UNIVERSITY OF CALIFORNIA
RIVERSIDE

Computing the Microbiome: Faster, More Accurate and More Efficient Methods for the Analysis of Metagenomes

A Dissertation submitted in partial satisfaction
of the requirements for the degree of

Doctor of Philosophy

in

Computer Science

by

Rachid Ounit

March 2017

Dissertation Committee:

Dr. Stefano Lonardi, Chairperson
Dr. Timothy J. Close
Dr. Tao Jiang
Dr. Eamonn Keogh
Dr. Tamar Shinar

The Dissertation of Rachid Ounit is approved:

_____

_____

_____

_____

_____

Committee Chairperson

University of California, Riverside

Acknowledgments

I have been privileged to join the Ph.D program at the University of California, Riverside in 2012. For the past five years, I have been lucky to meet and work with exceptional people who have impacted my education and the quality of my research work.

I would like to take this occasion to address my deepest and sincerest gratitude to my advisor Dr. Stefano Lonardi for the inestimable guidance and unfailing encouragement. Since I have joined the Lonardi lab in the Summer 2013, my experience and knowledge in bioinformatics – and more generally, scientific research – have increased and gained invaluable lessons after each meeting or discussion with him. I have also learned thanks to him precious skills for working in teams, where people with various background, skills and expectations are gathered to accomplish challenging goals. I owe to Dr. Lonardi an immense debt for being an outstanding researcher and mentor to me.

I would like to address my gratitude to Dr. Timothy J. Close and all his colleagues. I have been more than honored to be working with him in all our collaborations. His insights and observations had always been invaluable and helped significantly to direct my research work. I also address my greatest gratitude to Dr. Tao Jiang, Dr. Eamonn Keogh and Dr. Tamar Shinar for their important advice and suggestions that strongly help to orientate and focus my research work since my qualification examination. In addition, my thanks go to my teachers of the Ph.D. program that greatly influenced my education, especially Dr. Stefano Lonardi, Dr. Tamar Shinar, Dr. Tao Jiang, Dr. Marek Chrobak, Dr. Neal Young, Dr. Chinya Ravishankar, and Dr. Christian Shelton.

My gratitude goes to Dr. Elena Y. Harris for inviting me to participate in the BRAT-NOVA project. This collaboration was a useful and significant learning experience.

Portions of the Chapter 2 have appeared in *BMC Genomics*, in an article entitled "CLARK: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers" co-authored with Timothy J. Close, Steve Wanamaker and Stefano Lonardi [Ounit et al., 2015].

Some material in Chapter 3 is a result of a collaboration with Lars Hahn, Chris-André Leimeister and Burkhard Morgenstern, and have appeared in *Plos Computational Biology*, in an article entitled "$rasbhari$: Optimizing Spaced Seeds for Database Searching, Read Mapping and Alignment-Free Sequence Comparison"[Hahn et al., 2016].

Some material in Chapter 3 is a result of a collaboration with Alexa B. R. McIntyre, Ebrahim Afshinnekoo, Robert Prill, Gail L. Rosen, Elizabeth Hénaff, Noah Alexander, Sofia Ahsanuddin, Scott Tighe, Rita R. Colwell, and Christopher E. Mason, and are part of a manuscript in preparation.

I am grateful to Dr. Christopher E. Mason for inviting me to join the Metagenomics and Metadesign of Subways and Urban Biomes (MetaSUB) group, where I was offered a great opportunity to learn new methodologies about microbiome analysis and sequencing technologies. His

feedback and advice were always helpful and strengthen my research work (for example, about the classifiers' evaluation regarding our *Bioinformatics* paper [Ounit and Lonardi, 2016], or for several material presented in Chapter 4).

I am also grateful to Dr. Niamh B. O'Hara for inviting me in her nation-wide study focusing on the microbiome of ambulances. It has also been an immense privilege to be working with her every day during my internship in the Summer 2016 and for developing new methodologies and Bioinformatics tools.

I would like to thank all the members of the Department of Computer Science and Engineering of the University. Amy Ricks, Vanda Yamaguchi, Madie Heersink, Alicia Serrano, Kristina Fernandez, and Isaac Owusu-Frimpong have all been kind and patient to me and made my experience at the Department a great and enjoyable one. I will always admire their dedication and commitment to all the students of the department they help everyday.

Finally, I would like to address a warm and special thought to some important people in my life, without them my Ph.D experience would not have had the same taste. To Jeremie, Adrien, Krista, Adam, Oriane and Dasha: your support and encouragements over the years show that you are truly the best friends anyone can ask for.

To my parents, Khadija and Lahcen, my sisters, Rabiaa, Rkia, Naïma and my brother,

Youssoufe, for all the constant care, love and support.

ABSTRACT OF THE DISSERTATION

Computing the Microbiome: Faster, More Accurate and More Efficient Methods for the Analysis of Metagenomes

by

Rachid Ounit

Doctor of Philosophy, Graduate Program in Computer Science
University of California, Riverside, March 2017
Dr. Stefano Lonardi, Chairperson

Metagenomics is revolutionizing microbial ecology and has unlocked unprecedented opportunities in many domains of Life Science. For instance, metagenomics has allowed the discovery of new forms of life in unexplored habitats (e.g., in the marine environment). In medicine, metagenomics is allowing doctors to diagnose and help patients that have diseases related to imbalances in their microbial communities (e.g., gastrointestinal microbiota). In public health, metagenomics is becoming an invaluable instrument for pathogen surveillance and to monitor outbreaks in epidemic areas.

As sequencing technologies have considerably improved in speed and cost over the past decade, the number of reference sequences in public databases has grown exponentially. As a consequence, faster, accurate and efficient computational methods are needed for analyzing these large data. The research presented in this dissertation focuses on (i) how to build faster, more accurate and more efficient sequence classification methods to determine the microbial composition of metagenomic samples and (ii) how to infer and recover the microbial composition of a sample in a large network of connected samples (e.g., in the context of a city-scale biosurveillance).

Our classification system is composed of a family of tools, namely CLARK, CLARK-l and CLARK-S, which are currently used by several research teams worldwide for metagenomics and genomics analysis. While CLARK is able to perform with high accuracy sequence classification and unprecedented speed, CLARK-S achieves the same precision and a much higher accuracy than CLARK, at a cost of a slightly slower speed.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

> It's clear to me that if you wiped all multicellular life forms off the face of the Earth, microbial life might shift a tiny bit. [...] If microbial life were to disappear, that would be it – instant death for the planet.
>
> Carl R. Woese

> With improved methods for analysis, funding stimulated by recent triumphs in the field, and attraction of diverse scientists to identify new problems and solve old ones, metagenomics will expand and continue to enrich our understanding of microorganisms.
>
> Jo Handelsman

The microbiome is defined as "a characteristic microbial community occupying a reasonably well defined habitat which has distinct physio-chemical properties. The term thus not only refers to the microorganisms involved but also encompasses their theatre of activity." [Whipps et al., 1988]. Microorganisms live everywhere in the biosphere: they exist in areas of extreme conditions, in oceans [Venter et al., 2004], urban areas (e.g., such as houses [Adams et al., 2015], city parks [Reese et al., 2015], subway system [Afshinnekoo et al., 2015, Hsu et al., 2016]), soil [Fierer et al.,

2012] and even in space [Vaishampayan et al., 2013]. Naturally, the human microbiome is of great importance. It is estimated that 100 trillion microbial cells live on and inside the human body (e.g., mouth, skin, gut, etc.) [Ley et al., 2006]. Understanding the interactions of the human cells and these non-human organisms is vital.

In 2008 the "Human Microbiome Project" (HMP) was initiated by the United States National Institutes of Health to identify and characterize microorganisms which are found in association with both healthy and diseased humans. HMP had a budget of $115 million for 5 years. HMP-funded researchers produced about 200 peer-reviewed articles. Several important discoveries were made. For example, microbes contribute more genes responsible for the human survival than humans' own genes [Qin et al., 2010]. HMP researchers also created a large public repository of microbial genomes and human data samples [Human Microbiome Project Consortium , 2012, Consortium et al., 2012], which is critically important to understand how the human microbiome and the human host interact. While the microbes inside our body play an active role in our health, microbes surroundings us, in urban areas or nature are likely to interact and impact our life as well. In the context of epidemiology, understanding the interaction between our microbiome and other microbiomes in the environment represents a major step to understand, cure and prevent the propagation of infectious diseases. Indeed, about 60% of emerging infectious diseases in humans are caused by zoonotic pathogens [Jones et al., 2008], (e.g., West Nile virus, avian influenza, or Ebola). The highest concentration of infectious diseases per million square kilometres of land was found between 30 and 60 degrees North and between 30 and 40 degrees South (including regions such as Europe, Japan, Southeastern Australia and Northeastern US). The environment influences infectious diseases but can also play a role in the resistance of bacteria to currents treatments (e.g., antibiotics).

Some of the resistance genes in pathogens have evolved in response to the heavy use of antibiotics and other interactions with the environment [Martínez, 2008][Allen et al., 2009]. The environment is a source of resistance genes for lateral gene transfer in bacteria, many of which have never been seen in human-associated bacteria [Martiny et al., 2011].

Because the majority of the human population lives in urban areas, the study of the urban microbiome is critical for understanding how the microbial environment can affect our health. The "Metagenomics and Metadesign of the Subways and Urban Biomes" Consortium (MetaSUB) (`http://metasub.org/`) have been created recently with the aim to improve city utilization and planning through the detection, measurement, and design of metagenomics within urban environments [Consortium et al., 2016]. The data produced by MetaSub is expected to lead to the discovery of global maps of antimicrobial resistance genes/markers. Given the importance of microbiomes and their impact in public health, in May 2016 the White House Office of Science and Technology Policy announced the "National Microbiome Initiative" (NMI) with the aim "to advance understanding of microbiomes in order to aid in the development of useful applications in areas such as health care, food production and environmental restoration" with federal investments of more than $121 million [NMI, 2016].

At the core of the NMI is metagenomics, the gold standard methodology for the analysis of environmental samples. Metagenomics is the culture-independent sequencing and analysis of all DNA recovered from a sample. Unlike traditional procedures, for example performing multiple targeted assays, each looking for a specific pathogen or organism, laboratories can use a single sequencing based test that is able to identify all microorganisms in a sample without the need for culture [Handelsman, 2004]. With metagenomics, once an environmental sample is sequenced, one

of the first task is to determine the identities of the microbial species present in the sample.

When we started this work in the Summer of 2013, the most common and accurate approach was to compare all sequenced DNA fragments (called *reads*) by sequence-alignment (e.g., MegaBLAST [Zhang et al., 2000]) against the database of reference genomes and then label the identified reads at the lowest taxonomy rank possible [Huson et al., 2007, 2011]. In [Qin et al., 2010], MEGAN [Huson et al., 2007], another sequence-alignment based tool, was used for the taxonomic classification. The underlying computational problem is equivalent to the string matching problem (i.e., queries of strings or short text are compared against a database of texts). Metagenomics has been used in the HMP, but also in the context of clinical diagnosis [Seth-Smith et al., 2013], pathogen detection in urban spaces [Nicholas et al., 2015] and more (e.g., see the review in [Miller et al., 2013]). However, unlike the HMP, the NMI is benefiting from improved technologies such as the mobile and "real-time" sequencing machines, like the the MinION by Oxford Nanopore Technologies. While other sequencing technologies (e.g. Illumina sequencing machines) are static and require laboratory equipments, the MinION sequencer has about the same dimensions than a USB stick, and only needs a laptop to run [Gardy et al., 2015]. These portable sequencers allow us to imagine a future of "real-time" biosurveillance [Gardy et al., 2015], [Quick et al., 2016].

Nowadays, the sequencing cost is very low (about $150 to sequence a sample) and modern sequencing machines can sequence billion of nucleotides in few days. As the number of known bacterial, archaeal, eukaryotic and viral genomes is exponentially growing, the analysis of these massive datasets is a computational challenge. Because a sequenced sample can contain millions of reads and that a database can contain thousands of reference genomes, approaches based on sequence alignment are too slow. Accurate, ultra-fast and efficient methods are needed to perform

as these analyses. This is particularly critical in time-sensitive scenarios, e.g., when we are dealing with clinical diagnoses or the safety of civilians (e.g., for Ebola surveillance [Quick et al., 2016]).

This dissertation aims to describe new computational methods for the taxonomic classification of metagenomic reads, and for the prediction of the microbial population in the context of missing data. The dissertation is organized as follow:

- Chapter 2: I describe a new versatile, accurate, efficient and ultra-fast sequence classification method called "CLARK". CLARK is an alignment-free sequence comparison based on "discriminative" k-mers sets (i.e., k-mers that belong specifically to a group of sequence or taxon). In metagenomics, we show that CLARK, at the genus and species level, is as accurate as the best state-of-the-art methods. However, compared to its closest competitor, CLARK is five times faster, in its fastest mode of execution and single-threaded. The manuscript describing CLARK was published in BMC GENOMICS [Ounit et al., 2015] more than a year ago and to the best of our knowledge, it is still the fastest metagenomic classifier among all published read-level classifiers. Its performance has also been acknowledged by several independent and international research groups such as MetaSUB or the Metagenomic Research Group from the Association of Biomolecular Resource facilities (ABRF/MGRG) for the "Extreme Microbiome Project"[1], in which CLARK is used as a standard method in their bioinformatics analysis. Finally, we also show that CLARK can be used in the analysis of genomic data, specifically on the barley genome. We showed it was able to classify barley BAC clones and unigenes with higher speed and accuracy than the state-of-the-art approach [Muñoz-Amatriaín et al., 2015].

