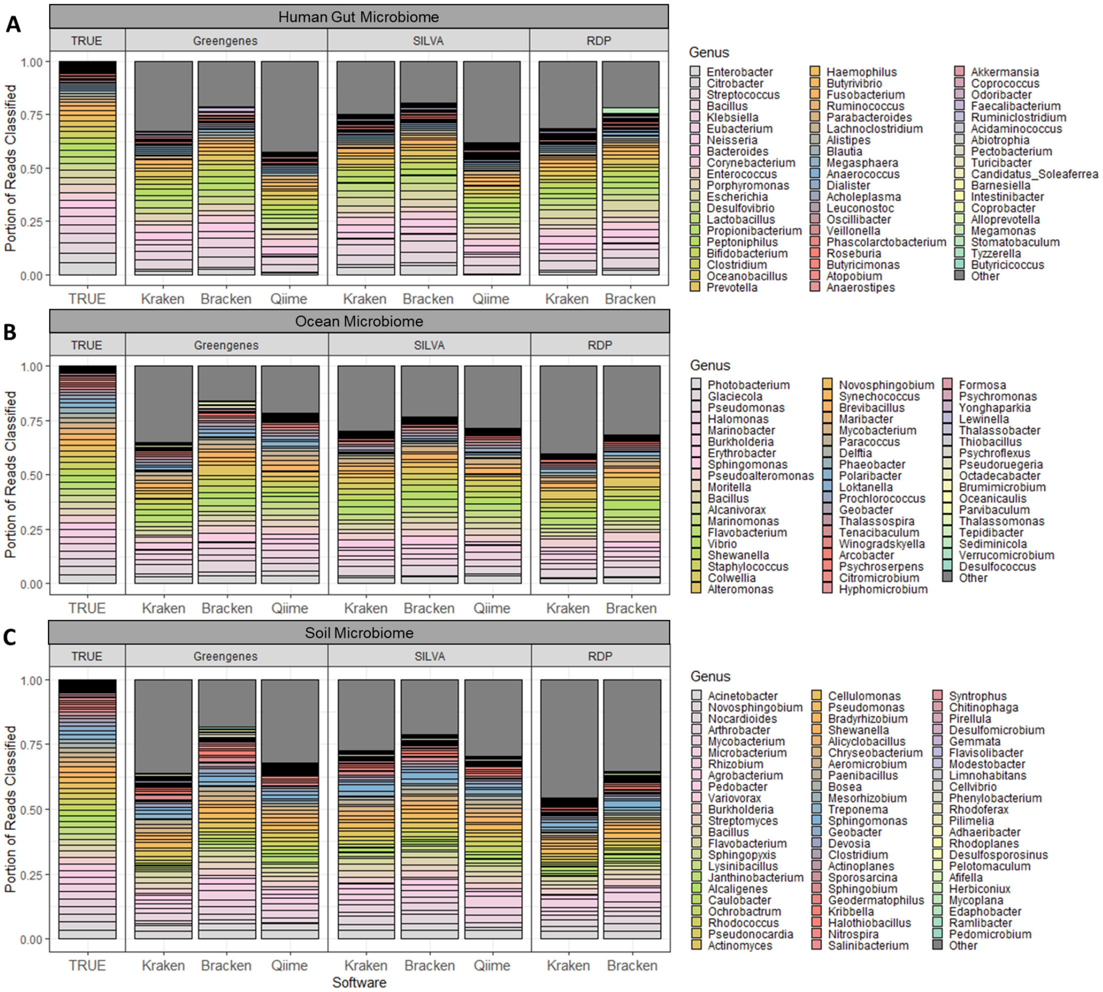


Figure 5.2: **Genera Distribution for Simulated Microbiota** This plot compares the true genus abundances against those abundances estimated by Kraken, Bracken, and QIIME, for each of the three simulated microbiome samples (**A** = human gut microbiome, **B** = ocean microbiome, **C** = soil microbiome). Only the correct genera are represented by different bars while read assignments to any incorrect taxon is included in “Other”.

**Figure 5.2** compares the true distribution of genera in each metagenomics sample against the genus-level counts reported by Kraken 2, Bracken, and QIIME 2. For clarity, this figure shows the combined read counts across the V12, V34, V4, and V45 samples for each metagenome.

I used two different metrics to evaluate the genus distribution accuracy: Mean Ab-

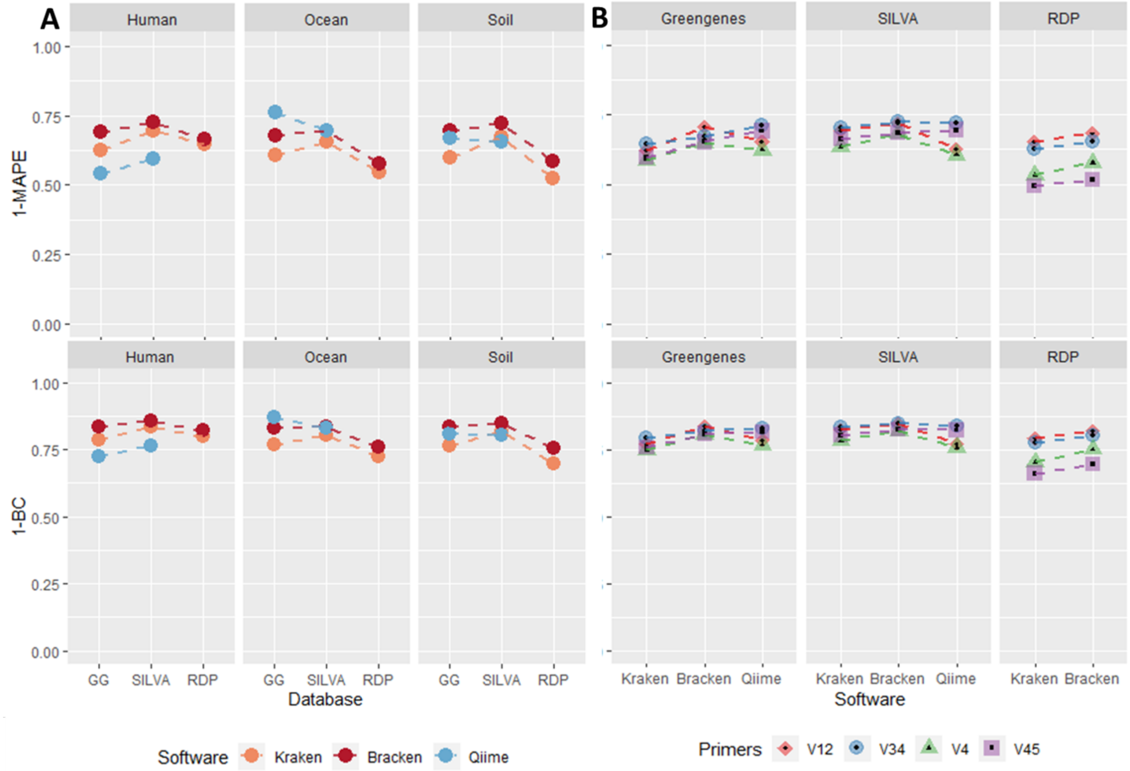


Figure 5.3: **MAPE and Bray-Curtis Dissimilarity** This plot evaluates classification accuracy by using the inverse of two error metrics: Mean Absolute Proportion Error ($MAPE$) and Bray-Curtis Dissimilarity ($BC$). **A** compares the accuracy of Kraken, Bracken, and Qiime when predicting the genus read counts across all samples for given metagenome/database. **B** compares the accuracy between the individual primers averaged across all 3 metagenomes for a given software/database. The top plots calculate accuracy as $1 - MAPE$ while the bottom plots evaluate $1 - BC$.

solute Percentage Error ($MAPE$) and Bray-Curtis dissimilarity ($BC$). Both error rates measure how different the predicted sample distribution is from the true genera counts. Given these two metrics, I evaluate accuracy as $1 - MAPE$ and $1 - BC$. **Figure 5.3A** compares the accuracy of each tool when calculating the correct combined read counts at the genus level for each metagenome. For further insight into how the choice of 16S primer affects genus distribution accuracy, I evaluated the average $MAPE$ and average $BC$ across all 3 metagenome samples for each program/database. **Figure 5.3B** uses these averages to compare the accuracy between 16S primers.

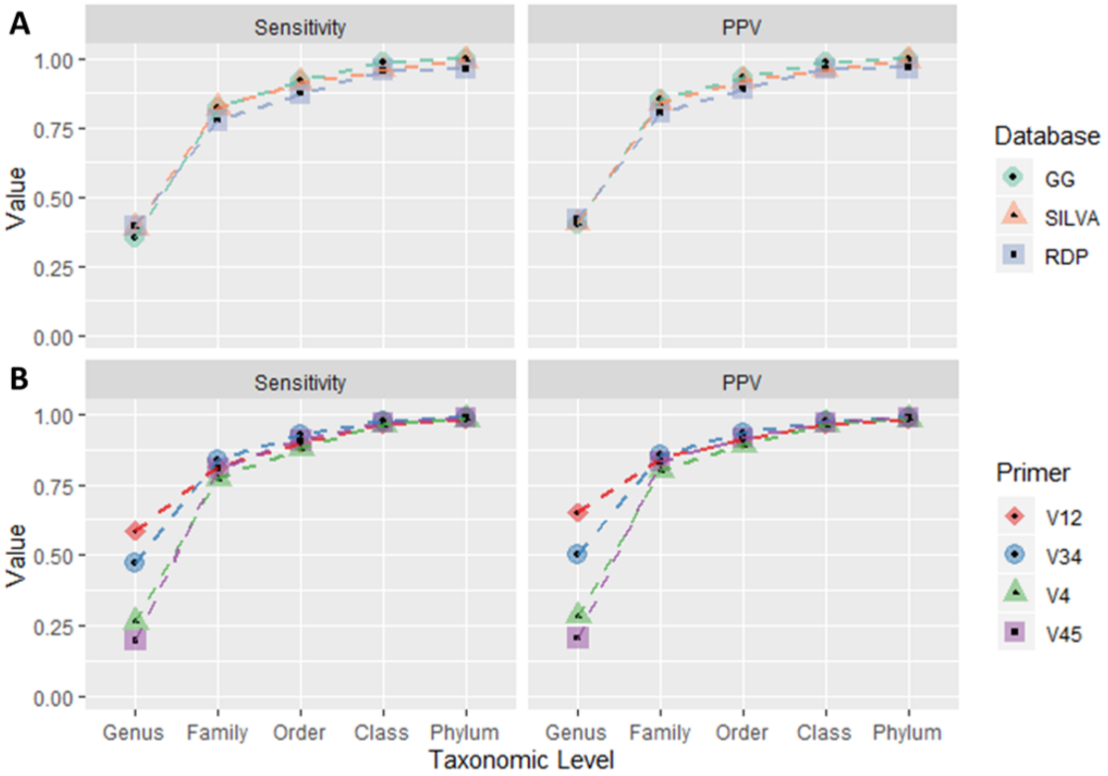


Figure 5.4: **Kraken Per-Read Accuracy** As Kraken is the only tool tested that provides per-read taxonomy assignments, I evaluate the sensitivity and precision ($PPV$) of Kraken 2's taxonomy assignments at each major taxonomic level

While all tools tested provide general read counts per genus, Kraken is the only tool that directly assigns each read with a taxonomic label. Using this information, I can calculate Kraken's accuracy when classifying reads at major taxonomic levels in terms of sensitivity and precision. I measure precision by positive predictive value (PPV, see Supplemental Methods for more details). **Figure 5.4** displays Kraken's average sensitivity and PPV for each database used (**Figure 5.4A**) and for each 16S primer used in generating the samples (**Figure 5.4B**).

## 5.4 Discussion

In this study, I evaluated three systems for classification and abundance estimation of 16S sequencing data sets: Kraken 2, Bracken, and QIIME 2. For Kraken and Bracken, I used three 16S databases: Greengenes, SILVA, and RDP; while for QIIME, I only evaluated Greengenes and SILVA. I then used these tools/databases to classify 12 samples generated by Almeida et. al [114], which represent 3 simulated metagenomes (human gut, ocean, and soil) and 4 different 16S primers (V12, V34, V4, and V45). In total, I collected 36 different results using Kraken/Bracken and 24 different results using QIIME.

### 5.4.1 Database Building Time

For all systems compared here, database build time is a function of the number of sequences in the database. Because 16S Greengenes is the smallest database (with $\sim 200,000$ sequences) and 16S RDP is the largest (with $\sim$ 3.4 million sequences), generation of database files is fastest with Greengenes and slowest with RDP.

When comparing single-threaded Kraken/Bracken against QIIME, Kraken and Bracken combined require far less build time. For the smallest 16S database, Greengenes, QIIME required more than an hour to generate the naïve Bayes classifier (**Figure 5.1A**). By comparison, single-threaded Kraken and Bracken combined required less than 10 minutes to create the database files. For 16S SILVA, with nearly 700,000 sequences, QIIME 2 required more than 58 hours for classifier generation while the single-threaded Kraken/Bracken required only $\sim$30 minutes. I additionally note that the largest 16S database, RDP, required a little more than an hour for single-threaded Kraken 2 and Bracken to create the database files. As mentioned above, the RDP database is incompatible with QIIME 2. The multi-threaded nature of Kraken 2 and Bracken further accelerate the database building process, with 4 threads halving the required build time (**Figure 5.1A**).

### 5.4.2 Classification Time/Memory Requirements

As observed by Almeida et. al. [114], QIIME 2 requires more computational resources than other methods during classification. With the use of 16 CPU threads, QIIME required $\sim$ 35 minutes on average to classify the human, ocean, and soil metagenomic samples using the Greengenes database (**Figure 5.1B**). The QIIME's SILVA classifier required $\sim$ 16 hours on average. By comparison, single-threaded Kraken 2 and Bracken required on average 1 minute per metagenomic sample. This runtime decreases from 1 minute to 15, 10, and 6 seconds for 4, 8, and 16 threads respectively (**Figure 5.1C**). The runtime of Kraken 2 and Bracken was nearly the same for all three databases. Thus Kraken or Braken is at least 350 times faster (6 seconds vs. 35 minutes) than QIIME 2 when run with 16 parallel threads.

The amount of computer memory (RAM) required by each system also varied widely (**Figure 5.1D**). For all three databases, single-threaded Kraken required $<$ 260 MB of RAM. However, the single-threaded QIIME Greengenes classifier required $\sim$ 3.6 GB of RAM. Increasing the number of threads for Kraken also increases the total RAM used, with 16 threads using $\sim 400 - 500$ MB of RAM for each of the Kraken databases (**Figure 5.1E**). However, for QIIME, increasing the number of threads decreased the total RAM: the QIIME Greengenes classifier with 16 threads used $\sim$2.7 GB, and the QIIME SILVA classifier with 16 threads used 48 GB of RAM (**Figure 5.1D**).

### 5.4.3 Accuracy of abundance estimation

Finally, I compared the accuracy of all three tools based on their ability to recreate the true genus distribution of the simulated samples (**Figure 5.2**). I quantified the accuracy of these distributions using both MAPE and Bray-Curtis dissimilarity.

In all cases, Bracken performed better than Kraken 2, which was expected because Kraken is a classification tool, not an abundance estimation system. Kraken classifies reads at any level in the taxonomy, which means that some reads might be assigned to a higher level genus; e.g., any read that has equally good matches to two genera will be assigned to the family containing them. For the simulated datasets in this study, Kraken assigned from 7-30% of the reads to levels above genus. These reads are not incorrectly classified, but the result is that Kraken underestimates the abundances of their genera. By contrast, Bracken is designed to use Kraken's classification data to estimate all read counts at the genus level, thereby improving on Kraken's genus-level distribution.

On average, Bracken performed the best, having the lowest average error rates across all three 16S databases. Bracken also had the lowest error rate for 8/9 combinations of samples and databases. The only sample where QIIME 2 had a lower error rate than Bracken was in the classification of the ocean samples against the 16S Greengenes database (**Figure 5.3A**). However, QIIME 2 had the highest error rate when classifying the human sample against Greengenes or SILVA, regardless of whether measured by MAPE or Bray-Curtis dissimilarity.

In analyzing the trends across the databases using both MAPE and Bray-Curtis, Bracken performed the best using the 16S SILVA database and performed the worst using the 16S RDP database. 16S RDP yielded on average 0.391 MAPE and 0.221 BC Index while 16S SILVA only yielded a 0.286 MAPE and a 0.153 BC Index. 16S Greengenes with Bracken had an average of 0.313 MAPE and a 0.165 BC Index. Although QIIME 2 was not tested on 16S RDP, QIIME 2 yielded the same trends when comparing 16S Greengenes and SILVA, with 16S SILVA outperforming 16S Greengenes in almost all cases.

In addition to evaluating the different tools, I also evaluated the accuracy of each of the primer sets (V12, V34, V4, and V45) that were used by Almeida et. al. [114]. **Figure 5.3B** shows the average accuracy of each primer set across all 3 metagenomes for a given software/database pairing. For both Greengenes and SILVA, the samples generated using V34 and V12 performed slightly better. However, for RDP, the difference in accuracy between primer samples is further magnified. When classifying with the RDP database, both Kraken and Bracken had significantly better results for the V12 and V34 samples (**Figure 5.3B**).

### 5.4.4 Per-Read Classification Accuracy

Kraken is the only program of the three tested here that provide per-read assignments by default, allowing us to compute the read-level accuracy of its taxonomy assignments. Per-read accuracy is somewhat dependent on the reference database, but highly dependent on the 16S primer set (**Figure 5.4B**). In particular, Kraken had three times higher sensitivity (60%) and PPV (65%) when classifying reads generated using V12 primers versus those generated from V45 primers (20% and 21%).

As expected, sensitivity and precision increased with taxonomic level, with class and phylum sensitivity and precision exceeding 0.95 for all sample sets and all databases.