[1]The Extreme Microbiome Project focuses on developing and surveying metagenomic methods to help facilitate the recovery of DNA and RNA from unique sample types (in areas with extreme conditions such as deep ocean, Arctic zones, etc.) as well as develop bioinformatics tools for *de novo* assembly `http://extrememicrobiome.org/`

- Chapter 3: I describe a new classifier, derived from the CLARK framework, based on discriminative spaced k-mers (instead of discriminative contiguous k-mers). The new tool based on spaced k-mers called CLARK-S was tested on several simulated and real data and showed higher accuracy than that of CLARK at the genus and phylum level [Ounit and Lonardi, 2015]. CLARK-S was presented in September 2015 at the 15th International Workshop for Algorithms in Bioinformatics (WABI'15) in Atlanta, GA. After the conference, I carried out additional research to demonstrate the performance of CLARK-S at a lower taxonomic rank and through a large scale. Indeed, at the species-level, strong evidence is needed to demonstrate the tool's accuracy because of the high similarity between species [Stackebrandt and Goebel, 1994, Mende et al., 2013]. The final version of CLARK-S was published in BIOINFORMATICS [Ounit and Lonardi, 2016].

- Chapter 4: In the context of the pathogen surveillance at a city-scale, I present how to take advantage of CLARK-S' results as well as a Bayesian model to accurately predict the microbes present in a site of interest for which samples cannot be collected or samples are lost/missing/contaminated. Our first results indicate that sites which are close to each other show similar/correlated microbial composition. We asked ourselves if it is possible to infer unknown microbes population of a given site using the known composition of its surroundings. Would it be possible to minimize the amount of sites to continuously monitor the microbial composition without significant loss of detection and reduce the overall biosurveillance cost of multiple sites? I have preliminary results indicating these questions can be answered affirmatively.

Chapter 2

CLARK, a faster and more precise sequence classification method

The Nature's book is written in mathematical language and its symbols are triangles, circles and other geometrical figures, without whose help it is impossible to comprehend a single word of it.

Galileo Galilei

The first rule was never to accept anything as true unless I recognized it to be evident [...]. The second was to divide each of the difficulties which I encountered into as many parts as possible, and as might be required for an easier solution. The third was to think in an orderly fashion, beginning with the things which were simplest and easiest to understand, and gradually and by degrees reaching toward more complex knowledge, even treating as though ordered materials which were not necessarily so. The last was always to make enumerations so complete, and reviews so general, that I would be certain that nothing was omitted.

René Descartes

2.1 Introduction

The classification problem of determining the origin of a given DNA sequence (e.g., a read or a transcript) in a given set of target sequences (e.g., a set of known genomes) is common to several fields of computational molecular biology.

In metagenomics, the objective is to study the composition of microbial community in an environmental sample. For example, sequencing of seawater samples has enabled discoveries in microbial diversity in the marine environment [Venter et al., 2004]. Similarly, the study of samples from the human body has elucidated the symbiotic relationships between the human microbiome and human health [Human Microbiome Project Consortium , 2012, Consortium et al., 2012]. Once a metagenomic sample is sequenced, the first task is to determine the identities of the microbial species present in the sample.

In the Summer 2013[1], several tools were already available to classify metagenomic reads against known bacterial genomes via sequence-alignment (e.g., Megan [Huson et al., 2007], [Liu et al., 2011], MetaPhlAn [Segata et al., 2012] or PhymmBL [Brady and Salzberg, 2011]) or sequence composition (e.g., PhyloPythiaS [Patil et al., 2011], NBC [Rosen et al., 2011] or LMAT [Ames et al., 2013]). The main comparative evaluation of these tools at the time was [Bazinet and Cummings, 2012]. In that work, the authors demonstrated that NBC exhibits the highest accuracy and sensitivity at the genus level among state-of-the-art methods such as Megan, PhymmBL, MetaPhyler and PhyloPhythiaS. This study also showed that NBC and other probabilistic methods (e.g., PhymmBL) as well BLAST-based methods (e.g., Megan, MetaPhyler) are computationally expensive. In 2014, a faster method called Kraken was introduced, but it did not achieve the sensitivity of

[1]This is at this time that I have joined the Lonardi lab and started focusing on the topic of sequence classification in Bioinformatics.

NBC [Wood and Salzberg, 2014]. At that time, we observed that no tool had a sensitivity comparable to NBC and a speed comparable to Kraken. In addition, few tools provided confidence scores about their sequence assignment, which can be useful for downstream analysis; also some tools were less user-friendly than others (e.g., dependencies with Jellyfish [Marçais and Kingsford, 2011] for Kraken or sequence aligners for Megan, etc.), which did not facilitate its usage.

Another application of sequence classification is associated with *de novo* clone-by-clone sequencing and assembly. Given a BAC clone (or a transcript), the classification problem is to determine which chromosome (or chromosome arm) is the most likely origin of that clone/transcript. The problem assumes that reads for each BAC/transcript as well as reads for each chromosome arm are available, but that the fully-assembled reference genome is not. This was the situation in barley, an organism whose complete genome is yet to be produced. The BAC/transcript assignment problem was usually addressed using general-purpose alignment tools (e.g., BLAST [Altschul et al., 1990] or BLAT [Kent, 2002]), as in described in [Lonardi et al., 2013].

Observe that in both of these applications the computational problem is the same: given a set of DNA sequences to be classified (henceforth called "objects") and a set of reference sequences (e.g., genus-level sequences, chromosome arms, etc., henceforth called "targets"), identify which target is the most likely origin of each object based on sequence similarity. Although this problem has been extensively studied, it is still computationally challenging due to the rapid advances in sequencing technologies: cheaper, faster, sequencing instruments can now generate billion of reads in a few days [Levy and Myers, 2016, Goodwin et al., 2016]. As the number of objects grows, so does the number of targets, as demonstrated by the exponential growth of GenBank [Benson et al., 2012] of the National Center for Biotechnology Information (NCBI) database, see Figure 2.1.

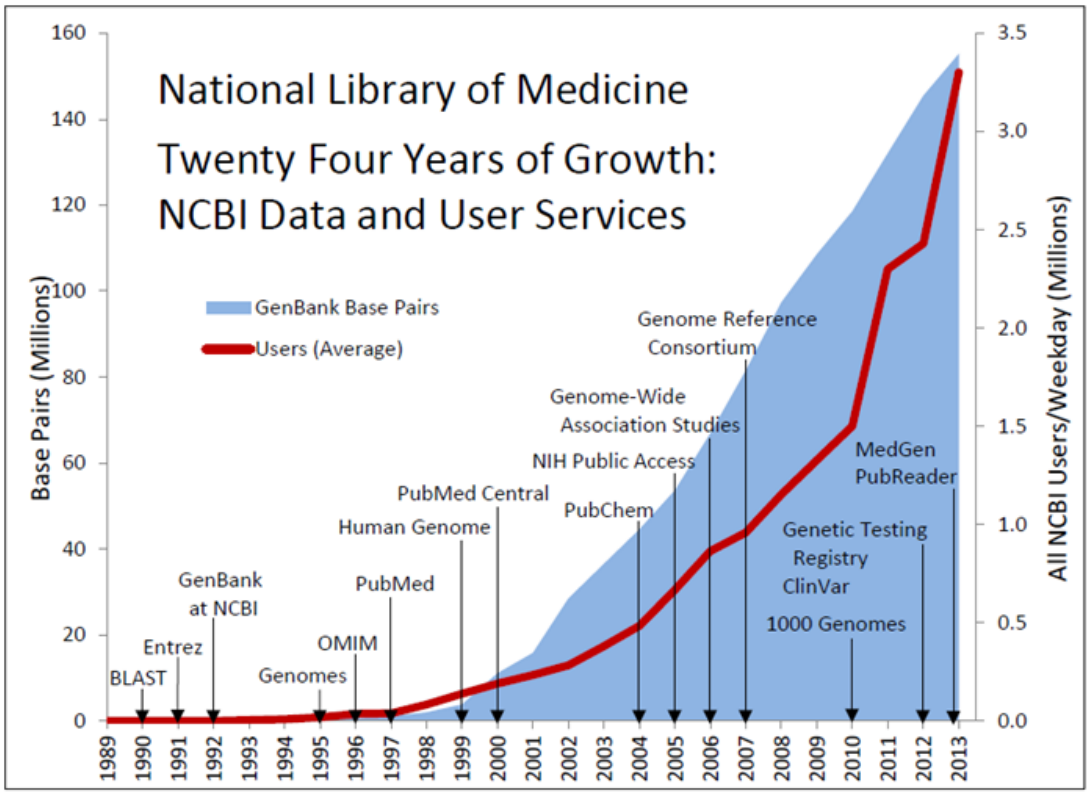

Figure 2.1: Data for twenty four year of growth: National Center for Biotechnology Information (NCBI) data and User services. (Resource: NCBI website)

Given these demands, it is critical for software tools to minimize computational resources (time, memory, I/O, etc.) required for analysis. This is why we introduced a new classification system called CLARK (CLAssifier based on Reduced K-mers), which can accurately and efficiently classify objects to targets, based on reduced sets of k-mers (i.e., DNA words of length k). CLARK is the first method able to perform classification of short metagenomics reads at the genus/species level with a sensitivity comparable to that of NBC, while achieving a comparable speed to Kraken. In several situations, CLARK is faster and more precise than Kraken at the genus/species level. Unlike tools like LMAT, MetaPhylAn, PhyloPythiaS, MetaPhyler or NBC, CLARK produces assignments with confidence scores, which are critical to post-process assignments in downstream analyses. Ad-

ditionally, CLARK is designed to be user-friendly, self-contained (i.e., does not depend on any other tool or library), and multi-core-friendly. CLARK does not need as much disk space as Kraken or PhymmBL. Finally, a "RAM-light" version of CLARK can be run on a memory-limited architecture (such as a 4 GB RAM laptop).

2.2 Methodology

The following sections describe the sequence classification method implemented in the CLARK framework.

2.2.1 The philosophy of lightweight algorithm

The philosophy of "lightweight" algorithms considers that only few rules and steps should characterize a program or algorithm. In situations where there is a massive amount of data, whether it is in the reference genomes[2] or in sequenced samples, the problem of comparing sequences based on similarity is intractable using naive or exhaustive approaches. While we are facing a continuous deluge of data, we believe that lightweight algorithms can become standards solutions if they can be fast, efficient and scalable. In the context of our system, we translate this philosophy into:

- The design and use of simple and efficient data-structures (e.g., Hash-tables).
- The preference of the basic/elementary operations (e.g., addition, multiplication, etc.) over complex non-linear operations that are time-consuming (e.g., matrix inversion, etc.).
- The systematic use of data reduction techniques (e.g., *Principal Component Analysis*, *Multiple Correspondence Analysis*, etc.) or approximation algorithms, whether it is for the

[2]The size of all complete genomes in NCBI/RefSeq is higher than 700 GBytes of data in August 2016.

database/input processing or the sequence classification.

2.2.2 Notations

Object and targets are described by their *sequence* which is a non-empty string over the alphabet $\Sigma = \{A, T, G, C\}$ of *nucleotides* (U can replace T in the case of mRNA). Observe that two bits are sufficient to identify a nucleotide. Given a sequence s, we use $s[i]$ to denote the i^{th} letter in s, and $|s|$ to denote its length. A *target* is a sequence representing either a genome, or a species (i.e., a set of genomes from different individuals), or a genus (i.e., a set of genomes). We use the variable n to indicate the number of targets. An *object* is a sequence that is assumed to originate from at most one of the n targets. We use the variable p to indicate the number of objects.

We say that a non-empty object s *originates* from target g if sequence s is a substring of sequence g. Given n targets $\{g_1, g_2, \ldots, g_n\}$ we say that a sequence s is *specific* to target g_c (or g_c-*specific*) $1 \leq c \leq n$, if s is a substring of g_c and s is not a substring of any other target. We say that a sequence s is a *repeat* of $\{g_1, g_2, \ldots, g_n\}$ if it is a substring of more than one target.

Given a positive integer k, a k-*mer* is any sequence of k consecutive nucleotides. Given that $|\Sigma| = 4$, there is a total of 4^k possible k-mers, i.e., any k-mer can be then associated to a unique *dimension* ranging from 1 to 4^k. It is easy to observe that exactly $N - k + 1$ k-mers (distinct or not) can be extracted from a sequence of length N, when $k \leq N$.

The k-*spectrum* $T(s)$ of an object s is the vector of size 4^k defined as follows: for any $1 \leq i \leq 4^k$, $T(s)_i$ is the number of occurrences in s of the k-mer with dimension i. Now consider $(E_k, ||.||_1)$, where $E_k = \mathbb{R}^{4^k}$ is a normed vector space of dimension 4^k and $||.||_1$ is the 1-norm. Although spectrums are vectors of integers, it is more convenient to consider the set $\mathbb{R}$ rather than $\mathbb{Z}$ because the former is a field. Thus, $(E_k, < \cdot|\cdot >)$, where $< \cdot|\cdot >$ is the standard dot product, is an

Euclidean space, on which useful notions such as projection and orthogonality can be defined. If $\vec{e_i}$ is the *unit vector* (entry i equal to 1 and 0 everywhere else), then $(\vec{e}_1, \vec{e}_2, \ldots, \vec{e}_{4^k})$ is the *canonical basis* of E_k.

The 1-norm of a vector $\vec{v} \in E_k$ is defined as $||\vec{v}||_1 = \sum_{i=1}^{4^k} |v_i|$. Since E_k is a vector space of finite dimensions, all p-norms are equivalent in E_k. However, we prefer the 1-norm due some of its properties. For instance, for any k-spectrum $T(s)$ for a sequence s of length N, we have $||T(s)||_1 = N - k + 1$. In other words, sequences of same length have the same 1-norm.

2.2.3 Problem Definition

The assignment problem can be defined as follows. Given a set of targets $\{g_1, g_2, \ldots, g_n\}$, a set of objects $\{s_1, s_2, \ldots, s_p\}$, and a positive integer k, assign each object s_i to the target g_{c^*}, such that the number of g_c-specific k-mers contained for s_i is the highest (where ties are broken arbitrarily) for $c = c^*$, where $1 \leq c \leq n$.

2.2.4 Probability of Two Sequences to Share the Same k-spectrum

We first observe that the mapping between a sequence and its spectrum is not one-to-one (i.e., it is not invertible), because the spectrum ignores the order of k-mers in the sequence. The consequence is that two or more sequences can have the same spectrum. For example, the spectrums of all $N - 1$ circular rotations of a string of length N are identical to each other.