### 5.4.5 Taxonomy Inconsistencies

In our experiments, I discovered that the accuracy of 16S analysis is highly dependent on the choice of 16S database. The 170 distinct genera present in our human, ocean, and soil metagenomes were selected from the NCBI taxonomy, but none of the three 16S database taxonomies contains precisely the same genera. Each 16S database is independently curated from different reference sets, resulting in substantial differences among the taxonomies [116]. Among the 170 unique genera uses here, 22 are missing from Greengenes, 19 have different names or are mapped to multiple genera in RDP, and 16 have different names in Silva. For example, *Agrobacterium*, *Burkholderia*, and

*Rhizobium* are not unique genera in the 16S SILVA taxonomy, but are combined into a single “Allorhizobium-Neorhizbium-Pararhizobium-Rhizobium” genus. *Escherichia* and *Shigella* are also combined into the “Escherichia-Shigella” genus in 16S SILVA. The *Clostridium* sequences in 16S SILVA are split between 19 different genera, each with the prefix of “Clostridium sensu stricto” followed by a number 1-19.

## 5.5 Conclusion

Although each of the 16S databases represents a large number of bacterial organisms, the accuracy of metagenomics classifiers varied substantially among them. In our experiments, 16S SILVA provided the lowest error rates and highest per-read accuracy regardless of the software used in classification. Across all databases, Kraken 2 and Bracken outperformed QIIME 2 in terms of computational requirements, runtime, and accuracy. Single-threaded Kraken/Bracken was nearly 8x faster than QIIME 2 at building the 16S Greengenes database and 100x faster at building a 16S SILVA database. Kraken and Bracken also allow for multi-threaded database building, which allows any 16S database to be built in less than 20 minutes. For classification, Kraken/Bracken used 20 times less RAM, performed 300 times faster, and achieved better genus-level resolution than QIIME 2.

# Chapter 6

# Conclusion

In the constantly evolving and growing field of metagenomics, SkewIT, the EuPathDB, Kraken, and Bracken provide new methods and resources for accurate analysis of microbial genomes. Each of these projects addresses different but related needs in metagenomics research, allowing insight into microbial organisms related to human diseases or living in microscopic communities worldwide. Therefore, each of these projects provides future opportunities for improving the quality of microbial genomes and investigating additional microbial environments.

Although SkewIT has provided insight into the 15,000+ bacterial complete genomes in NCBI's RefSeq database, new bacterial genomes are generated daily, representing new bacterial species or related strains of existing species. Applying SkewIT to these new genome assemblies will provide a simple quantitative measurement of the nucleotide distribution across the genome. With these measurements, SkewIT will reveal the similarities or differences between the genomic structure of related bacterial organisms while highlighting potential mis-assemblies.

The cleaned EuPathDB-46 is a valuable resource for identifying eukaryotic pathogens in human diseases or for accurate identification of the eukaryotic microorganisms in environmental samples. However, with the lack of eukaryotic microorganism genomes in main genomic databases (such as NCBI RefSeq), the cleaning method used for removing contamination in the EuPathDB genomes is equally important. Applying the cleaning method to new draft genomes for eukaryotic microorganisms will minimize the false positives when using the genomes in metagenomics classification experiments. The resulting genomes can then be added to existing databases of bacterial and viral organisms that previously lacked the representation of these species.

Finally, the classification methods of Kraken 1 and Kraken 2 and the abundance estimation methods of Bracken continue to be important for the broad metagenomics community. As previously described, these methods allow for insight into the human gut microbiome or skin microbiome environments, but also have the possibility of being applied to other human samples. Furthermore, while Kraken and Bracken have been applied to identify microbes in soil or ocean water, these tools allow analysis of a wide range of sequenced environmental samples, such as sand, lake, ocean, and forest samples. Lastly, Kraken and Bracken can also characterize the microbes coexisting with other non-human organisms, such as symbionts of fish, poultry, cows, etc.

In addition to various sample types, the metagenomics community is also evolving to incorporate new sequencing technologies, with long-read sequencing, such as Nanopore sequencing, presenting new challenges for metagenomics. With the adaptation of long-read sequencing technologies for metagenomics experiments, tools such as Kraken and Bracken must evolve to accommodate much longer DNA fragments.

# Bibliography

[1] J. C. Venter, M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural, G. G. Sutton, H. O. Smith, M. Yandell, C. A. Evans, R. A. Holt, J. D. Gocayne, P. Amanatides, R. M. Ballew, D. H. Huson, J. R. Wortman, Q. Zhang, C. D. Kodira, X. H. Zheng, L. Chen, M. Skupski, G. Subramanian, P. D. Thomas, J. Zhang, G. L. Gabor Miklos, C. Nelson, S. Broder, A. G. Clark, J. Nadeau, V. A. McKusick, N. Zinder, A. J. Levine, R. J. Roberts, M. Simon, C. Slayman, M. Hunkapiller, R. Bolanos, A. Delcher, I. Dew, D. Fasulo, M. Flanigan, L. Florea, A. Halpern, S. Hannenhalli, S. Kravitz, S. Levy, C. Mobarry, K. Reinert, K. Remington, J. Abu-Threideh, E. Beasley, K. Biddick, V. Bonazzi, R. Brandon, M. Cargill, I. Chandramouliswaran, R. Charlab, K. Chaturvedi, Z. Deng, V. Di Francesco, P. Dunn, K. Eilbeck, C. Evangelista, A. E. Gabrielian, W. Gan, W. Ge, F. Gong, Z. Gu, P. Guan, T. J. Heiman, M. E. Higgins, R. R. Ji, Z. Ke, K. A. Ketchum, Z. Lai, Y. Lei, Z. Li, J. Li, Y. Liang, X. Lin, F. Lu, G. V. Merkulov, N. Milshina, H. M. Moore, A. K. Naik, V. A. Narayan,

B. Neelam, D. Nusskern, D. B. Rusch, S. Salzberg, W. Shao, B. Shue, J. Sun, Z. Wang, A. Wang, X. Wang, J. Wang, M. Wei, R. Wides, C. Xiao, C. Yan, A. Yao, J. Ye, M. Zhan, W. Zhang, H. Zhang, Q. Zhao, L. Zheng, F. Zhong, W. Zhong, S. Zhu, S. Zhao, D. Gilbert, S. Baumhueter, G. Spier, C. Carter, A. Cravchik, T. Woodage, F. Ali, H. An, A. Awe, D. Baldwin, H. Baden, M. Barnstead, I. Barrow, K. Beeson, D. Busam, A. Carver, A. Center, M. L. Cheng, L. Curry, S. Danaher, L. Davenport, R. Desilets, S. Dietz, K. Dodson, L. Doup, S. Ferriera, N. Garg, A. Gluecksmann, B. Hart, J. Haynes, C. Haynes, C. Heiner, S. Hladun, D. Hostin, J. Houck, T. Howland, C. Ibegwam, J. Johnson, F. Kalush, L. Kline, S. Koduru, A. Love, F. Mann, D. May, S. McCawley, T. McIntosh, I. McMullen, M. Moy, L. Moy, B. Murphy, K. Nelson, C. Pfannkoch, E. Pratts, V. Puri, H. Qureshi, M. Reardon, R. Rodriguez, Y. H. Rogers, D. Romblad, B. Ruhfel, R. Scott, C. Sitter, M. Smallwood, E. Stewart, R. Strong, E. Suh, R. Thomas, N. N. Tint, S. Tse, C. Vech, G. Wang, J. Wetter, S. Williams, M. Williams, S. Windsor, E. Winn-Deen, K. Wolfe, J. Zaveri, K. Zaveri, J. F. Abril, R. Guigó, M. J. Campbell, K. V. Sjolander, B. Karlak, A. Kejariwal, H. Mi, B. Lazareva, T. Hatton, A. Narechania, K. Diemer, A. Muruganujan, N. Guo, S. Sato, V. Bafna, S. Istrail, R. Lippert, R. Schwartz, B. Walenz, S. Yooseph, D. Allen, A. Basu, J. Baxendale, L. Blick, M. Caminha, J. Carnes-Stine, P. Caulk, Y. H. Chiang, M. Coyne, C. Dahlke, A. Mays, M. Dombroski, M. Donnelly, D. Ely, S. Esparham, C. Fosler, H. Gire, S. Glanowski, K. Glasser, A. Glodek, M. Gorokhov, K. Graham, B. Gropman, M. Harris, J. Heil, S. Henderson, J. Hoover, D. Jennings, C. Jordan, J. Jordan, J. Kasha, L. Kagan, C. Kraft, A. Levitsky, M. Lewis, X. Liu, J. Lopez, D. Ma, W. Majoros, J. McDaniel, S. Murphy, M. Newman, T. Nguyen, N. Nguyen, M. Nodell, S. Pan, J. Peck, M. Peterson, W. Rowe, R. Sanders, J. Scott, M. Simpson, T. Smith, A. Sprague, T. Stockwell, R. Turner, E. Venter, M. Wang, M. Wen, D. Wu, M. Wu, A. Xia, A. Zandieh, and X. Zhu, "The sequence of the human genome," *Science*, vol. 291, no. 5507, pp. 1304–1351, Feb. 2001.