We now proceed to compute the probability that a pair of sequences (targets or objects) of length N share the same k-spectrum. The problem of recovering a sequence from a set of k-mers is one of the "flavors" of genome assembly. From the k-spectrum one can build the corresponding *de Bruijn graph* (nodes are k-mers and edges connect two nodes if the two corresponding k-mers have

a $k-1$ overlap). Any Eulerian path of this graph recovers one of sequences having such spectrum [Compeau et al., 2011]. Here we want to count the number of sequences with the same k-spectrum, which is equal to the number of distinct Eulerian paths in the corresponding de Bruijn graph. Given a sequence s, we call $B_{k,N}$ the set of distinct sequences of length N whose k-spectrum is $T(s)$. Then, $|B_{k,N}|$ is the number of Eulerian paths in the de Bruijn graph G_s built from the k-spectrum of s.

Let us consider a set D of sequences of length N and an integer k. Let s, s' be two sequences in D. The probability that s and s' have the same k-spectrum is

$$P\left(T(s) = T(s') | s \neq s'\right) = \frac{|B_{k,N} \cap D|}{|D| - 1}$$

Since we will be using spectrums for classification, we want the probability of a conflict to be as small as possible. However, $|B_{k,N} \cap D|$ is not easy to evaluate for a generic set D of sequences. We can compute this quantity when D is the set of all sequences of length N. In this case, $|D| = 4^N$ and $|B_{k,N} \cap D| = |B_{k,N}|$.

The quantity $|B_{k,N}|$ is an upper bound to the number of Eulerian paths in G_s for a sequence s of length N. Thus, we have $|B_{k,N}| \leq (N-k+1)4^{N-k-3} \cdot 3 \cdot 2 \cdot 1$, because there are at most $(N-k+1)$ possibilities for choosing the first k-mer, then at most four distinct k-mers for the second position, then at most four distinct k-mers for the third position, and so on and so forth. For the last three positions there are three, then two, and one k-mer. Thus,

$$P\left(T(s) = T(s') | s \neq s'\right) \leq \frac{(N-k+1)4^{N-k-3} \cdot 3 \cdot 2 \cdot 1}{4^N - 1} = \frac{2(N-k+1)}{4^{k+2}} \tag{2.1}$$

For instance, when N is small (say, $N = 1000$), and k=12, we can estimate that $P(T(s) = T(s')|s \neq s') \leq 10^{-5}$. If N is bigger (say, $N = 10^8$, which is the size of a small genome) and k = 12, then $P(T(s) = T(s')|s \neq s') \leq 0.7451$. For $N = 10^8$ and k = 19 we get $P(T(s) = T(s')|s \neq s') \leq 4.547\ 10^{-5}$, and for $N = 10^9$ and k = 19, we get $P(T(s) = T(s')|s \neq s') \leq 4.547\ 10^{-4}$.

Inequality 2.1 can be used to determine the value of k that will make the probability of a spectrum conflict small enough (given N). Recall that we assumed that D contains all possible sequences of length N. When $|D| << 4^N$, it is reasonable to assume that Inequality 2.1 still holds when k is large enough, since in this case $|B_{k,N} \cap D| = 0$ (e.g., consider the extreme case $N = k$).

2.2.5 Spectral decomposition

Now we describe how k-spectrums can be used to assign objects to targets. Given a target g_c, $1 \leq c \leq n$, let $T(g_c)$ be its k-spectrum. Henceforth, we assume that vectors $T(g_1), T(g_2), T(g_3), \ldots, T(g_n)$ are non-null and linearly independent, i.e., the determinant of the matrix obtained from these vector is not zero:

$$det\ [T(g_1), T(g_2), T(g_3), \ldots, T(g_n)] \neq 0 \tag{2.2}$$

This assumption is met in practice and it is sufficiently general due to the fact that sequences from distinct targets contain unique substrings. From Inequality 2.1, we can also choose k large enough so the probability of two distinct sequences to share the same spectrum is as small as needed.

Let B_c be the basis of unit vectors such that these unit vectors are associated to non-zero-count dimensions in the k-spectrum of $T(g_c)$, i.e., $B_c = (\vec{e_i})_{i \in I_c}$, where I_c = $\{i,\ i \in \{1, 2, 3, \ldots, 4^k\} \mid T(g_c).\vec{e_i} \neq 0\}$. Since B_c contains all non-null dimensions from $T(g_c)$, we can

define $E_k^c = span(B_c)$, which is the space described by linear combinations of the unit vectors in B_c. E_k^c represents the vector space associated to the k-spectrum of target g_c.

Next we build another basis, but only for target-specific k-mers. Let $\widetilde{B}_c$ be the basis of unit vectors corresponding to the set of dimension of non-zero counts in the k-spectrum of $T(g_c)$, which has at the same time, zero counts in the spectrum of other targets, i.e., $\widetilde{B}_c = (\vec{e_i})_{i \in \widetilde{I}_c}$, where $\widetilde{I}_c = \{i,\ i \in \{1, 2, 3, \ldots, 4^k\} \mid T(g_c).\vec{e_i} \neq 0$ and for all $c' \neq c$ we have $T(g_{c'}).\vec{e_i} = 0\}$. By the Equation 2.2, we have for all c, $\widetilde{B}_c \neq \emptyset$ (if for some c, $\widetilde{B}_c = \emptyset$, then we need to increase k). Therefore, we can define $\widetilde{E}_k^c = span(\widetilde{B}_c)$, which is the vector space built from all subspaces specific to E_k^c. $\widetilde{E}_k^c$ is called *target-specific k-mer space* of g_c or simply g_c-specific k-mer space.

If the targets are chromosome arm sequences then these definitions can be extended to infer centromeric regions (see [Ounit et al., 2015] and [Muñoz-Amatriaín et al., 2015] for more details).

2.2.6 Orthogonal decomposition

The vector spaces $\widetilde{E}_k^c$ allow a decomposition of the k-mer vector space E_k. This section explains the construction of this decomposition. First, we prove the fact that a k-mer from an object s cannot belong to more than one target specific k-mer space.

Claim 1. *For all $(c, c') \in \{1, \ldots, n\}^2$, $c \neq c'$, we have $\widetilde{E}_k^c \perp \widetilde{E}_k^{c'}$.*

Proof. By construction, for all $\vec{u} \in \widetilde{E}_k^c, \forall \vec{u'} \in \widetilde{E}_k^{c'}$, we have $\vec{u} = \sum_{i \in \widetilde{I}_c} u_i \vec{e_i}$ and $\vec{u'} = \sum_{i \in \widetilde{I}_{c'}} u'_i \vec{e_i}$. Then, $< \vec{u} \mid \vec{u'} >= \sum_{i \in \{1,2,3,\ldots,4^k\}} u_i u'_i = \sum_{i \in \widetilde{I}_c \cap \widetilde{I}_{c'}} u_i u'_i$. By definition of the basis, $\widetilde{I}_c \cap \widetilde{I}_{c'} = \emptyset$ because $c \neq c'$, so $< \vec{u} \mid \vec{u'} >= 0$. ■

Since we have established that spaces $\widetilde{E}_k^c$ are pairwise orthogonal, we can define $\widetilde{E}_k$ as

the vector space resulting from the direct sum of all $\widetilde{E}_k^c$, i.e.,

$$\widetilde{E}_k = \bigoplus_{c=1}^{n} \widetilde{E}_k^c \tag{2.3}$$

Since $\widetilde{E}_k$ contains non-null spaces, $\widetilde{E}_k$ is not a null space. Also, since E_k is an Euclidean space and $\widetilde{E}_k \subset E_k$, we can define the orthogonal decomposition of E_k as

$$E_k = \widetilde{E}_k \oplus \widetilde{E}_k^{\perp} \tag{2.4}$$

where the vector space $\widetilde{E}_k^{\perp}$ represents the space of common k-mers within all targets.

The last two relations are useful when we consider the assignment of an object s to a target sequence g_c. Since $T(s) \in E_k$, Equation 2.4 suggests that there must exist two unique vectors $\vec{u}$ and $\vec{u}^{\perp}$ such that $T(s) = \vec{u} + \vec{u}^{\perp}$, where $\vec{u}$ is the orthogonal projection of $T(s)$ to $\widetilde{E}_k$, and $\vec{u}^{\perp}$ is the orthogonal projection of $T(s)$ to $\widetilde{E}_k^{\perp}$. In order words, $\vec{u} = T(s)_{/\widetilde{E}_k}$, and $\vec{u}^{\perp} = T(s)_{/\widetilde{E}_k^{\perp}}$. Let us now focus on $\vec{u}$. Equation 2.3 allows us to decompose this vector by projecting it into each $\widetilde{E}_k^c$:

$$\vec{u} = T(s)_{/\widetilde{E}_k} = \sum_{c=1}^{n} T(s)_{/\widetilde{E}_k^c}$$

It follows that

$$\|\vec{u}\|_1 = \left\| T(s)_{/\widetilde{E}_k} \right\|_1 = \sum_{c=1}^{n} \left\| T(s)_{/\widetilde{E}_k^c} \right\|_1 \tag{2.5}$$

where $\left\| T(s)_{/\widetilde{E}_k^c} \right\|_1$ is the count of g_c-specific k-mers in s. As a consequence, projecting the spectrum of any object s to each target-specific space $\widetilde{E}_k^c$ reveals the uniquely shared substring between object s and target c.

2.2.7 Orthogonal projections

Let us now introduce more properties based on the decomposition described above.

Claim 2. *If an object s is not a substring of a target g_c then* $\left\|T(s)_{/\widetilde{E}_k^c}\right\|_1 = 0$.

Proof. If s is not a substring of $g_c \in \{g_1, g_2, \ldots, g_n\}$ then any k-mer from s cannot be g_c-specific. Therefore, the count g_c-specific k-mers contained in s is 0. The conclusion follows. ■

Claim 3. *If s is a repeat of $\{g_1, g_2, \ldots, g_n\}$ then for all $c \in \{1, 2, \ldots, n\}$, we have* $\left\|T(s)_{/\widetilde{E}_k^c}\right\|_1 = 0$.

Proof. Recall that $T(s) = \vec{u} + \vec{u}^{\perp}$ and $\|\vec{u}\|_1 = \sum_{c\in\{1,2,\ldots,n\}} \left\|T(s)_{/\widetilde{E}_k^c}\right\|_1$. For any c, $\left\|T(s)_{/\widetilde{E}_k^c}\right\|_1$ is the count of k-mers specific to g_c contained in s. Now, let us assume for some c, $\left\|T(s)_{/\widetilde{E}_k^c}\right\|_1 \neq 0$, this implies that s contains at least one k-mer that is specific to g_c and no other target. So s contains a substring that appears only in one target sequence. In other words, s is not repeated in its entirety, so this contradicts the hypothesis that s is a repeat. This implies that for all $c \in \{1, 2, \ldots, n\}$, we have $\left\|T(s)_{/\widetilde{E}_k}\right\|_1 = 0$. ■

Theorem 4. *Given a set of targets $\{g_1, g_2, \ldots, g_n\}$, and a set of objects $\{s_1, s_2, \ldots, s_p\}$, if s_l originates from at least one target in $\{g_1, g_2, \ldots, g_n\}$ then there exists at most one index $c^*(1 \leq c^* \leq n)$ such that for all $c \in \{1, 2, \ldots, n\}, c \neq c^*, \left\|T(s_l)_{/\widetilde{E}_k^c}\right\|_1 = 0$, where for each target c, $1 \leq c \leq n$, $\widetilde{E}_k^c$ is the g_c-specific k-mer space.*

Proof. Let s_l be a sequence in $\{s_1, s_2, \ldots, s_p\}$. If s_l is a repeat of $\{g_1, g_2, \ldots, g_n\}$ then Claim 3 holds. Then, the conclusion follows. Otherwise, if s_l is not a repeat then s_l is a substring of exactly one sequence g_{c^*}. By Claim 2, for all $c \in \{1, 2, \ldots, n\}, c \neq c^*, \left\|T(s_l)_{/\widetilde{E}_k^c}\right\|_1 = 0$. ■

When s_l is a substring of exactly one target sequence g_{c^*}, if s_l does not contain any g_{c^*}-specific k-mers then $\left\|T(s_l)_{/\widetilde{E}_k^{c^*}}\right\|_1 = 0$. This may happen when the sequence s is too short to capture any g_{c^*}-specific k-mers or if k is too small. However, if $\left\|T(s_l)_{/\widetilde{E}_k^{c^*}}\right\|_1 \neq 0$ then the origin of the sequence s is g_{c^*}.

Theorem 4 shows that, given an object s the projections of $T(s)$ on all targets-specific spaces are guaranteed to be null, except for the one that is related to the origin of s. As a consequence, if a sequence s is known to be a substring of at most one target in $\{g_1, g_2, \ldots, g_n\}$, then the problem of assigning s is reduced to the problem of studying non-null projections of $T(s)$ on the n specific vector spaces.

2.2.8 Classification overview

Theorem 1 lays the theoretical foundation of the CLARK's classification method. Given an object s, let us consider what are the possible situations when we compute the projections of the spectrum $T(s)$ on all target-specific spaces. Based on the theory developed so far, we either expect that (a) the number of non-null projection is zero then there is no information to classify the object s (in this case, one may need to increase the value of k and repeat the projections) or (b) there is exactly one non-null projection, say $\left\|T(s)_{/\widetilde{E}_k^{c}}\right\|_1 \neq 0$, for some c, then the object s contains g_c-specific k-mers. In this case, the object s can be classified to target c.

In practice, however, real data are noisy and thus one can observe that the null projections have instead low counts. The two previous cases can be summarized by the following rule: first compute

$$c^* = \arg \max_{1 \leq c \leq n} \left\|T(s)_{/\widetilde{E}_k^{c}}\right\|_1 \tag{2.6}$$

then the object s is assigned to target c^* if $\left\| T(s)_{/\widetilde{E}_k^{c^*}} \right\|_1 > 0$.