[2] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh, R. Funke, D. Gage, K. Harris, A. Heaford, J. Howland, L. Kann, J. Lehoczky, R. LeVine, P. McEwan, K. McKernan, J. Meldrim, J. P. Mesirov, C. Miranda, W. Morris, J. Naylor, C. Raymond, M. Rosetti, R. Santos, A. Sheridan, C. Sougnez, Y. Stange-Thomann, N. Stojanovic, A. Subramanian, D. Wyman, J. Rogers, J. Sulston, R. Ainscough, S. Beck, D. Bentley, J. Burton, C. Clee, N. Carter, A. Coulson, R. Deadman, P. Deloukas, A. Dunham, I. Dunham, R. Durbin, L. French, D. Grafham, S. Gregory, T. Hubbard, S. Humphray, A. Hunt, M. Jones, C. Lloyd, A. McMurray, L. Matthews, S. Mercer, S. Milne, J. C. Mullikin, A. Mungall, R. Plumb, M. Ross, R. Shownkeen, S. Sims, R. H. Waterston, R. K. Wilson, L. W. Hillier, J. D. McPherson, M. A. Marra, E. R. Mardis, L. A. Fulton, A. T. Chinwalla,

K. H. Pepin, W. R. Gish, S. L. Chissoe, M. C. Wendl, K. D. Delehaunty, T. L. Miner, A. Delehaunty, J. B. Kramer, L. L. Cook, R. S. Fulton, D. L. Johnson, P. J. Minx, S. W. Clifton, T. Hawkins, E. Branscomb, P. Predki, P. Richardson, S. Wenning, T. Slezak, N. Doggett, J. F. Cheng, A. Olsen, S. Lucas, C. Elkin, E. Uberbacher, M. Frazier, R. A. Gibbs, D. M. Muzny, S. E. Scherer, J. B. Bouck, E. J. Sodergren, K. C. Worley, C. M. Rives, J. H. Gorrell, M. L. Metzker, S. L. Naylor, R. S. Kucherlapati, D. L. Nelson, G. M. Weinstock, Y. Sakaki, A. Fujiyama, M. Hattori, T. Yada, A. Toyoda, T. Itoh, C. Kawagoe, H. Watanabe, Y. Totoki, T. Taylor, J. Weissenbach, R. Heilig, W. Saurin, F. Artiguenave, P. Brottier, T. Bruls, E. Pelletier, C. Robert, P. Wincker, D. R. Smith, L. Doucette-Stamm, M. Rubenfield, K. Weinstock, H. M. Lee, J. Dubois, A. Rosenthal, M. Platzer, G. Nyakatura, S. Taudien, A. Rump, H. Yang, J. Yu, J. Wang, G. Huang, J. Gu, L. Hood, L. Rowen, A. Madan, S. Qin, R. W. Davis, N. A. Federspiel, A. P. Abola, M. J. Proctor, R. M. Myers, J. Schmutz, M. Dickson, J. Grimwood, D. R. Cox, M. V. Olson, R. Kaul, C. Raymond, N. Shimizu, K. Kawasaki, S. Minoshima, G. A. Evans, M. Athanasiou, R. Schultz, B. A. Roe, F. Chen, H. Pan, J. Ramser, H. Lehrach, R. Reinhardt, W. R. McCombie, M. de la Bastide, N. Dedhia, H. Blöcker, K. Hornischer, G. Nordsiek, R. Agarwala, L. Aravind, J. A. Bailey, A. Bateman, S. Batzoglou, E. Birney, P. Bork, D. G. Brown, C. B. Burge, L. Cerutti, H. C. Chen, D. Church, M. Clamp, R. R. Copley, T. Doerks, S. R. Eddy, E. E. Eichler, T. S. Furey, J. Galagan, J. G. Gilbert, C. Harmon, Y. Hayashizaki, D. Haussler, H. Hermjakob, K. Hokamp, W. Jang, L. S. Johnson, T. A. Jones, S. Kasif, A. Kaspryzk, S. Kennedy, W. J. Kent, P. Kitts, E. V. Koonin, I. Korf, D. Kulp, D. Lancet, T. M. Lowe, A. McLysaght, T. Mikkelsen, J. V. Moran, N. Mulder, V. J. Pollara, C. P. Ponting, G. Schuler, J. Schultz, G. Slater, A. F. Smit, E. Stupka, J. Szustakowki, D. Thierry-Mieg, J. Thierry-Mieg, L. Wagner, J. Wallis, R. Wheeler, A. Williams, Y. I. Wolf, K. H. Wolfe, S. P. Yang, R. F. Yeh, F. Collins, M. S. Guyer, J. Peterson, A. Felsenfeld, K. A. Wetterstrand, A. Patrinos, M. J. Morgan, P. de Jong, J. J. Catanese, K. Osoegawa, H. Shizuya, S. Choi, Y. J. Chen, J. Szustakowki, and International Human Genome Sequencing Consortium, "Initial sequencing and analysis of the human genome," *Nature*, vol. 409, no. 6822, pp. 860–921, Feb. 2001.

[3] J. C. Venter, K. Remington, J. F. Heidelberg, A. L. Halpern, D. Rusch, J. A. Eisen, D. Wu, I. Paulsen, K. E. Nelson, W. Nelson, D. E. Fouts, S. Levy, A. H. Knap, M. W. Lomas, K. Nealson, O. White, J. Peterson, J. Hoffman, R. Parsons, H. Baden-Tillson, C. Pfannkoch, Y.-H. Rogers, and H. O. Smith, "Environmental genome shotgun sequencing of the sargasso sea," *Science*, vol. 304, no. 5667, pp. 66–74, Apr. 2004.

[4] NCBI Resource Coordinators, "Database resources of the national center for biotechnology information," *Nucleic Acids Res.*, vol. 42, no. Database issue,

pp. D7–17, Jan. 2014.

[5] N. A. O'Leary, M. W. Wright, J. R. Brister, S. Ciufo, D. Haddad, R. McVeigh, B. Rajput, B. Robbertse, B. Smith-White, D. Ako-Adjei, A. Astashyn, A. Badretdin, Y. Bao, O. Blinkova, V. Brover, V. Chetvernin, J. Choi, E. Cox, O. Ermolaeva, C. M. Farrell, T. Goldfarb, T. Gupta, D. Haft, E. Hatcher, W. Hlavina, V. S. Joardar, V. K. Kodali, W. Li, D. Maglott, P. Masterson, K. M. McGarvey, M. R. Murphy, K. O'Neill, S. Pujar, S. H. Rangwala, D. Rausch, L. D. Riddick, C. Schoch, A. Shkeda, S. S. Storz, H. Sun, F. Thibaud-Nissen, I. Tolstoy, R. E. Tully, A. R. Vatsan, C. Wallin, D. Webb, W. Wu, M. J. Landrum, A. Kimchi, T. Tatusova, M. DiCuccio, P. Kitts, T. D. Murphy, and K. D. Pruitt, "Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation," *Nucleic Acids Res.*, vol. 44, no. D1, pp. D733–45, Jan. 2016.

[6] C. S. Riesenfeld, P. D. Schloss, and J. Handelsman, "Metagenomics: genomic analysis of microbial communities," *Annu. Rev. Genet.*, vol. 38, pp. 525–552, 2004.

[7] F. P. Breitwieser, M. Pertea, A. V. Zimin, and S. L. Salzberg, "Human contamination in bacterial genomes has created thousands of spurious proteins," *Genome Res.*, vol. 29, no. 6, pp. 954–960, Jun. 2019.

[8] M. S. Longo, M. J. O'Neill, and R. J. O'Neill, "Abundant human DNA contamination identified in non-primate genome databases," *PLoS One*, vol. 6, no. 2, p. e16410, Feb. 2011.

[9] S. Mukherjee, M. Huntemann, N. Ivanova, N. C. Kyrpides, and A. Pati, "Large-scale contamination of microbial isolate genomes by illumina PhiX control," *Stand. Genomic Sci.*, vol. 10, p. 18, Mar. 2015.

[10] K. Kryukov and T. Imanishi, "Human contamination in public genome assemblies," *PLoS One*, vol. 11, no. 9, p. e0162424, Sep. 2016.

[11] M. Steinegger and S. L. Salzberg, "Terminating contamination: large-scale search identifies more than 2,000,000 contaminated entries in GenBank," *bioRxiv*, p. 2020.01.26.920173, Jan. 2020.

[12] D. E. Wood and S. L. Salzberg, "Kraken: ultrafast metagenomic sequence classification using exact alignments," *Genome Biol.*, vol. 15, no. 3, p. R46, Mar. 2014.

[13] D. E. Wood, J. Lu, and B. Langmead, "Improved metagenomic analysis with kraken 2," *Genome Biol*, p. 20, Sep. 2019.

[14] J. G. Caporaso, J. Kuczynski, J. Stombaugh, K. Bittinger, F. D. Bushman, E. K. Costello, N. Fierer, A. G. Peña, J. K. Goodrich, J. I. Gordon, G. A.