In other words, instead of expecting up to two non-null projections, we should expect up to two projections having high 1-norm compared to all others. Given a object s, CLARK computes the highest norm, namely $\left\| T(s)_{/\widetilde{E}_k^{c^*}} \right\|_1$, and the second highest norm, namely $\left\| T(s)_{/\widetilde{E}_k^{c^{**}}} \right\|_1$. Then, CLARK evaluates the confidence of the assignment by using the following confidence score, which ranges from 0.5 to 1.

$$\text{confidence} = \frac{\left\| T(s)_{/\widetilde{E}_k^{c^*}} \right\|_1}{\left\| T(s)_{/\widetilde{E}_k^{c^*}} \right\|_1 + \left\| T(s)_{/\widetilde{E}_k^{c^{**}}} \right\|_1}$$

Another useful statistic is $\gamma = \sum_{1 \leq c \leq n} \left\| T(s)_{/\widetilde{E}_k^{c}} \right\|_1 / \left\| T(s) \right\|_1$, which indicates the proportion of k-mers hitting all targets.

2.2.9 Classification algorithm

Given a set of targets $\{g_1, g_2, \ldots, g_n\}$, a set of objects $\{s_1, s_2, \ldots, s_p\}$ and an integer k, CLARK's computes for each object s (1) the top two target assignments, (2) the confidence score, (3) the number of hits against each target and (4) γ.

To achieve efficient computations, we use a key-value storage (hash table) to store all k-mers from the targets. Observe that each discriminative or target-specific k-mer can be associated to at most one target. This data structure also allows one to remove all common k-mers and to perform fast queries (constant time, on average). We have designed our own hash table of size L based on a chaining structure. The hash function h is defined as follows. Given a k-mer km represented by a number l, where $l = \sum_{i=1}^{k} a[i]4^{i-1}$ (with $a[i] = 0$ if $km[i] = A$, $a[i] = 1$ for C, $a[i] = 2$ for G and $a[i] = 3$ for T or U), we define $h(l) = l \mod L$, where L is defined below. To reduce the

amount of bits to be stored per k-mer, we only save in the hash table the value l/L for bucket $h(l)$. Indeed, since L is known, $h(l)$ and l/L contain enough information to compute back l because $l = (l/L) \times L + h(l)$. If $k = 31$ and $L > 4^{15}$ then (l/L) can be stored in four bytes. As a consequence, any 31-mer can be represented with only four bytes instead of eight. If $k \leq 23$ and $L > 4^{15}$ then two bytes are enough to store any k-mer; if $k \leq 19$ and $L > 4^{15}$ then only one byte is enough. The implementation of our algorithm using a hash table is illustrated in Table 2.1 below.

2.2.10 Full, Default and Express mode

CLARK offers several modes of execution. The first mode (henceforth named Full) outputs for each object the number of hits against all the targets and the confidence score of the assignment.

The second mode (called Default) employs a pseudo-random sampling of the target-specific k-mers to reduce the number the k-mers to load in memory for classification, and it outputs assignments without any detailed statistics so that the output size is significantly reduced. Because it uses less target-specific k-mers than in the Full mode, the default mode is slightly less accurate, but it is faster (see Table 2.2 and 2.3).

The third mode (called Express) loads in memory all the target-specific k-mers, however it outputs results in the same way than that of the Default mode. In addition, this mode is designed to achieve high classification speed by performing a reduced number of k-mer queries to the database (i.e., it queries only non-overlapping k-mers that are found in the object). This enables the Express mode to be significantly faster compared to the Default mode or the Full mode, especially in the case the objects are long sequences (see Table 2.2 and 2.3).

2.2.11 Parallel computing

To process large input files in parallel and in a memory-scalable fashion, CLARK exploits the following multithreading algorithm. In default (or express) mode for single-end reads, CLARK partitions the input file into n bins of reads of equal size (where n is the number of parallel threads requested by the user) and classifies the reads of each of these bins in parallel. Once all threads are completed, the program writes the results in disk. In full mode, CLARK first selects a continuous block of reads (up to two million), and then classifies the reads in a block as described in the default (or express) mode. The full mode consumes more memory as it provides more information per read (i.e., confidence and gamma scores or hit counts per target) and thus the reads extraction is needed to control the memory used. In the case of paired-end reads, the two FASTQ files are first merged (i.e., each read pair is concatenated with a spacer composed of several "N" in between them) before partitioning the reads (in default or express mode) or extracting the reads (in full mode). Our multithreading algorithm assures that the RAM-usage remains constant independently of the size of the sample file. A similar technique was used in the tool BRAT-NOVA for bisulfite-treated reads [Harris et al., 2016].

2.2.12 CLARK-*l*, a RAM-light variant of CLARK

Often the memory needed by CLARK can exceed the RAM available for users with limited computational resources. For users with limited amounts of RAM, we have designed CLARK-l (light). CLARK-l is a variant of CLARK that has a much smaller RAM footprint but can classify objects with similar speed and accuracy.

The reduction in RAM is achieved by constructing a hash-table of smaller size and by

selecting a smaller sets of discriminative k-mers. Instead of considering all k-mers in a target, CLARK-l samples a fraction of them. CLARK-l uses $k = 27$ (27-mers appeared to be a good trade-off between speed, low memory usage and precision) and skips four consecutive/non-overlapping 27-mers. As a result, CLARK-l's peak RAM usage is about 3.8 GB during the index creation, and 2.8 GB when computing the classification (see next section of the chapter). CLARK-l has also the advantage to be very fast in building the hash table.

As described in the next section of this chapter, we show that while the precision and sensitivity are lower compared to CLARK, CLARK-l still achieves high precision and high speed and represents a good solution for users with limited RAM machines.

2.3 Evaluation of the performance

We have evaluated CLARK on synthetic datasets and real metagenomes. The synthetic datasets are composed of DNA sequences whose taxonomy (or "ground truth") is known using synthetic reads generator such as ART [Huang et al., 2012] (cf. [Escalona et al., 2016] for a comparison of the standard synthetic reads simulators) and thus enable us to estimate the performance in accuracy and running time (or speed). While the ground truth for real metagenomes is unknown, it is nonetheless possible to evaluate the speed and whether or not the results are consistent with published results.

2.3.1 Synthetic and real datasets

We used three microbial synthetic metagenomics datasets called "HiSeq", "MiSeq" and "simBA-5" that were introduced in [Wood and Salzberg, 2014]. According to [Wood and Salzberg,

2014], "the HiSeq and MiSeq metagenomes were built using twenty sets of bacterial whole-genome shotgun reads. These reads were found either as part of the GAGE-B project [Magoc et al., 2013] or in the NCBI Sequence Read Archive. Each metagenome contains sequences from ten genomes (see Additional file 1: Table S1 in [Wood and Salzberg, 2014] for the list of genomes). For these metagenomes, 10% of their sequences were selected from each of the ten component genome data sets (i.e., each genome had equal sequence abundance)". The set simBA-5 included "simulated bacterial and archaeal reads, and was created with an error rate five times higher than" the default [Wood and Salzberg, 2014]. We also analyzed the set simHC of synthetic reads [Mavromatis et al., 2007], which simulates high complexity communities lacking dominant populations. SimHC contains 113 sets of reads from various microbial genomes. From simHC, we selected arbitrarily twenty distinct genomes, and extracted the first 500 reads for each genome to build a total of 10,000 reads (see Table 2.10 for the list of genomes). We called this latter dataset simHC.20.500. HiSeq and MiSeq can be considered set of read of low/medium complexity while simBA-5 and simHC.20.500 can be considered set of reads of high complexity. Each of these datasets contains 10,000 reads. The average read length in HiSeq was 92 bp, 156 bp in MiSeq, and 951 bp in simHC.20.500. In simBA-5, all reads are 100 bp long.

We have arbitrarily chosen three real metagenomic samples selected from the Human Microbiome Project [Consortium et al., 2012, Human Microbiome Project Consortium , 2012]. The three samples we used were SRS015072 (mid-vagina) containing 572 thousand paired-end reads, SRS019120 (saliva) containing 4.3 million paired-end reads, and SRS023847 (nose) containing 5.2 million paired-end reads. The microbial abundance and composition of these samples have been determined by using standard and sequence-alignment based methods, such as MetaPhlAn.

2.3.2 Comparison against the best state-of-the-art methods

We have run CLARK on the four synthetic datasets described above and compared its classification results against the state-of-the-art methods, namely NBC [Rosen et al., 2011], which we chose for its high accuracy (currently the most sensitive metagenomics classifier, according to [Bazinet and Cummings, 2012]), and Kraken, which we chose due to its high speed [3] and its high precision at the genus level.

2.3.3 Evaluation of the speed and accuracy

Database in metagenomics

We have tested CLARK using the set of bacterial and archaeal genomes from NCBI/RefSeq. At the time we carried out the experiments the NCBI/RefSeq database was composed of 2,752 complete bacterial genomes, distributed into 695 distinct genera, or 1,473 species. The total length of all these bacterial genomes was about 9.5 Gbp. The average size of a genome was about 3.5 Mbp.

HiSeq, MiSeq, simBA-5 and simHC.20.500

For a given level in the taxonomy tree (e.g., genus), we define *precision* as the fraction of correct assignments over the total number of assignments, and *sensitivity* as the ratio between the number of correct assignments and the number of objects to be classified. In order to have a fair comparison against KRAKEN's assignments, when KRAKEN produces an assignment that is not available at or below the genus or species level, it is then considered as not assigned.

[3] According to [Wood and Salzberg, 2014], Kraken'speed was unmatched among the standard and best state-of-the-art read-level classifiers available the time such as NBC, PHYMMBL[Brady and Salzberg, 2011] and MEGABLAST [Zhang et al., 2000], when it was published in 2014.

Table 2.2 reports precision, sensitivity and processing speeds (in 10^3 reads per minute) obtained by NBC, KRAKEN and CLARK on the HiSeq, MiSeq, simBA-5 and simHC.20.500 datasets, for several values of the k-mer length. The table illustrates how the performance of these tools is affected by the choice of k. By increasing k one generally increases precision, but can lower sensitivity (also see Figure 2.2). To carry out a fair comparison between tools, we decided to first determine NBC's and KRAKEN's optimal k-mer length, and then run CLARK with a value of k that would match either sensitivity or precision.

NBC was tested with $k = 11, 13, 15$. We observed that $k = 15$ produced the highest sensitivity on all datasets. The value $k = 15$ is the highest possible value, which is recommended by the authors of [Rosen et al., 2011] for datasets composed of short reads. Since NBC produces detailed statistics on the assignments, we executed CLARK in Full mode for a fair comparison. Using $k = 20$ for CLARK (Full mode) we obtained a similar sensitivity to NBC (CLARK is actually more sensitive than NBC on HiSeq and simHC.20.500). At the same level of sensitivity of NBC, CLARK achieves a higher precision and it is thousands of times faster.

In the case of KRAKEN, $k = 31$ was the value used in [Wood and Salzberg, 2014] for HiSeq, MiSeq and simBA-5 and it is supposed to achieve the highest precision. We also tried to run KRAKEN for other values of k. As expected, Table 2.2 shows that $k = 31$ produces the best precision for all the datasets. For this comparison, we also ran CLARK with $k = 31$. Observe that CLARK (Default mode) is slightly less sensitive than KRAKEN but is more precise and faster. The difference in speed is significant for all datasets of short reads ($300 - 800$ thousand additional reads/min). On simHC.20.500, KRAKEN and CLARK achieve the same speed due to the fact that these datasets contain longer reads. Finally, CLARK has better sensitivity than KRAKEN on simHC.20.500.

The same comparisons were carried out between the two variants of KRAKEN and CLARK optimized for speed, called KRAKEN-Q and CLARK-E (E for "Express"). As indicated in Table 2.2, KRAKEN-Q achieves the best precision for all the datasets when $k = 31$, which is consistent with [Wood and Salzberg, 2014]. However, when $k = 31$ CLARK-E runs four–five times faster than KRAKEN-Q and is also more precise. In addition, observe that as we decrease k, both variants gets faster but CLARK-E maintains a precision above 90% while KRAKEN-Q produces progressively lower precisions. In the last row of Table 2.2, we report the performance of CLARK-l, another variant of CLARK designed for low RAM architectures that runs only for $k = 27$ (see Methods section). CLARK-l performs assignments with a lower precision than CLARK (the difference is at most 3.5% in these experiments) but can process more than 1.5 million of reads per minute on HiSeq or simBA-5, and only uses about 4% of the memory used by CLARK (cf. Table 2.5).

All experimental results reported so far were obtained in single-threaded mode. If a multicore architecture is available, CLARK and KRAKEN can take advantage of it. In Table 2.6, we summarize the classification speed of the two tools using 1, 2, 4 or 8 threads for $k = 31$. Observe that using eight threads, CLARK achieves a speed-up of 5.2x compared to one thread, while KRAKEN only achieves a speed-up of 1.2x. When comparing CLARK-E to KRAKEN-Q, we can make similar observations. In general, note that CLARK-E is at least five times faster than KRAKEN-Q, independently of the number of threads used.

For the analysis at the species level, we repeated the classification of the objects in the four datasets described above against species-level targets. This time we used values of k that allowed best sensitivity for NBC ($k = 15$) and best precision for KRAKEN ($k = 31$). Observe in Table 2.3 that NBC achieves the best sensitivity on all datasets. However, when CLARK is ran in Full mode

using $k = 20$, it achieves a higher precision than NBC on HiSeq, MiSeq and simHC.20.500, and is several orders of magnitude faster. In addition, CLARK in Default mode using $k = 31$ achieves higher precision than KRAKEN on all datasets (as much as 10% higher on HiSeq and MiSeq) when $k = 31$. CLARK also outperforms the speed of KRAKEN on HiSeq, MiSeq and simBA-5. On simHC.20.500, since the reads are much longer, the speed of KRAKEN and CLARK are comparable. But, CLARK has higher sensitivity than KRAKEN on HiSeq, MiSeq and simHC.20.500. Finally, the fast variant CLARK-E, as previously observed for the experiments at the genus level, outperforms KRAKEN-Q in both speed and precision.

Human microbiome samples

In the second experiment, we used CLARK to classify Human Microbiome Project reads against 695 genus-level targets described above. This time, however, the "ground truth" was not available.

Using $k = 31$, CLARK was able to assign 42.1% of the reads in SRS015072 (mid-vagina), 30.8% of the reads in SRS019120 (saliva) and 49.8% of the reads in SRS023847 (nose). KRAKEN achieved similar rates of assigned reads using $k = 31$. Reducing k would increase the number of assignments, at the cost of increasing the probability of misclassification. We investigated whether we could take advantage of CLARK's confidence scores to compensate for a smaller value of k, and improve the fraction of assigned reads.

Figure 2.2-a to Figure 2.2-d show that CLARK's sensitivity on the four datasets is the highest for $k = 20$ or $k = 21$. However, the precision for $k = 20$ and $k = 21$ is about 15% lower than for $k = 31$, which implies that a large proportion of assignments may be incorrect. We have strong experimental evidence that shows that the higher is CLARK's confidence score for

an assignment, the more likely that assignment is correct (see next paragraph). In addition, we observe in Figure 2.2-a to Figure 2.2-d that the precision of high confidence assignments is higher than the average precision of all assignments, and is relatively constant for all k-mer length. The idea is to use $k = 20$ to maximize the number of assigned reads, but only consider high confidence assignments to increase the precision. We call an assignment *high confidence* if the confidence score is higher than 0.75, *low confidence* otherwise.

Observe in Table 2.4 that the number of high confidence assignments for $k = 20$ is significantly higher than for $k = 31$. The relative increase in assignments is about 40% (from 42.1% to 62.3% in SRS015072, 30.8% to 55.1% on SRS019120, and 49.8% to 68.3% on SRS023847). Table 2.4 also reports the most frequent five genera in high confidence assignments. For the saliva sample, the dominance of *Streptococcus*, *Haemophilus* and *Prevotella* is consistent with findings in [Human Microbiome Project Consortium , 2012] and [Wood and Salzberg, 2014]. Study [Said et al., 2013], which focused on salivary microbiota of 35 inflammatory bowel disease patients, also reports *Streptococcus*, *Prevotella*, *Neisseria*, *Haemophilus* and *Veillonella* as dominant genera. Concerning the mid-vagina sample, we have found that *Lactobacillus* is the dominant genus, in agreement with findings reported in [Antonio et al., 1999, Hyman et al., 2005, Human Microbiome Project Consortium , 2012]. The proportion of *Lactobacillus* we have identified (64.7%) is very close to the reported proportion (69%–71%) in [Antonio et al., 1999, Hyman et al., 2005]. The presence of *Pseudomonas* and *Gardnerella* is expected because some individuals who lack *Lactobacillus* have instead *Gardnerella* or *Pseudomonas* as the predominant bacteria [Antonio et al., 1999, Hyman et al., 2005]. In the nose sample, the high presence of *Propionibacterium* and *Staphylococcus* is consistent with the results in [Human Microbiome Project Consortium , 2012].

2.3.4 Applications in genomics: Barley BACs and unigenes

In this section, we show that CLARK can be also applied to another problem in genomics, namely BACs/unigenes assignment to chromosomes or chromosomes arms/centromeres. We carried out this classification on the barley genome.

Inputs to this classification task were (1) barley chromosome arms (targets) and (2) barley BACs or unigenes (objects). Samples of each barley chromosome arm were obtained using flow-sorting [Doležel et al., 2012]. The procedure to obtain gene-rich barley BACs was described in [Lonardi et al., 2013]. Sequences for chromosome arms and BACs were generated on an Illumina HiSeq 2000 at UC Riverside.

For the targets, we processed thirteen datasets of shotgun sequenced reads: one for barley chromosome 1H and twelve for barley chromosome arms (namely, 2HL, 2HS, 3HL, 3HS, 4HL, 4HS, 5HL, 5HS, 6HL, 6HS, 7HL, and 7HS). After quality-trimming the reads, we had a total of about 181 Gbp of sequence data. The cumulative size of the assembled barley chromosome arms obtained via SOAPDENOVO [Luo et al., 2012] resulted in about 2 Gbp (about 40% of the barley genome).

The objects were 50,938 barley unigenes (transcript assembly from ESTs) obtained from [Close et al., 2007] for a total of about 222.4 Mbp. Additionally, we trimmed short reads for 15,721 BACs obtained from [Lonardi et al., 2013], for a total of about 1.73 Gbp. We also had access to 15,697 BAC assemblies (not all BACs had a sufficient number of reads for an assembly) for a total of about 1.80 Gbp. While the genomic location for the majority of these "objects" was unknown, we had 1,652 unigenes for which a location was derived from the Golden Gate oligonucleotide pool assay (OPA) [Close et al., 2009], which allowed us to determine a presumed location of 2,252

BACs [Lonardi et al., 2013]. We should point out that although we have used these locations as the "ground truth" to establish the accuracy of the classification, our observations indicate about 5% errors in these OPA assignments [Lonardi et al., 2013].

As stated above, the most critical parameter in CLARK is the length of the k-mer used for classification. By assuming that the subset of the unigenes that have a location via OPA were correct, we were able to estimate CLARK's precision and sensitivity for various choices of k. Figure 2.2-e shows these statistics, along with the assignment rate (fraction of unigenes assigned) and the average confidence score for all assignments. Observe that as k increases, the number of assignments decreases but the precision/sensitivity increases. Based on this analysis we determined that $k = 19$ represents a good tradeoff for this dataset.

Table 2.8 summarizes CLARK's assignment of barley unigenes (assemblies) to barley chromosomes arms (assemblies) using $k = 19$. When both targets and objects are assemblies, we call this an "A2A" assignment. Observe that most of the assignments have high confidence and they are relatively evenly distributed among barley chromosome arms (the seven barley chromosomes are believed to be relatively similar in length). Observe in Figure 2.2-e that CLARK's precision and sensitivity for this classification task is very high (both at 98.49%) while the average confidence score is above 0.96, and 99.44% of unigenes are assigned.

Table 2.7 presents a summary of CLARK's assignment of barley BACs (assemblies) to arms (assemblies), while Table 2.9 refers to the same assignments based on the reads instead of the assemblies ("R2R" assignment). The consistency between these results (same distribution of BACs assignments over chromosome arms, and similar proportion of high and low confidence assignments) demonstrates the robustness of our approach. The agreement with OPA-based locations

is 92.9% for R2R assignments, and 93.2% for A2A assignments. Observe that the agreement for BAC/arm assignments is lower than unigene/arm assignments (98.49%).

Running time analysis

All experiments presented in this study were run on a Dell PowerEdge T710 server (dual Intel Xeon X5660 2.8 Ghz, 12 cores, 192 GB of RAM). CLARK-l was also run on a Mac OS X, Version 10.9.5 (2.53 GHz Intel Core 2 Duo, 4 GB of RAM). When comparing KRAKEN to CLARK in their Default mode, and KRAKEN-Q to CLARK-E, we always set KRAKEN to "preload" its database in main memory and print results to a file (instead of the standard output) to achieve the highest speed. For consistency, CLARK was also run under the same conditions. For the results in Table 2.2, CLARK, NBC (v1.1), and KRAKEN (v0.10.4-beta) were run in single-threaded mode, three times on the same inputs in order to smooth fluctuations due to I/O and cache issues (the reported numbers are best values). We have also run the latest versions of Kraken (v0.10.5-beta abd v0.10.6-beta), and we did not observe a significant variation of accuracy and usage of RAM. However, we observed a 15% decrease in the classification speed compared to version v0.10.4-beta.

2.3.5 Impact of the choice of k on the accuracy

To determine the optimal value of k for a particular dataset one can take advantage of prior knowledge, as we did in the case of unigene/BAC assignment to chromosomes. In that case, we had 1,657 unigenes for which the correct assignment (approximately 95% accuracy) was experimentally determined via the use of two Illumina GoldenGate assays (BOPA1 and BOPA2) to assign genes to BACs. Given these known assignments, we estimated precision and sensitivity, as well as the average confidence score for all assignments and the assignment rate (see Figure 2.2-e). Observe

that k=19 maximizes all of the four measurements. Higher precision and average confidence score can be achieved by using higher k but at the cost of decreasing sensitivity and assignment rate.

Similar evaluation were carried out on the metagenomic datasets. Figure 2.2-a to Figure 2.2-d show precision, sensitivity, as well as assignment rate and average confidence score as a function of k. In both cases we observe that as we increase k, precision and the average confidence score are increasing, while the sensitivity is decreasing. We observe that the maximum sensitivity is achieved for k in the range 19–22 for all metagenomic datasets, independently of the reads length or complexity.

As a consequence, users interested in high sensitivity (or high number of assignments) must choose k between 19 and 22, and user interested in high precision (or high confidence score) must choose k higher than 26. The largest value of k supported in the CLARK implementation[4] is 32.

2.3.6 Confidence score analysis

CLARK, unlike most other sequence classifiers, provides confidence scores. Here we want to study the relation between confidence scores and correctness of results.

Figure 2.3 shows the distribution of the number of assignments as a function of the confidence score for all the datasets presented in this study, namely barley BACs and unigenes (A2A), barley BACs (R2R and A2A), and the four metagenomic datasets ("HiSeq", "MiSeq", "simBA-5", and "simHC.20.500"). Observe the high density of high confidence assignments in all cases, especially for "HiSeq" and "MiSeq" datasets. For all these datasets, when running CLARK in Full mode, we observe that at least 95% of all assignments have confidence score higher or equal than

[4]The current version of CLARK is v1.2.3.

0.98. This is clear evidence that, in the Full mode, conflicts (due to sequencing errors and/or other noises) in the classification rarely occur.

Figure 2.4 shows the proportion of correct assignments (y-axis) as a function of confidence score ranges (x-axis). Observe that at least 95% of assignments having confidence of 0.90 or higher are correct.

2.4 Conclusion

We have presented CLARK, a new method for sequence classification that is ultra-fast, accurate and versatile. Experimental results demonstrate that CLARK has several advantages over previous methods. (i) CLARK is able to classify short metagenomic reads with high accuracy at multiple taxonomic ranks (i.e., species and genus level) and its assignments on real metagenomic samples are consistent with findings published in the literature. (ii) It achieves the same or better accuracy than the best state-of-the-art metagenomic classifiers. (iii) The classification speed of CLARK, in the context of metagenomics, is unmatched, with 32 million metagenomic short reads per minute using one CPU (five times faster than KRAKEN). In addition, CLARK scales better on a multi-core architectures: the speed-up one can obtain by adding more threads is higher than Kraken. (iv) CLARK is able to output statistics for each assignment, is user-friendly and self-contained (unlike most of other classifiers, it does not require external tool such as BLAST or MEGABLAST, etc). (v) it can be executed with relatively small amounts of RAM (unlike LMAT) or disk space (unlike PHYMMBL or KRAKEN). Indeed, LMAT can use about 500 GB of RAM, while the maximum amount of RAM needed by CLARK is less than 165 GB (see Table 2.5). PHYMMBL or KRAKEN require respectively about 120 GB and 140 GB of disk space to run, while CLARK

requires less than 40 GB for classification. (vi) In the context of genomics, CLARK can classify with high speed BACs, which are much longer sequences than reads from metagenomics ($\sim$150 kbp [Schulte et al., 2011, Lonardi et al., 2013]), and transcripts with better accuracy than previously used BLAST-based method [Lonardi et al., 2013], and it can infer centromeric regions (see [Ounit et al., 2015, Muñoz-Amatriaín et al., 2015], for more details). Although in this chapter we focused the attention on genus and species level classification, CLARK can also accurately classify at higher taxonomic levels such as phylum (see next chapter) and still achieve high accuracy and speed at the same time. After its publication in March 2015, the CLARK tool has been intensively evaluated, by several independent groups (for example [Lindgreen et al., 2016, Galata et al., 2016]) and is now a standard tool in various international research groups using metagenomics.

The main strength of CLARK resides in its simplicity. The simplicity in the design of CLARK's algorithm allows it not only to run with unprecedented speed and accuracy, whether it is in the context of metagenomics or genomics. We believe CLARK can be useful for a variety of molecular biology applications. For instance, it can be used to analyze large number of sequenced BACs, for other large and repetitive genomes like bread wheat (*Triticum aestivum*), which is currently being sequenced by International Wheat Genome Sequencing Consortium (`https://www.wheatgenome.org/`). In genome assembly project, CLARK can be used to detect contaminants in raw reads or to identify chimeric reads. In the context of antimicrobial resistance studies, it is crucial to understand the resistance of bacteria at the genomic scale and interactions between organisms [McArthur and Wright, 2015]. The detection of microbial resistance to antibiotics in environmental samples can be carried out by CLARK based on, for example, the comprehensive antibiotic resistance database (CARD) [McArthur et al., 2013, Jia et al., 2017].

Table 2.1: Description of CLARK's algorithm ("Full" mode).