Huttley, S. T. Kelley, D. Knights, J. E. Koenig, R. E. Ley, C. A. Lozupone, D. McDonald, B. D. Muegge, M. Pirrung, J. Reeder, J. R. Sevinsky, P. J. Turnbaugh, W. A. Walters, J. Widmann, T. Yatsunenko, J. Zaneveld, and R. Knight, "QIIME allows analysis of high-throughput community sequencing data," *Nat. Methods*, vol. 7, no. 5, pp. 335–336, May 2010.

[15] N. A. Bokulich, B. D. Kaehler, J. R. Rideout, M. Dillon, E. Bolyen, R. Knight, G. A. Huttley, and J. Gregory Caporaso, "Optimizing taxonomic classification of marker-gene amplicon sequences with QIIME 2's q2-feature-classifier plugin," *Microbiome*, vol. 6, no. 1, p. 90, May 2018.

[16] J. R. Lobry, "Asymmetric substitution patterns in the two DNA strands of bacteria," *Mol. Biol. Evol.*, vol. 13, no. 5, pp. 660–665, May 1996.

[17] A. Grigoriev, "Analyzing genomes with cumulative skew diagrams," *Nucleic Acids Res.*, vol. 26, no. 10, pp. 2286–2290, May 1998.

[18] R. Okazaki, T. Okazaki, K. Sakabe, K. Sugimoto, and A. Sugino, "Mechanism of DNA chain growth. i. possible discontinuity and unusual secondary structure of newly synthesized chains," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 59, no. 2, pp. 598–605, Feb. 1968.

[19] A. S. Bhagwat, W. Hao, J. P. Townes, H. Lee, H. Tang, and P. L. Foster, "Strand-biased cytosine deamination at the replication fork causes cytosine to

thymine mutations in escherichia coli," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 113, no. 8, pp. 2176–2181, Feb. 2016.

[20] A. C. Frank and J. R. Lobry, "Asymmetric substitution patterns: a review of possible underlying mutational or selective mechanisms," *Gene*, vol. 238, no. 1, pp. 65–77, Sep. 1999.

[21] M. Picardeau, J. R. Lobry, and B. J. Hinnebusch, "Physical mapping of an origin of bidirectional replication at the centre of the borrelia burgdorferi linear chromosome," *Mol. Microbiol.*, vol. 32, no. 2, pp. 437–445, Apr. 1999.

[22] C. M. Fraser, S. Casjens, W. M. Huang, G. G. Sutton, R. Clayton, R. Lathigra, O. White, K. A. Ketchum, R. Dodson, E. K. Hickey, M. Gwinn, B. Dougherty, J. F. Tomb, R. D. Fleischmann, D. Richardson, J. Peterson, A. R. Kerlavage, J. Quackenbush, S. Salzberg, M. Hanson, R. van Vugt, N. Palmer, M. D. Adams, J. Gocayne, J. Weidman, T. Utterback, L. Watthey, L. McDonald, P. Artiach, C. Bowman, S. Garland, C. Fuji, M. D. Cotton, K. Horst, K. Roberts, B. Hatch, H. O. Smith, and J. C. Venter, "Genomic sequence of a lyme disease spirochaete, borrelia burgdorferi," *Nature*, vol. 390, no. 6660, pp. 580–586, Dec. 1997.

[23] F. R. Blattner, G. Plunkett, 3rd, C. A. Bloch, N. T. Perna, V. Burland, M. Riley, J. Collado-Vides, J. D. Glasner, C. K. Rode, G. F. Mayhew, J. Gregor, N. W. Davis, H. A. Kirkpatrick, M. A. Goeden, D. J. Rose, B. Mau, and Y. Shao,

"The complete genome sequence of escherichia coli K-12," *Science*, vol. 277, no. 5331, pp. 1453–1462, Sep. 1997.

[24] M. J. McLean, K. H. Wolfe, and K. M. Devine, "Base composition skews, replication orientation, and gene orientation in 12 prokaryote genomes," *J. Mol. Evol.*, vol. 47, no. 6, pp. 691–696, Dec. 1998.

[25] E. P. Rocha, A. Danchin, and A. Viari, "Universal replication biases in bacteria," *Mol. Microbiol.*, vol. 32, no. 1, pp. 11–16, Apr. 1999.

[26] G. Zhang and F. Gao, "Quantitative analysis of correlation between AT and GC biases among bacterial genomes," *PLoS One*, vol. 12, no. 2, p. e0171408, Feb. 2017.

[27] L.-X. Chen, K. Anantharaman, A. Shaiber, A. Murat Eren, and J. F. Banfield, "Accurate and complete genomes from metagenomes," *Genome Res*, p. 30, Mar. 2020.

[28] A. Gurevich, V. Saveliev, N. Vyahhi, and G. Tesler, "QUAST: quality assessment tool for genome assemblies," *Bioinformatics*, vol. 29, no. 8, pp. 1072–1075, Apr. 2013.

[29] M. Hunt, T. Kikuchi, M. Sanders, C. Newbold, M. Berriman, and T. D. Otto, "REAPR: a universal tool for genome assembly evaluation," *Genome Biol.*,

vol. 14, no. 5, p. R47, May 2013.

[30] X. Zhu, H. C. M. Leung, R. Wang, F. Y. L. Chin, S. M. Yiu, G. Quan, Y. Li, R. Zhang, Q. Jiang, B. Liu, Y. Dong, G. Zhou, and Y. Wang, "misfinder: identify mis-assemblies in an unbiased manner using reference and paired-end reads," *BMC Bioinformatics*, vol. 16, p. 386, Nov. 2015.

[31] A. L. Delcher, A. Phillippy, J. Carlton, and S. L. Salzberg, "Fast algorithms for large-scale genome alignment and comparison," *Nucleic Acids Res.*, vol. 30, no. 11, pp. 2478–2483, Jun. 2002.

[32] B. Langmead and S. L. Salzberg, "Fast gapped-read alignment with bowtie 2," *Nat. Methods*, vol. 9, no. 4, pp. 357–359, Mar. 2012.

[33] H. Long, W. Sung, S. Kucukyildirim, E. Williams, S. F. Miller, W. Guo, C. Patterson, C. Gregory, C. Strauss, C. Stone, C. Berne, D. Kysela, W. R. Shoemaker, M. E. Muscarella, H. Luo, J. T. Lennon, Y. V. Brun, and M. Lynch, "Evolutionary determinants of genome-wide nucleotide composition," *Nat Ecol Evol*, vol. 2, no. 2, pp. 237–240, Feb. 2018.

[34] C. A. Glaser, S. Gilliam, D. Schnurr, B. Forghani, S. Honarmand, N. Khetsuriani, M. Fischer, C. K. Cossen, L. J. Anderson, and California Encephalitis Project, 1998-2000, "In search of encephalitis etiologies: diagnostic challenges

in the california encephalitis project, 1998-2000," *Clin. Infect. Dis.*, vol. 36, no. 6, pp. 731–742, Mar. 2003.

[35] N. J. Loman, C. Constantinidou, M. Christner, H. Rohde, J. Z.-M. Chan, J. Quick, J. C. Weir, C. Quince, G. P. Smith, J. R. Betley, M. Aepfelbacher, and M. J. Pallen, "A culture-independent sequence-based metagenomics approach to the investigation of an outbreak of shiga-toxigenic escherichia coli O104:H4," *JAMA*, vol. 309, no. 14, pp. 1502–1510, Apr. 2013.

[36] H. Hasman, D. Saputra, T. Sicheritz-Ponten, O. Lund, C. A. Svendsen, N. Frimodt-Møller, and F. M. Aarestrup, "Rapid whole-genome sequencing for detection and characterization of microorganisms directly from clinical samples," *J. Clin. Microbiol.*, vol. 52, no. 1, pp. 139–146, Jan. 2014.

[37] M. R. Wilson, S. N. Naccache, E. Samayoa, M. Biagtan, H. Bashir, G. Yu, S. M. Salamat, S. Somasekar, S. Federman, S. Miller, R. Sokolic, E. Garabedian, F. Candotti, R. H. Buckley, K. D. Reed, T. L. Meyer, C. M. Seroogy, R. Galloway, S. L. Henderson, J. E. Gern, J. L. DeRisi, and C. Y. Chiu, "Actionable diagnosis of neuroleptospirosis by next-generation sequencing," *N. Engl. J. Med.*, vol. 370, no. 25, pp. 2408–2417, Jun. 2014.

[38] S. L. Salzberg, F. P. Breitwieser, A. Kumar, H. Hao, P. Burger, F. J. Rodriguez, M. Lim, A. Quiñones-Hinojosa, G. L. Gallia, J. A. Tornheim, M. T. Melia, C. L.

Sears, and C. A. Pardo, "Next-generation sequencing in neuropathologic diagnosis of infections of the nervous system," *Neurol Neuroimmunol Neuroinflamm*, vol. 3, no. 4, p. e251, Aug. 2016.

[39] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool," *J. Mol. Biol.*, vol. 215, no. 3, pp. 403–410, Oct. 1990.