	Input: integer k, n target sequences $(g_c)_{1\leq c\leq n}$, p object sequences $(s_l)_{1\leq l\leq p}$
1	if hash table H related to $(T(g_c))_{1\leq c\leq n}$ already exists then
2	load H
3	goto 15
4	create an empty hash table H
5	for all $c, 1 \leq c \leq n$:
6	for each $(km, w) \in T(g_c)$: /* where km is a k-mer and w is the occurrence of km in g_c */
7	if ($km \in H$) then
8	update the list of targets associated to km by adding c and increase the occurrence of km by w
	else
9	insert (km, w, c) in H
10	for each $km \in H$:
11	if the list of targets for km has more than three elements then
12	remove km from H
	else
13	if the list of origins for km has exactly two elements $(c_1, c_2, c_1 < c_2)$ and from different chromosomes then
14	remove km from H
	Store H in disk for future run
15	for all $l, 1 \leq l \leq p$:
16	if $T(s_l) = 0$ then
17	output l, "not assigned" continue
18	create n empty bins: $b_1, b_2, \ldots, b_n, \ldots, b_n$
19	for each $(km, w) \in T(s_l)$:
20	if $km \in H$ (in target c) then
21	$b_c = b_c + w$
22	c^* = $\arg\max\{b_1, b_2, \ldots, b_n, \ldots, b_n\}$
23	c^{**} = $\arg\max\{\{b_1, b_2, \ldots, b_n, \ldots, b_n\} - \{b_{c^*}\}\}$
24	$\gamma = \sum_{1\leq t\leq n} b_t / T(s_l)$
25	If $\gamma = 0$ then
26	output l, "not assigned" continue
27	$confidence = \frac{b_{c^*}}{b_{c^*} + b_{c^{**}}}$
28	output l, b_1, b_2, …, b_n, γ, c^*, c^{**}, $confidence$

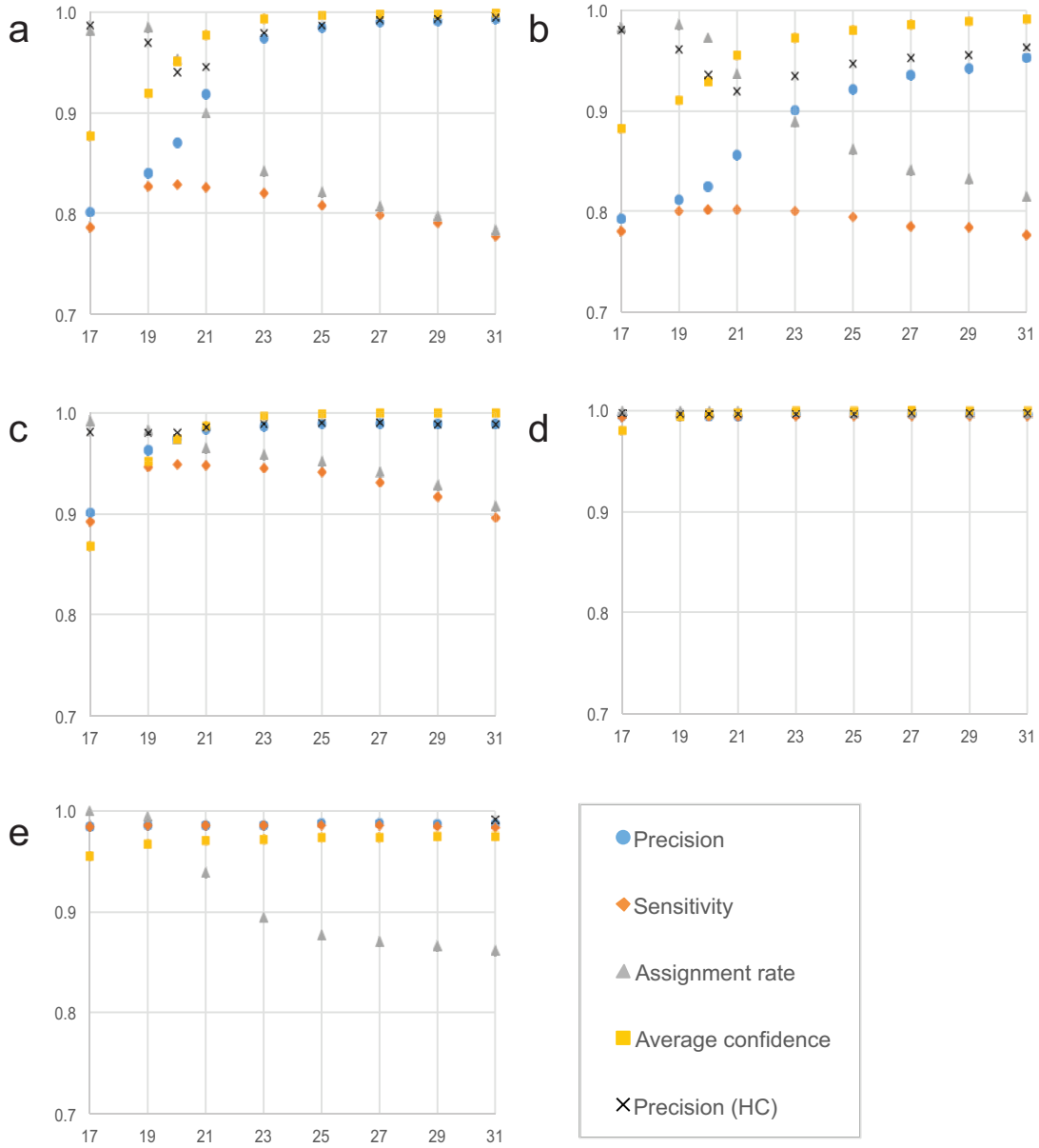

Figure 2.2: Classification performance of CLARK for several k-mer length and for various datasets. CLARK's precision, sensitivity, assignment rate, average confidence scores and precision of high confidence assignments (HC) for several choices of the k-mer length on the "HiSeq" metagenomic dataset (a), the "MiSeq" metagenomic dataset (b), the "simBA-5" metagenomic dataset (c), the "simHC.20.500" metagenomic dataset (d), and barley unigenes (e). (a) - (d) are results of the classification against the 695 genus-level targets.

Table 2.2: Performance statistics for several choices of the k-mer length for NBC, KRAKEN, CLARK and their fast variants on the classification of "HiSeq", "MiSeq", "simBA-5" and "simHC.20.500" metagenomic datasets against the 695 genus-level targets; precision and sensitivity are expressed as percentages, while speed is expressed in 10^3 reads per minute; KRAKEN-Q and CLARK-E are faster, but less accurate, variants of these tools; CLARK-l is a less memory-intensive version of CLARK which runs only for k=27; experiments were carried out in single-threaded mode; *parameter k is referred as N in the NBC manuscript.

	k	HiSeq			MiSeq			simBA-5			simHC.20.500		
		Prec	*Sens*	*Speed*	*Prec*	*Sens*	*Speed*	*Prec*	*Sens*	*Speed*	*Prec*	*Sens*	*Speed*
NBC	15*	**82.57**	**82.57**	0.008	**81.00**	**81.00**	0.007	**97.69**	**97.69**	0.007	**99.40**	**99.40**	0.005
	13*	78.85	78.85	0.011	77.70	77.70	0.009	92.41	92.41	0.010	98.57	98.57	0.006
	11*	58.97	58.97	**0.020**	64.43	64.43	**0.016**	46.10	46.10	**0.017**	86.83	86.83	**0.008**
CLARK(Full)	31	**99.26**	77.78	**541**	**95.33**	77.69	**435**	98.88	89.67	**591**	**99.68**	**99.42**	121
	27	98.98	79.88	538	93.50	78.57	433	**98.90**	93.09	585	99.67	**99.42**	**122**
	23	97.33	81.97	530	90.06	80.02	426	98.71	94.54	559	99.59	**99.42**	119
	20	87.00	**82.87**	532	82.45	**80.19**	420	97.38	**94.80**	549	99.43	99.41	115
KRAKEN	31	**99.26**	77.76	**2,332**	**95.50**	77.59	**1,361**	98.28	89.35	**1,976**	**96.83**	96.55	**237**
	27	99.01	79.85	2,048	93.91	78.47	1,240	**98.31**	92.73	1,917	96.85	96.57	231
	23	97.45	81.89	1,923	90.56	79.75	1,186	98.25	94.18	1,824	96.80	96.57	228
	20	90.22	**82.67**	1,546	86.28	**79.99**	965	98.07	**94.44**	1,478	96.71	**96.59**	211
CLARK	31	**99.31**	77.25	**3,116**	**95.66**	77.44	**1,670**	**98.91**	88.62	**2,855**	**99.68**	**99.42**	**251**
	27	99.07	79.37	2,796	93.90	78.29	1,522	98.90	92.26	2,554	99.67	**99.42**	241
	23	97.85	81.36	2,679	90.98	79.57	1,482	98.75	94.26	2,394	99.60	**99.42**	244
	20	88.60	**82.26**	2,567	83.35	**79.77**	1,456	97.73	**94.49**	2,306	99.43	99.41	239
KRAKEN-Q	31	**99.20**	76.84	6,224	**95.81**	**74.13**	5,308	**98.17**	87.46	7,023	**91.17**	**85.79**	3,809
	27	98.79	78.19	6,410	94.12	73.73	5,555	98.11	**89.89**	7,992	90.99	83.71	4,196
	23	96.67	**78.48**	7,015	90.57	72.35	6,329	97.21	89.07	8,989	90.46	79.27	4,574
	20	82.07	70.11	**9,437**	80.05	65.25	**9,537**	90.02	77.04	**10,961**	70.86	57.40	**5,819**
CLARK-E	31	**99.55**	72.72	**32,450**	**98.11**	**74.58**	**28,988**	**99.00**	77.85	26,171	97.63	97.31	15,426
	27	99.43	74.67	29,897	96.93	75.68	28,459	98.93	84.86	**27,451**	97.47	97.18	**16,124**
	23	98.93	78.20	31,112	95.01	76.88	26,747	98.34	90.20	26,647	**98.56**	**98.32**	15,408
	20	94.74	**78.46**	30,029	90.57	76.60	25,789	96.61	**89.98**	26,545	93.94	93.82	15,587
CLARK-l	27	98.45	62.30	1,525	92.11	69.64	861	95.96	52.00	1,705	99.49	98.94	143

Table 2.3: Summary of performance statistics (precision, sensitivity are expressed as percentages, while speed is expressed in 10^3 reads per minute) for NBC, KRAKEN, and CLARK on the classification of "HiSeq", "MiSeq", "simBA-5" and "simHC.20.500" metagenome datasets against the 1473 species-level targets, in single-threaded mode.

	HiSeq			MiSeq			simBA-5			simHC.20.500		
	Prec	*Sens*	*Speed*	*Prec*	*Sens*	*Speed*	*Prec*	*Sens*	*Speed*	*Prec*	*Sens*	*Speed*
NBC (k=15)	68.67	68.70	0.008	68.33	68.33	0.007	91.74	91.74	0.007	94.32	94.32	0.005
CLARK (k=20)	69.44	61.46	272	70.72	62.45	239	91.32	82.48	269	94.34	94.32	96
KRAKEN (k=31)	74.00	53.49	2,332	77.72	58.72	1,361	92.99	78.70	1,976	84.67	84.31	237
CLARK (k=31)	86.74	58.59	3,011	89.49	61.84	1,566	98.85	76.80	2,855	94.67	94.26	251
KRAKEN-Q (k=31)	75.88	50.78	6,224	78.07	53.68	5,308	92.67	74.39	7,023	82.40	74.84	3,809
CLARK-E (k=31)	90.08	55.18	30,976	94.31	58.36	24,029	98.92	66.02	24,996	92.78	92.38	15,583
CLARK-l (k=27)	85.35	53.95	1,676	85.89	64.91	904	85.55	46.28	1,702	94.06	93.53	141

Table 2.4: Summary of CLARK's assignment ($k = 20$) for three Human Microbiome Project datasets against the 695 genus-level targets. Columns: (1) short read sample ID; (2) percentage of high confidence assignments; (3) percentage of low confidence assignments; (4) percentage of unassigned reads; (5) average confidence score for all assignments; (6) five most frequent genera in high confidence assignments (listed in decreasing order). An assignment is *high confidence* if the confidence score is higher than 0.75, *low confidence* otherwise.

SRS ID	*high confidence assignments (%)*	*low confidence assignments (%)*	*no assignment (%)*	*average confidence score*	*Most frequent genera (high confidence assignments)*
015072 (vagina)	62.3%	25.9%	11.8%	0.868	*Lactobacillus* (64.7%) *Pseudomonas* (7.3%) *Desulfosporosinus* (4.4%) *Clostridium* (1.7%) *Gardnerella* (1.2%)
019120 (mouth)	55.1%	28.2%	16.7%	0.842	*Streptococcus* (27.2%) *Haemophilus* (15.0%) *Prevotella* (11.4%) *Neisseria* (5.0%) *Veillonella* (2.9%)
023847 (nose)	68.3%	23.8%	7.9%	0.954	*Propionibacterium* (61.5%) *Staphylococcus* (8.5%) *Achromobacter* (7.5%) *Alteromonas* (6.3%) *Desulfosporosinus* (5.0%)

Table 2.5: Details of the time and memory usage (RAM and disk) for the installation (or database construction of the 2,752 bacterial genomes of NCBI/RefSeq), and the classification of NBC, KRAKEN and CLARK, in Default mode. Measurements of the installation time and RAM peak usage are done for NBC, KRAKEN and CLARK using default settings and single-thread. RAM peak usage was obtained by the attribute "`maximum resident set size`" of the command "`/usr/bin/time -v`" available on Linux.

	Installation/Database construction			Classification
	Time (HH:MM)	RAM Peak usage (GB)	Memory Disk (GB)	RAM Peak usage (GB)
NBC	19:10	< 1	52.0	< 1
KRAKEN	06:07	167.9	141.0	77.7
CLARK	02:45	164.1	42.4	70.1
CLARK-l	00:05	3.8	< 1	2.8

Table 2.6: Classification speed (expressed as 10^3 reads/min) as a function of the number of threads ($k = 31$).

	"HiSeq" dataset			
Number of threads	1	2	4	8
KRAKEN	2,332	3,647	3,534	3,876
CLARK	3,116	5,484	9,626	15,807
KRAKEN-Q	6,224	7,712	7,693	7,506
CLARK-E	32,450	39,841	46,386	52,896
	"MiSeq" dataset			
Number of threads	1	2	4	8
KRAKEN	1,361	2,038	3,605	3,616
CLARK	1,670	3,040	4,905	8,120
KRAKEN-Q	5,308	5,553	8,362	8,642
CLARK-E	28,988	32,199	41,970	49,383

Table 2.7: Summary of CLARK's assignment of 15,695 BACs (represented as assemblies) to barley chromosome arms (assemblies) and centromeres ($k = 19$). Columns: (1) barley chromosome 1H, twelve chromosome arms, and six centromeres; (2) number of distinct k-mers in each target; (3) number of discriminative k-mers present in target sequences (must occur at least once); (4) number of assigned objects per target; (5) number of low confidence assignment per target; (6) number of high confidence assignment per target; (7) percentage of low confidence assignment (as a fraction of the total number of assigned objects per target); (8) percentage of high confidence assignment (as a fraction of the total number of assigned objects per target).

Targets	*19-mers*	*discriminative 19-mers*	*assignments*	*low confidence*	*high confidence*
1H	180,176,713	108,894,740	2,111	7.1%	92.9%
2HC	-	814,357	0	-	-
2HL	103,679,920	64,700,161	1,424	3.4%	96.6%
2HS	90,912,314	54,449,430	1,071	3.5%	96.5%
3HC	-	1,532,968	0	-	-
3HL	123,140,951	78,158,244	1,411	3.3%	96.7%
3HS	111,951,787	70,473,478	897	5.5%	94.5%
4HC	-	3,105,047	56	67.9%	32.1%
4HL	106,999,773	64,749,958	1,132	3.5%	96.5%
4HS	89,027,872	51,612,790	890	4.4%	95.6%
5HC	-	604,030	0	-	-
5HL	117,915,094	77,128,375	1,658	2.8%	97.2%
5HS	58,067,400	34,037,607	654	5.4%	94.6%
6HC	-	469,530	0	-	-
6HL	74,485,223	44,221,184	1,132	3.4%	96.6%
6HS	111,834,123	83,957,421	846	6.5%	93.5%
7HC	-	795,923	0	-	-
7HL	92,603,503	58,159,248	1,179	3.6%	96.4%
7HS	90,217,777	55,276,671	1,234	4.8%	95.2%
Total	1,351,012,450	853,141,162	15,695	4.6%	95.4%

Table 2.8: Summary of CLARK's assignment of 50,646 unigenes (EST assemblies) to barley chromosome arms (assemblies) and centromeres ($k = 19$). Columns: (1) barley chromosome 1H, twelve chromosome arms, and six centromeres; (2) number of distinct k-mers in each target; (3) number of discriminative k-mers present in target sequences (must occur at least once); (4) number of assigned objects per target; (5) number of low confidence assignment per target; (6) number of high confidence assignment per target; (7) percentage of low confidence assignment (as a fraction of the total number of assigned objects per target); (8) percentage of high confidence assignment (as a fraction of the total number of assigned objects per target).

Targets	*19-mers*	*discriminative 19-mers*	*assignments*	*low confidence*	*high confidence*
1H	180,176,713	108,894,740	8,197	21.1%	78.9%
2HC	-	814,357	15	93.3%	6.7%
2HL	103,679,920	64,700,161	4,776	15.8%	84.2%
2HS	90,912,314	54,449,430	3,334	17.3%	82.7%
3HC	-	1,532,968	29	79.3%	20.7%
3HL	123,140,951	78,158,244	4,726	16.7%	83.3%
3HS	111,951,787	70,473,478	3,159	20.4%	79.6%
4HC	-	3,105,047	54	50.0%	50.0%
4HL	106,999,773	64,749,958	3,531	14.4%	85.6%
4HS	89,027,872	51,612,790	2,468	16.4%	83.6%
5HC	-	604,030	9	88.9%	11.1%
5HL	117,915,094	77,128,375	6,111	12.2%	87.8%
5HS	58,067,400	34,037,607	1,619	17.8%	82.2%
6HC	-	469,530	9	100.0%	0.0%
6HL	74,485,223	44,221,184	2,973	12.4%	87.6%
6HS	111,834,123	83,957,421	2,721	24.4%	75.6%
7HC	-	795,923	9	88.9%	11.1%
7HL	92,603,503	58,159,248	3,556	10.9%	89.1%
7HS	90,217,777	55,276,671	3,350	12.6%	87.4%
Total	1,351,012,450	853,141,162	50,646	16.5%	83.5%

Table 2.9: Summary of CLARK's assignment of 15,665 BACs (represented as reads) to barley chromosome arms (reads) and centromeres ($k = 19$). Description of columns can be found in Table 2.8.

Targets	*19-mers*	*discriminative 19-mers*	*assignments*	*low confidence*	*high confidence*
1H	448,768,897	126,997,864	2,068	4.2%	95.8%
2HC	-	1,738,722	0	-	-
2HL	451,729,142	102,959,160	1,417	2.1%	97.9%
2HS	401,605,473	79,225,936	1,071	2.4%	97.6%
3HC	-	4,631,639	0	-	-
3HL	553,420,081	138,939,217	1,423	2.2%	97.8%
3HS	538,777,930	113,354,224	892	3.5%	96.5%
4HC	-	6,428,726	70	14.3	85.7%
4HL	494,923,209	106,930,230	1,127	2.3%	97.7%
4HS	462,144,322	85,650,765	888	3.4%	96.6%
5HC	-	1,643,194	0	-	-
5HL	558,710,983	121,491,586	1,657	2.3%	97.7%
5HS	281,062,766	57,181,745	658	2.4%	97.6%
6HC	-	1,287,133	0	-	-
6HL	311,443,157	70,856,097	1,136	2.0%	98.0%
6HS	877,169,677	255,819,549	850	2.9%	97.1%
7HC	-	1,697,991	0	-	-
7HL	366,612,780	82,987,499	1,175	2.0%	98.0%
7HS	365,475,556	83,848,867	1,233	2.8%	97.2%
Total	6,111,843,973	1,443,670,144	15,665	2.7%	97.3%

Table 2.10: List of genomes used for the "simHC.20.500" dataset (sequences downloaded from the JGI database).

IMG Taxon ID	Genome
640753002	*Alkaliphilus metalliredigens* QYMF
640427103	*Bradyrhizobium sp.* BTAi1
637000047	*Burkholderia cepacia* AMMD
637000160	*Chelativorans sp.* BNC1
640069309	*Clostridium thermocellum* ATCC 27405
637000088	*Dechloromonas aromatica* RCB
643348537	*Desulfitobacterium hafniense* DCB-2
637000116	*Frankia sp.* CcI3
637000119	*Geobacter metallireducens* GS-15
639633037	*Marinobacter aquaeolei* VT8
637000162	*Methanosarcina barkeri Fusaro*, DSM 804
637000192	*Nitrobacter hamburgensis* X14
639633046	*Nocardioides sp.* JS614
637000208	*Polaromonas sp.* JS666
637000216	*Pseudoalteromonas atlantica* T6c
637000221	*Pseudomonas fluorescens* Pf0-1
640069327	*Rhodobacter sphaeroides* 2.4.1, ATCC BAA-808
637000237	*Rhodopseudomonas palustris* BisB18
637000260	*Shewanella sp.* MR 7
639633063	*Syntrophobacter fumaroxidans* MPOB

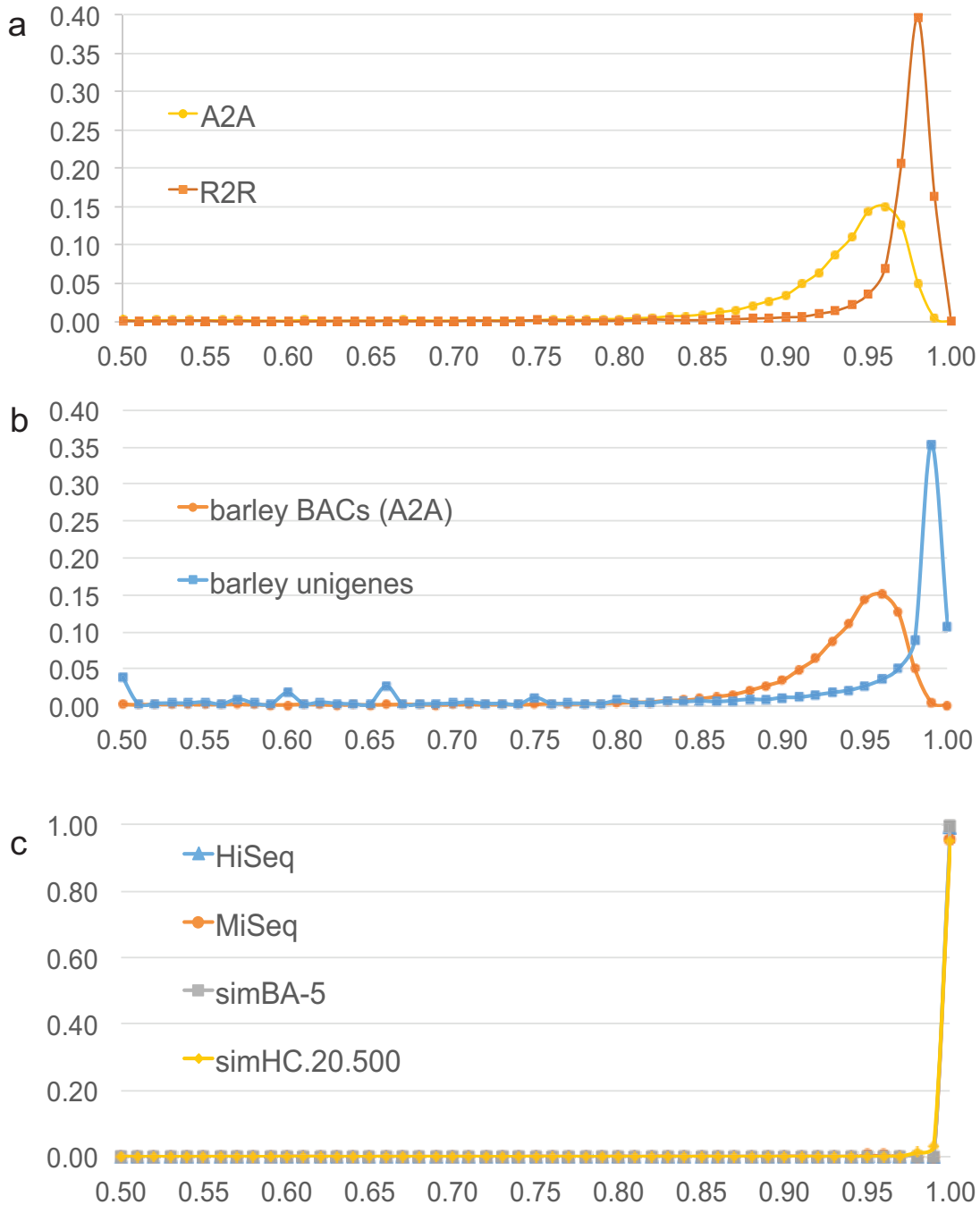

Figure 2.3: Distribution of the number of assignments as a function of the confidence score for (a) barley BACs (R2R) and (A2A) (b) barley unigenes and BACs (A2A) and (c) the four simulated metagenome sets ("HiSeq", "MiSeq", "simBA-5", and "simHC.20.500").

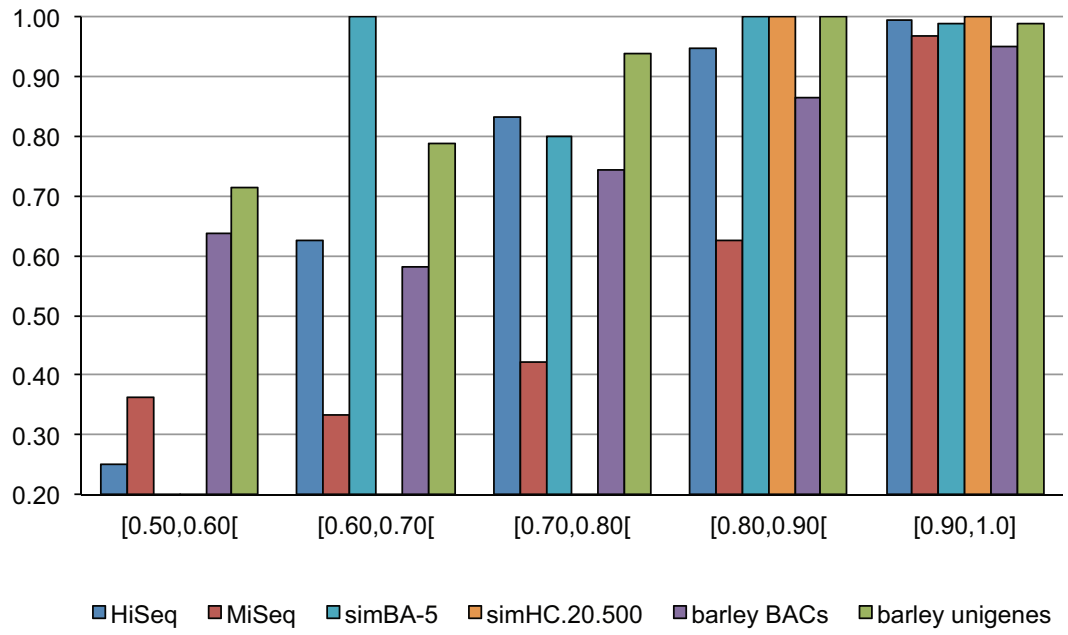

Figure 2.4: Probability (y-axis) of a correct assignment for a particular range of CLARK's confidence scores (x-axis).

Chapter 3

A higher classification sensitivity for short metagenomic reads

3.1 Introduction

In this chapter, we present a new approach to improve CLARK's classification sensitivity. The new approach exploits the concept of (discriminative) spaced k-mers. We first describe the notion of spaced k-mers, then discuss how these spaced k-mers are implemented into a new classification tool called CLARK-S (S for "spaced"), then finally compare the performance of CLARK-S against two of the most sensitive classifiers in the literature (i.e., NBC and KRAKEN), on several simulated/real metagenomic datasets.

We show that at the phylum, genus and species level CLARK-S outperforms the best state-of-the-art methods, including the default variant of CLARK, NBC and KRAKEN on all metrics.