[40] D. A. Benson, K. Clark, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and E. W. Sayers, "GenBank," *Nucleic Acids Res.*, vol. 43, no. Database issue, pp. D30–5, Jan. 2015.

[41] R. I. Amann, W. Ludwig, and K. H. Schleifer, "Phylogenetic identification and in situ detection of individual microbial cells without cultivation," *Microbiol. Rev.*, vol. 59, no. 1, pp. 143–169, Mar. 1995.

[42] R. Daniel, "The metagenomics of soil," *Nat. Rev. Microbiol.*, vol. 3, no. 6, pp. 470–478, Jun. 2005.

[43] C. Schleper, E. F. DeLong, C. M. Preston, R. A. Feldman, K. Y. Wu, and R. V. Swanson, "Genomic analysis reveals chromosomal variation in natural populations of the uncultured psychrophilic archaeon cenarchaeum symbiosum," *J. Bacteriol.*, vol. 180, no. 19, pp. 5003–5009, Oct. 1998.

[44] L. Butinar, I. Spencer-Martins, and N. Gunde-Cimerman, “Yeasts in high arctic glaciers: the discovery of a new habitat for eukaryotic microorganisms,” *Antonie Van Leeuwenhoek*, vol. 91, no. 3, pp. 277–289, Apr. 2007.

[45] P. Hugenholtz, “Exploring prokaryotic diversity in the genomic era,” *Genome Biol.*, vol. 3, no. 2, p. REVIEWS0003, Jan. 2002.

[46] D. M. Karl, “Hidden in a sea of microbes,” *Nature*, vol. 415, no. 6872, pp. 590–591, Feb. 2002.

[47] J. A. Eisen, “Environmental shotgun sequencing: its potential and challenges for studying the hidden world of microbes,” *PLoS Biol.*, vol. 5, no. 3, p. e82, Mar. 2007.

[48] K. Haldar, S. Kamoun, N. L. Hiller, S. Bhattacharje, and C. van Ooij, “Common infection strategies of pathogenic eukaryotes,” *Nat. Rev. Microbiol.*, vol. 4, no. 12, pp. 922–931, Dec. 2006.

[49] J. L. Jones, D. Kruszon-Moran, M. Wilson, G. McQuillan, T. Navin, and J. B. McAuley, “Toxoplasma gondii infection in the united states: seroprevalence and risk factors,” *Am. J. Epidemiol.*, vol. 154, no. 4, pp. 357–365, Aug. 2001.

[50] P. A. Thomas, “Fungal infections of the cornea,” *Eye*, vol. 17, no. 8, pp. 852–862, Nov. 2003.

[51] J. Y. Niederkorn, H. Alizadeh, H. Leher, and J. P. McCulley, “The pathogenesis of acanthamoeba keratitis,” *Microbes Infect.*, vol. 1, no. 6, pp. 437–443, May 1999.

[52] C. Aurrecoechea, A. Barreto, E. Y. Basenko, J. Brestelli, B. P. Brunk, S. Cade, K. Crouch, R. Doherty, D. Falke, S. Fischer, B. Gajria, O. S. Harb, M. Heiges, C. Hertz-Fowler, S. Hu, J. Iodice, J. C. Kissinger, C. Lawrence, W. Li, D. F. Pinney, J. A. Pulman, D. S. Roos, A. Shanmugasundram, F. Silva-Franco, S. Steinbiss, C. J. Stoeckert, Jr, D. Spruill, H. Wang, S. Warrenfeltz, and J. Zheng, “EuPathDB: the eukaryotic pathogen genomics database resource,” *Nucleic Acids Res.*, vol. 45, no. D1, pp. D581–D591, Jan. 2017.

[53] S. Merchant, D. E. Wood, and S. L. Salzberg, “Unexpected cross-species contamination in genome sequencing projects,” *PeerJ*, vol. 2, p. e675, Nov. 2014.

[54] A. Morgulis, G. Coulouris, Y. Raytselis, T. L. Madden, R. Agarwala, and A. A. Schäffer, “Database indexing for production MegaBLAST searches,” *Bioinformatics*, vol. 24, no. 16, pp. 1757–1764, Aug. 2008.

[55] U. Böhme, T. D. Otto, J. A Cotton, S. Steinbiss, M. Sanders, S. O. Oyola, A. Nicot, S. Gandon, K. P. Patra, C. Herd, E. Bushell, K. K. Modrzynska, O. Billker, J. M. Vinetz, A. Rivero, C. I. Newbold, and M. Berriman, “Complete avian malaria parasite genomes reveal features associated with lineage-specific

evolution in birds and mammals," *Genome Res.*, Mar. 2018.

[56] F. Prugnolle, P. Durand, C. Neel, B. Ollomo, F. J. Ayala, C. Arnathau, L. Etienne, E. Mpoudi-Ngole, D. Nkoghe, E. Leroy, E. Delaporte, M. Peeters, and F. Renaud, "African great apes are natural hosts of multiple related malaria species, including plasmodium falciparum," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 107, no. 4, pp. 1458–1463, Jan. 2010.

[57] Z. Li, F. P. Breitwieser, J. Lu, A. S. Jun, L. Asnaghi, S. L. Salzberg, and C. G. Eberhart, "Identifying corneal infections in Formalin-Fixed specimens using next generation sequencing," *Invest. Ophthalmol. Vis. Sci.*, vol. 59, no. 1, pp. 280–288, Jan. 2018.

[58] L. Brocchieri, "Low-complexity regions in plasmodium proteins: in search of a function," *Genome Res.*, vol. 11, no. 2, pp. 195–197, Feb. 2001.

[59] M. M. Zilversmit, S. K. Volkman, M. A. DePristo, D. F. Wirth, P. Awadalla, and D. L. Hartl, "Low-complexity regions in plasmodium falciparum: missing links in the evolution of an extreme genome," *Mol. Biol. Evol.*, vol. 27, no. 9, pp. 2198–2209, Sep. 2010.

[60] D. A. Relman, T. M. Schmidt, A. Gajadhar, M. Sogin, J. Cross, K. Yoder, O. Sethabutr, and P. Echeverria, "Molecular phylogenetic analysis of cy-

clospora, the human intestinal pathogen, suggests that it is closely related to eimeria species," *J. Infect. Dis.*, vol. 173, no. 2, pp. 440–445, Feb. 1996.

[61] R. Fayer, "Epidemiology of protozoan infections: The coccidia," *Vet. Parasitol.*, vol. 6, no. 1, pp. 75–103, Jan. 1980.

[62] L. P. Joyner, "Experimental eimeria mitis infections in chickens," *Parasitology*, vol. 48, no. 1-2, pp. 101–112, May 1958.

[63] W. B. Paterson and S. S. Desser, "The biology of two eimeria species (protista: Apicomplexa) in their mutual fish hosts in ontario," *Can. J. Zool.*, vol. 60, no. 5, pp. 764–775, May 1982.

[64] T. Garnier, K. Eiglmeier, J.-C. Camus, N. Medina, H. Mansoor, M. Pryor, S. Duthoy, S. Grondin, C. Lacroix, C. Monsempe, S. Simon, B. Harris, R. Atkin, J. Doggett, R. Mayes, L. Keating, P. R. Wheeler, J. Parkhill, B. G. Barrell, S. T. Cole, S. V. Gordon, and R. G. Hewinson, "The complete genome sequence of mycobacterium bovis," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 100, no. 13, pp. 7877–7882, Jun. 2003.

[65] J. M. Grange, "Mycobacterium bovis infection in human beings," *Tuberculosis*, vol. 81, no. 1-2, pp. 71–77, 2001.

[66] R. Brosch, S. V. Gordon, M. Marmiesse, P. Brodin, C. Buchrieser, K. Eiglmeier,

T. Garnier, C. Gutierrez, G. Hewinson, K. Kremer, L. M. Parsons, A. S. Pym, S. Samper, D. van Soolingen, and S. T. Cole, “A new evolutionary scenario for the mycobacterium tuberculosis complex,” *Proc. Natl. Acad. Sci. U. S. A.*, vol. 99, no. 6, pp. 3684–3689, Mar. 2002.

[67] E. Helgason, O. A. Okstad, D. A. Caugant, H. A. Johansen, A. Fouet, M. Mock, I. Hegna, and A. B. Kolstø, “Bacillus anthracis, bacillus cereus, and bacillus thuringiensis–one species on the basis of genetic evidence,” *Appl. Environ. Microbiol.*, vol. 66, no. 6, pp. 2627–2630, Jun. 2000.

[68] Y. Liu, Q. Lai, M. Göker, J. P. Meier-Kolthoff, M. Wang, Y. Sun, L. Wang, and Z. Shao, “Genomic insights into the taxonomic status of the bacillus cereus group,” *Sci. Rep.*, vol. 5, p. 14082, Sep. 2015.

[69] R. Lan and P. R. Reeves, “Escherichia coli in disguise: molecular origins of shigella,” *Microbes Infect.*, vol. 4, no. 11, pp. 1125–1132, Sep. 2002.

[70] M. A. Peabody, T. Van Rossum, R. Lo, and F. S. L. Brinkman, “Evaluation of shotgun metagenomics sequence classification methods using in silico and in vitro simulated communities,” *BMC Bioinformatics*, vol. 16, p. 363, Nov. 2015.

[71] T. Thiel, B. S. Pratte, J. Zhong, L. Goodwin, A. Copeland, S. Lucas, C. Han, S. Pitluck, M. L. Land, N. C. Kyrpides, and T. Woyke, “Complete genome

sequence of anabaena variabilis ATCC 29413," *Stand. Genomic Sci.*, vol. 9, no. 3, pp. 562–573, Jun. 2014.

[72] L. Schaeffer, H. Pimentel, N. Bray, P. Melsted, and L. Pachter, "Pseudoalignment for metagenomic read assignment," *Bioinformatics*, vol. 33, no. 14, pp. 2082–2088, Jul. 2017.

[73] F. E. Angly, D. Willner, A. Prieto-Davó, R. A. Edwards, R. Schmieder, R. Vega-Thurber, D. A. Antonopoulos, K. Barott, M. T. Cottrell, C. Desnues, E. A. Dinsdale, M. Furlan, M. Haynes, M. R. Henn, Y. Hu, D. L. Kirchman, T. McDole, J. D. McPherson, F. Meyer, R. M. Miller, E. Mundt, R. K. Naviaux, B. Rodriguez-Mueller, R. Stevens, L. Wegley, L. Zhang, B. Zhu, and F. Rohwer, "The GAAS metagenomic tool and its estimations of viral and microbial average genome size in four major biomes," *PLoS Comput. Biol.*, vol. 5, no. 12, p. e1000593, Dec. 2009.

[74] M. S. Lindner and B. Y. Renard, "Metagenomic abundance estimation and diagnostic testing on species level," *Nucleic Acids Res.*, vol. 41, no. 1, p. e10, Jan. 2013.

[75] C. Luo, R. Knight, H. Siljander, M. Knip, R. J. Xavier, and D. Gevers, "ConStrains identifies microbial strains in metagenomic datasets," *Nat. Biotechnol.*, vol. 33, no. 10, pp. 1045–1052, Oct. 2015.

[76] N. Segata, L. Waldron, A. Ballarini, V. Narasimhan, O. Jousson, and C. Huttenhower, “Metagenomic microbial community profiling using unique clade-specific marker genes,” *Nat. Methods*, vol. 9, no. 8, pp. 811–814, Jun. 2012.

[77] M. B. Sohn, L. An, N. Pookhao, and Q. Li, “Accurate genome relative abundance estimation for closely related species in a metagenomic sample,” *BMC Bioinformatics*, vol. 15, p. 242, Jul. 2014.

[78] L. C. Xia, J. A. Cram, T. Chen, J. A. Fuhrman, and F. Sun, “Accurate genome relative abundance estimation based on shotgun metagenomic reads,” *PLoS One*, vol. 6, no. 12, p. e27992, Dec. 2011.

[79] S. Lindgreen, K. L. Adair, and P. P. Gardner, “An evaluation of the accuracy and speed of metagenome analysis tools,” *Sci. Rep.*, vol. 6, p. 19233, Jan. 2016.

[80] D. R. Mende, A. S. Waller, S. Sunagawa, A. I. Järvelin, M. M. Chan, M. Arumugam, J. Raes, and P. Bork, “Assessment of metagenomic assembly using simulated next generation sequencing data,” *PLoS One*, vol. 7, no. 2, p. e31386, Feb. 2012.

[81] Human Microbiome Project Consortium, “Structure, function and diversity of the healthy human microbiome,” *Nature*, vol. 486, no. 7402, pp. 207–214, Jun. 2012.

[82] R. M. Bowers, A. Clum, H. Tice, J. Lim, K. Singh, D. Ciobanu, C. Y. Ngan, J.-F. Cheng, S. G. Tringe, and T. Woyke, "Impact of library preparation protocols and template quantity on the metagenomic reconstruction of a mock microbial community," *BMC Genomics*, vol. 16, p. 856, Oct. 2015.

[83] C. R. Woese, G. E. Fox, L. Zablen, T. Uchida, L. Bonen, K. Pechman, B. J. Lewis, and D. Stahl, "Conservation of primary structure in 16S ribosomal RNA," *Nature*, vol. 254, no. 5495, pp. 83–86, Mar. 1975.

[84] C. R. Woese, "Bacterial evolution," *Microbiol. Rev.*, vol. 51, no. 2, pp. 221–271, Jun. 1987.

[85] J. A. Gilbert, J. K. Jansson, and R. Knight, "The earth microbiome project: successes and aspirations," *BMC Biol.*, vol. 12, p. 69, Aug. 2014.

[86] N. Fierer, J. W. Leff, B. J. Adams, U. N. Nielsen, S. T. Bates, C. L. Lauber, S. Owens, J. A. Gilbert, D. H. Wall, and J. G. Caporaso, "Cross-biome metagenomic analyses of soil microbial communities and their functional attributes," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 109, no. 52, pp. 21 390–21 395, Dec. 2012.

[87] J. Rousk, E. Bååth, P. C. Brookes, C. L. Lauber, C. Lozupone, J. G. Caporaso, R. Knight, and N. Fierer, "Soil bacterial and fungal communities across a ph gradient in an arable soil," *ISME J.*, vol. 4, no. 10, pp. 1340–1351, Oct. 2010.

[88] J. E. Kostka, O. Prakash, W. A. Overholt, S. J. Green, G. Freyer, A. Canion, J. Delgardio, N. Norton, T. C. Hazen, and M. Huettel, "Hydrocarbon-degrading bacteria and the bacterial community response in gulf of mexico beach sands impacted by the deepwater horizon oil spill," *Appl. Environ. Microbiol.*, vol. 77, no. 22, pp. 7962–7974, Nov. 2011.

[89] A. Kopf, M. Bicak, R. Kottmann, J. Schnetzer, I. Kostadinov, K. Lehmann, A. Fernandez-Guerra, C. Jeanthon, E. Rahav, M. Ullrich, A. Wichels, G. Gerdts, P. Polymenakou, G. Kotoulas, R. Siam, R. Z. Abdallah, E. C. Sonnenschein, T. Cariou, F. O'Gara, S. Jackson, S. Orlic, M. Steinke, J. Busch, B. Duarte, I. Caçador, J. Canning-Clode, O. Bobrova, V. Marteinsson, E. Reynisson, C. M. Loureiro, G. M. Luna, G. M. Quero, C. R. Löscher, A. Kremp, M. E. DeLorenzo, L. Øvreås, J. Tolman, J. LaRoche, A. Penna, M. Frischer, T. Davis, B. Katherine, C. P. Meyer, S. Ramos, C. Magalhães, F. Jude-Lemeilleur, M. L. Aguirre-Macedo, S. Wang, N. Poulton, S. Jones, R. Collin, J. A. Fuhrman, P. Conan, C. Alonso, N. Stambler, K. Goodwin, M. M. Yakimov, F. Baltar, L. Bodrossy, J. Van De Kamp, D. M. Frampton, M. Ostrowski, P. Van Ruth, P. Malthouse, S. Claus, K. Deneudt, J. Mortelmans, S. Pitois, D. Wallom, I. Salter, R. Costa, D. C. Schroeder, M. M. Kandil, V. Amaral, F. Biancalana, R. Santana, M. L. Pedrotti, T. Yoshida, H. Ogata, T. Ingleton, K. Munnik, N. Rodriguez-Ezpeleta, V. Berteaux-Lecellier, P. Wecker, I. Cancio, D. Vaulot, C. Bienhold, H. Ghazal, B. Chaouni,

S. Essayeh, S. Ettamimi, E. H. Zaid, N. Boukhatem, A. Bouali, R. Chahboune, S. Barrijal, M. Timinouni, F. El Otmani, M. Bennani, M. Mea, N. Todorova, V. Karamfilov, P. Ten Hoopen, G. Cochrane, S. L'Haridon, K. C. Bizsel, A. Vezzi, F. M. Lauro, P. Martin, R. M. Jensen, J. Hinks, S. Gebbels, R. Rosselli, F. De Pascale, R. Schiavon, A. Dos Santos, E. Villar, S. Pesant, B. Cataletto, F. Malfatti, R. Edirisinghe, J. A. H. Silveira, M. Barbier, V. Turk, T. Tinta, W. J. Fuller, I. Salihoglu, N. Serakinci, M. C. Ergoren, E. Bresnan, J. Iriberri, P. A. F. Nyhus, E. Bente, H. E. Karlsen, P. N. Golyshin, J. M. Gasol, S. Moncheva, N. Dzhembekova, Z. Johnson, C. D. Sinigalliano, M. L. Gidley, A. Zingone, R. Danovaro, G. Tsiamis, M. S. Clark, A. C. Costa, M. El Bour, A. M. Martins, R. E. Collins, A.-L. Ducluzeau, J. Martinez, M. J. Costello, L. A. Amaral-Zettler, J. A. Gilbert, N. Davies, D. Field, and F. O. Glöckner, "The ocean sampling day consortium," *Gigascience*, vol. 4, p. 27, Jun. 2015.

[90] D. Bulgarelli, K. Schlaeppi, S. Spaepen, E. Ver Loren van Themaat, and P. Schulze-Lefert, "Structure and functions of the bacterial microbiota of plants," *Annu. Rev. Plant Biol.*, vol. 64, pp. 807–838, Jan. 2013.

[91] P. R. Hardoim, L. S. van Overbeek, G. Berg, A. M. Pirttilä, S. Compant, A. Campisano, M. Döring, and A. Sessitsch, "The hidden world within plants: Ecological and evolutionary considerations for defining functioning of microbial endophytes," *Microbiol. Mol. Biol. Rev.*, vol. 79, no. 3, pp. 293–320, Sep. 2015.

[92] J. A. Peiffer, A. Spor, O. Koren, Z. Jin, S. G. Tringe, J. L. Dangl, E. S. Buckler, and R. E. Ley, "Diversity and heritability of the maize rhizosphere microbiome under field conditions," *Proc. Natl. Acad. Sci. U. S. A.*, vol. 110, no. 16, pp. 6548–6553, Apr. 2013.

[93] J. B. Patel, "16S rRNA gene sequencing for bacterial pathogen identification in the clinical laboratory," *Mol. Diagn.*, vol. 6, no. 4, pp. 313–321, Dec. 2001.

[94] J. M. Janda and S. L. Abbott, "16S rRNA gene sequencing for bacterial identification in the diagnostic laboratory: pluses, perils, and pitfalls," *J. Clin. Microbiol.*, vol. 45, no. 9, pp. 2761–2764, Sep. 2007.

[95] J. E. Clarridge, 3rd, "Impact of 16S rRNA gene sequence analysis for identification of bacteria on clinical microbiology and infectious diseases," *Clin. Microbiol. Rev.*, vol. 17, no. 4, pp. 840–62, table of contents, Oct. 2004.

[96] A. D. Kostic, D. Gevers, H. Siljander, T. Vatanen, T. Hyötyläinen, A.-M. Hämäläinen, A. Peet, V. Tillmann, P. Pöhö, I. Mattila, H. Lähdesmäki, E. A. Franzosa, O. Vaarala, M. de Goffau, H. Harmsen, J. Ilonen, S. M. Virtanen, C. B. Clish, M. Orešič, C. Huttenhower, M. Knip, DIABIMMUNE Study Group, and R. J. Xavier, "The dynamics of the human infant gut microbiome in development and in progression toward type 1 diabetes," *Cell Host Microbe*, vol. 17, no. 2, pp. 260–273, Feb. 2015.

[97] D. C. Emery, D. K. Shoemark, T. E. Batstone, C. M. Waterfall, J. A. Coghill, T. L. Cerajewska, M. Davies, N. X. West, and S. J. Allen, "16S rRNA next generation sequencing analysis shows bacteria in alzheimer's Post-Mortem brain," *Front. Aging Neurosci.*, vol. 9, p. 195, Jun. 2017.

[98] A. Sivan, L. Corrales, N. Hubert, J. B. Williams, K. Aquino-Michaels, Z. M. Earley, F. W. Benyamin, Y. M. Lei, B. Jabri, M.-L. Alegre, E. B. Chang, and T. F. Gajewski, "Commensal bifidobacterium promotes antitumor immunity and facilitates anti-PD-L1 efficacy," *Science*, vol. 350, no. 6264, pp. 1084–1089, Nov. 2015.

[99] E. Y. Hsiao, S. W. McBride, S. Hsien, G. Sharon, E. R. Hyde, T. McCue, J. A. Codelli, J. Chow, S. E. Reisman, J. F. Petrosino, P. H. Patterson, and S. K. Mazmanian, "Microbiota modulate behavioral and physiological abnormalities associated with neurodevelopmental disorders," *Cell*, vol. 155, no. 7, pp. 1451–1463, Dec. 2013.

[100] Integrative HMP (iHMP) Research Network Consortium, "The integrative human microbiome project," *Nature*, vol. 569, no. 7758, pp. 641–648, May 2019.

[101] J. Kuczynski, E. K. Costello, D. R. Nemergut, J. Zaneveld, C. L. Lauber, D. Knights, O. Koren, N. Fierer, S. T. Kelley, R. E. Ley, J. I. Gordon, and R. Knight, "Direct sequencing of the human microbiome readily reveals com-

munity differences," *Genome Biol.*, vol. 11, no. 5, p. 210, May 2010.

[102] T. Z. DeSantis, P. Hugenholtz, N. Larsen, M. Rojas, E. L. Brodie, K. Keller, T. Huber, D. Dalevi, P. Hu, and G. L. Andersen, "Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB," *Appl. Environ. Microbiol.*, vol. 72, no. 7, pp. 5069–5072, Jul. 2006.

[103] C. Quast, E. Pruesse, P. Yilmaz, J. Gerken, T. Schweer, P. Yarza, J. Peplies, and F. O. Glöckner, "The SILVA ribosomal RNA gene database project: improved data processing and web-based tools," *Nucleic Acids Res.*, vol. 41, no. Database issue, pp. D590–6, Jan. 2013.

[104] J. R. Cole, Q. Wang, J. A. Fish, B. Chai, D. M. McGarrell, Y. Sun, C. T. Brown, A. Porras-Alfaro, C. R. Kuske, and J. M. Tiedje, "Ribosomal database project: data and tools for high throughput rRNA analysis," *Nucleic Acids Res.*, vol. 42, no. Database issue, pp. D633–42, Jan. 2014.

[105] Q. Wang, G. M. Garrity, J. M. Tiedje, and J. R. Cole, "Naive bayesian classifier for rapid assignment of rRNA sequences into the new bacterial taxonomy," *Appl. Environ. Microbiol.*, vol. 73, no. 16, pp. 5261–5267, Aug. 2007.

[106] R. C. Edgar, "Search and clustering orders of magnitude faster than BLAST," *Bioinformatics*, vol. 26, no. 19, pp. 2460–2461, Oct. 2010.

[107] E. Kopylova, L. Noé, and H. Touzet, “SortMeRNA: fast and accurate filtering of ribosomal RNAs in metatranscriptomic data,” *Bioinformatics*, vol. 28, no. 24, pp. 3211–3217, Dec. 2012.

[108] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, “Scikit-learn: Machine learning in python,” *J. Mach. Learn. Res.*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[109] T. Rognes, T. Flouri, B. Nichols, C. Quince, and F. Mahé, “VSEARCH: a versatile open source tool for metagenomics,” *PeerJ*, vol. 4, p. e2584, Oct. 2016.

[110] C. Camacho, G. Coulouris, V. Avagyan, N. Ma, J. Papadopoulos, K. Bealer, and T. L. Madden, “BLAST+: architecture and applications,” *BMC Bioinformatics*, vol. 10, p. 421, Dec. 2009.

[111] J. F. Matias Rodrigues, T. S. B. Schmidt, J. Tackmann, and C. von Mering, “MAPseq: highly efficient k-mer search with confidence estimates, for rRNA sequence analysis,” *Bioinformatics*, vol. 33, no. 23, pp. 3808–3810, Dec. 2017.

[112] P. D. Schloss, S. L. Westcott, T. Ryabin, J. R. Hall, M. Hartmann, E. B. Hollister, R. A. Lesniewski, B. B. Oakley, D. H. Parks, C. J. Robinson, J. W. Sahl, B. Stres, G. G. Thallinger, D. J. Van Horn, and C. F. Weber, “Introducing

mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities," *Appl. Environ. Microbiol.*, vol. 75, no. 23, pp. 7537–7541, Dec. 2009.

[113] J. Lu, F. P. Breitwieser, P. Thielen, and S. L. Salzberg, "Bracken: estimating species abundance in metagenomics data," *PeerJ Comput. Sci.*, vol. 3, p. e104, Jan. 2017.

[114] A. Almeida, A. L. Mitchell, A. Tarkowska, and R. D. Finn, "Benchmarking taxonomic assignments based on 16S rRNA gene profiling of the microbiota from commonly sampled environments," *Gigascience*, vol. 7, no. 5, May 2018.

[115] J. R. Bray and J. T. Curtis, "An ordination of the upland forest communities of southern wisconsin," *Ecol. Monogr.*, vol. 27, no. 4, pp. 325–349, Feb. 1957.

[116] M. Balvočiūtė and D. H. Huson, "SILVA, RDP, greengenes, NCBI and OTT - how do these taxonomies compare?" *BMC Genomics*, vol. 18, no. Suppl 2, p. 114, Mar. 2017.

# Vita

Jennifer Lu received the Bachelors of Science degree in Chemical and Biomolecular Engineering from Johns Hopkins University in December 2014, and enrolled in the Biomedical Engineering Ph.D. program at Johns Hopkins University in August 2015. He was inducted into the Tau Beta Pi Honor Society in 2014 and the Alpha Eta Mu Beta Honor Society in 2017. Her research focuses on the computational metagenomics of infectious disease diagnostics. Specifically, Jennifer Lu develops software for evaluating sequenced microbial environments and evaluates genomes to be used in identifying infectious diseases by sequencing analysis.