3.2 Classification by discriminative spaced k-mers

3.2.1 Preliminaries

The concept and the utility of spaced seeds were initially described in context of sequence-comparison [Burkhardt and Kärkkäinen, 2001, Ma et al., 2002]. A *spaced seed* s is a string over the alphabet $\{1,*\}$, where '1' indicates that one should sample that position while '*' indicates that position should be ignored. The number of symbols in s is the *length* $|s|$ of s, while the number of 1s in s is the *weight* of s. A *spaced k-mer* is a spaced seed of length k. Let s be a spaced k-mer and weight w, and let m be a text of length k. We define $s(m)$ be the w-mer obtained from m using only the positions in s denoted by a 1. For example, if the text $m =$`AAGTCT` and $s =$`11*1*1` ($k = 6, w = 4$) then $s(m) =$`AATT`. The same text processed using the spaced 6-mer $s =$`1*11*1` would give the 4-mer $s(m) =$`AGTT`.

The work of Ma *et al.* in [Ma et al., 2002, Li et al., 2004] demonstrated that the use of single (and multiple) spaced seeds/k-mers significantly increased the chance of detecting a valid sequence alignment between the query and the target compared to contiguous seeds/k-mers, while incurring no additional computational cost. As a direct consequence of this work, spaced seeds are now used in the state-of-the-art homology search methods (e.g., BLAST [Altschul et al., 1990], MEGABLAST [Zhang et al., 2000]), but also protein alignment (e.g., DIAMOND[Buchfink et al., 2015]), or estimation of phylogenetic distances (e.g., [Leimeister et al., 2014, Morgenstern et al., 2015]). For more information about spaced seeds, we also refer the reader to [Brown et al., 2004, Li et al., 2004, Choi et al., 2004, Li et al., 2006, Ilie and Ilie, 2007, Ilie et al., 2011] and references therein.

Consider now the following problem: we are given a read r and two target sequences g_1

and g_2, and we want to know whether r is more likely to originate from g_1 or from g_2. As it is done in alignment-based homology search, we can use seeds/k-mers as "witnesses" of possible valid alignments. A time-efficient solution is to count the number shared k-mers between r and targets g_1 and g_2, and assign r to the target that has the highest count. As said, spaced seeds/k-mers increases the probability of detecting a valid alignment compared to contiguous seeds/k-mers. It is always possible, however, that a shared seed/k-mer (whether it is spaced or not) may be a false positive. In order to compensate for false positives, we use discriminative spaced k-mers, as described next.

3.2.2 Discriminative spaced k-mers

Given a set of reference sequences (or *targets*) $\{g_1, g_2, \ldots, g_p\}, i \in \{1, 2, \ldots, p\}$, the set D_i of discriminative k-mers for target g_i is the set of all k-mers in g_i that do not appear in any other reference sequences (as defined in previous chapter). Given a spaced seed s of length k and weight w, we define $D_{i,s}$ to be the set of all w-mers obtained via s from k-mers in D_i. We then define the set $E_{i,s}$ of discriminative spaced k-mers as the set of all w-mers of $D_{i,s}$ that do not appear in any set $D_{j,s}$ where $j \neq i$. Thus, any w-mer in $E_{i,s}$ is a spaced k-mer of weight w that can be found in one and only one target.

As stated earlier, the concept of spaced k-mers is not new. In metagenomics, several popular metagenome analysis tools, such as MEGAN [Huson et al., 2007, 2011], METAPHYLER [Liu et al., 2011] or PHYMMBL [Brady and Salzberg, 2011], as BLAST-based methods, have been using spaced seeds. In addition, other similarity-based methods that analyze genomic and metagenomic sequences use spaced k-mers, such as SEED [Bao et al., 2011]. However, to the best of our knowledge, the concept of "discriminative spaced k-mers" is novel and introduced only in this work.

3.2.3 Selection of optimal spaced seeds and index creation

The selection of specific spaced seed is critical to achieve high precision and sensitivity (see, e.g., [Ma et al., 2002, Brown et al., 2004, Li et al., 2004, Choi et al., 2004, Li et al., 2006, Ilie and Ilie, 2007, Ilie et al., 2011]). In order to determine the optimal structure we proceeded to model sequence similarly as it is done in alignments-based method (see, e.g., [Ma et al., 2002]). We considered that the succession of matches/mismatches follows a Bernoulli distribution with parameter p, where p represents the similarity level between the read and the reference sequence. If a short read belongs to a known reference sequence, then the similarity level should be high since the amount of mismatches dues to genomic variations or sequencing errors are low.

Finding an optimal set of spaced seeds through w, k and p is computationally difficult [Li et al., 2004, Brown et al., 2004], thus we decided to reduce the space search by judiciously setting w, k and p. For contiguous k-mers, the classification precision increases as we increase k. However, the highest sensitivity occurs with somewhat shorter k-mers. CLARK is more precise for long contiguous k-mers (e.g., $k = 31$), but the highest sensitivity occurs for k-mers of length 19–22 [Ounit et al., 2015]. As a consequence, we considered here spaced seeds of length $k = 31$ and weight $w = 22$. The choice of selecting a length of 31 is also motivated by a fair comparison against CLARK and KRAKEN, which achieve high accuracy thanks to long 31-mers in their default mode. Here our intent is to show the advantage of replacing discriminative contiguous seed with discriminative spaced seed(s). Then, we set p=0.95 to reflect the expected high similarity between genomic sequences at the species rank.

We searched exhaustively through all the spaced seeds of length $k = 31$ and weight $w = 22$ (starting/ending with '1') using a similarity level of 95%, and a random region of length

100bp, by using the dynamic programming approach from [Ma et al., 2002] and implemented in [Ilie et al., 2011]. The spaced seed with the highest hit probability [Ma et al., 2002], 0.998113, is `1111*111*111**1*111**1*11*11111`. In addition, we have also selected two additional spaced seeds with the highest hit probability namely `11111*1**111*1*11*11**111*11111` (0.998099) and finally `11111*1*111**1*11*111**11*11111` (0.998093). We have shown in [Hahn et al., 2016] that these three spaced seeds provides a high performance (in terms of precision, sensitivity and speed), which suggests that it is close to the optimal solution. Before a read can be classified, CLARK-S builds a database of discriminative spaced k-mers for each target. CLARK-S can take advantage of multiple spaced seeds, thus multiple databases can be created. For each spaced seed, discriminative spaced k-mers were built from contiguous discriminative 31-mers. Once the three databases of discriminative spaced k-mers were computed, they are stored in disk so they can be loaded for classification.

The classification algorithm of the "Spaced" variant is identical to that of the "Full" mode (extensively described in the previous chapter), except for two differences, namely (i) CLARK-S queries against discriminative spaced k-mers instead of discriminative k-mers and (ii) CLARK-S does three queries for each k-mer in a read, because there are three different databases. Finally, as done in the default variant of CLARK, the read is assigned to the target that has the highest amount of successful queries, and several statistics (such as the confidence score and gamma score, see previous chapter) are computed as well.

3.3 Results at the Genus and Phylum level

3.3.1 Datasets

To evaluate numerically the performance of the classifiers we used simulated datasets. From the available literature, we have selected the following three simulated metagenomes, which we made available at `http://clark.cs.ucr.edu/Spaced/` The first dataset is "A1.10.1000" which was derived from "A1", the first group of paired-end reads in the dataset "A" from [Lindgreen et al., 2016]. According to authors, this dataset closely mimics the complexities, size and characterization of real metagenomes. The A1 dataset contains about 28.9 M reads, 80% of which correspond to known sequenced genomes (from bacterial, archaeal and eukaryotes genomes), and 20% of which are randomized reads (from real genomes) that should not be assigned to any taxa. We have extracted 10,000 reads from A1 as follows. We have arbitrarily taken nine different genomes from the list of genomes used to build "A1" (see Supplementary Table 1 in [Lindgreen et al., 2016]). Then, we took the first 1,000 reads for each selected genome, and also 1,000 "random" reads. The resulting dataset, called "A1.10.1000", contains 10,000 reads (each 100 bp long) and can be considered as medium/high complexity.

The second dataset is "B1.20.500" which was derived from "B1", the first group of reads in the dataset "B", from [Lindgreen et al., 2016]. Similarly as done for A1.10.1000, we have extracted 10,000 reads from B1 as follows. We have arbitrarily taken 19 different genomes from the list of genomes used to build "B1" (see Supplementary Table 2 in [Lindgreen et al., 2016]). Note that these 19 selected genomes are different from those selected in A1. Then we took the first 500 reads for each selected genome, and also 500 "random" reads. The resulting dataset, called "B1.20.500", contains 10,000 reads (each 100 bp long) and can be considered as medium/high complexity.

The third dataset "simBA-5" comes from the paper that describes KRAKEN. According to the authors, it was created using bacterial and archaeal genomes, and with an error rate five times higher than the default. It contains 10,000 reads, each read is 100 bp long, and can be considered as high complexity.

To classify these metagenomic datasets, we use the entire set of bacterial/archaeal genomes from NCBI/RefSeq as reference genomes. At the time of writing, they represent 2,644 genomes and distributed in 36 phyla. The cumulative length of these genomes is 9.1 Gbp, and the average genome length is 3.4 Mbp.

3.3.2 Comparison with other tools

A large set of metagenomic classifiers exists in the literature. However, a comparison between CLARK and all existing classifiers is not necessary. An independent comprehensive evaluation of a wide range of metagenomics classifiers has been carried out recently using six large datasets of short paired-end reads [Lindgreen et al., 2016]. On the data tested, KRAKEN is among the most accurate methods at the phylum level compared to other popular and used methods, such as MOTU [Sunagawa et al., 2013], METAPHLAN2 [Segata et al., 2012, Truong et al., 2015], METAPHYLER or MEGAN. However, the experimental results in [Wood and Salzberg, 2014] shows that NBC is more sensitive than KRAKEN, MEGABLAST and PHYMMBL at the genus level. In the previous chapter, we have also shown that NBC is more sensitive than KRAKEN at the genus level. In addition, NBC is more sensitive than CLARK, at the genus level, even when the latter is run in its most sensitive settings (*i.e.*, "Full" mode and $k = 20$) [Ounit et al., 2015]. Note that the study [Bazinet and Cummings, 2012] also shows the high sensitivity of NBC. As a consequence of this analysis, it appears sufficient to compare CLARK against NBC and KRAKEN, as they are the two

most accurate classifiers among current published methods, at the phylum and genus level.

3.3.3 Classification accuracy

In this section, we present the performance of CLARK, NBC (v1.1) and KRAKEN (v0.10.5-beta) on the three simulated datasets described above. Consistently with other published studies (e.g., [Ounit et al., 2015],[Wood and Salzberg, 2014] or [Bazinet and Cummings, 2012]), the sensitivity is defined as the ratio between the number of correct assignments at a given taxonomy rank (e.g., phylum or genus) and the number of reads defined for that rank. The precision is defined as the ratio between the number of correct assignments at a given taxonomy rank (e.g., phylum or genus) and the number of assigned reads.

We present below results for the phylum and genus level. In Table 3.1 and Table 3.2, the first three rows report results from KRAKEN, CLARK, and NBC, all ran with default/recommended parameters. We ran KRAKEN and CLARK in the default mode, with $k = 31$, and NBC, with $k = 15$. The last two rows report the performance of CLARK-S. In the last row we report the precision and sensitivity when filtering only high confidence (HC) assignments (*i.e.*, assignment with confidence score ≥ 0.75 and gamma score ≥ 0.03).

Observe in Table 3.1 that (i) CLARK-S (HC) and NBC achieve very high sensitivity, (ii) KRAKEN's sensitivity is lower than NBC or CLARK-S for all datasets, (iii) CLARK-S outperforms NBC's sensitivity in A1.10.1000 and B1.20.500, (iv) both CLARK and KRAKEN have high precision and achieve more than 99.9% in all datasets (even though A1.10.1000 and B1.20.500 contain reads that do not belong to any bacterial/archaeal genomes), but (v) CLARK-S (HC) is as precise as them and outperforms NBC in all datasets. In Figure 3.1, 3.2 and 3.3 we report the performance of the tools for the dataset A1.10.100, B1.20.500 and simBA-500 respectively.

Table 3.1: Phylum-level accuracy (%) of KRAKEN, NBC, CLARK, CLARK-S and CLARK-S (HC) on A1.10.1000, B1.20.500 and simBA-5

	A1.10.1000		**B1.20.500**		**simBA-5**	
	Precision	*Sensitivity*	*Precision*	*Sensitivity*	*Precision*	*Sensitivity*
KRAKEN	99.91	77.59	99.98	90.91	99.98	94.49
CLARK	**99.93**	76.87	**100.00**	90.12	99.99	93.46
NBC	79.86	79.86	94.91	94.91	99.89	**99.89**
CLARK-S	94.50	**79.99**	98.95	**94.98**	99.87	99.70
CLARK-S (HC)	99.63	79.97	99.99	94.93	**100.00**	99.29

Table 3.2: Genus-level accuracy (%) of KRAKEN, NBC, CLARK, CLARK-S and CLARK-S (HC) on A1.10.1000, B1.20.500 and simBA-5

	A1.10.1000		**B1.20.500**		**simBA-5**	
	Precision	*Sensitivity*	*Precision*	*Sensitivity*	*Precision*	*Sensitivity*
KRAKEN	**99.80**	70.61	99.94	90.55	**99.85**	91.97
CLARK	**99.80**	69.98	**99.95**	89.69	99.82	90.77
NBC	77.94	77.94	94.76	**94.76**	98.97	**98.97**
CLARK-S	92.71	**78.38**	98.76	94.74	98.58	98.22
CLARK-S (HC)	99.35	76.41	**99.95**	94.52	99.61	97.24

Table 3.2 shows that (i) CLARK's sensitivity is lower than NBC, (ii) CLARK-S (HC) and NBC achieve the highest sensitivity and outperforms KRAKEN, (iii) CLARK-S is more NBC in A1.10.1000, (iv) KRAKEN and CLARK show high precision and achieve both more than 99.8% in our datasets, (v) CLARK-S (HC) is as precise as KRAKEN and CLARK, it outperforms NBC in all datasets, especially for A1.10.100 or B1.20.500. For simBA-5, NBC achieves the best sensitivity with 98.97, less than 2% more than the level performed by CLARK-S (HC).

Given the performance of CLARK-S (HC) over CLARK-S, henceforth have set CLARK-S (HC) to be default implementation of CLARK-S.

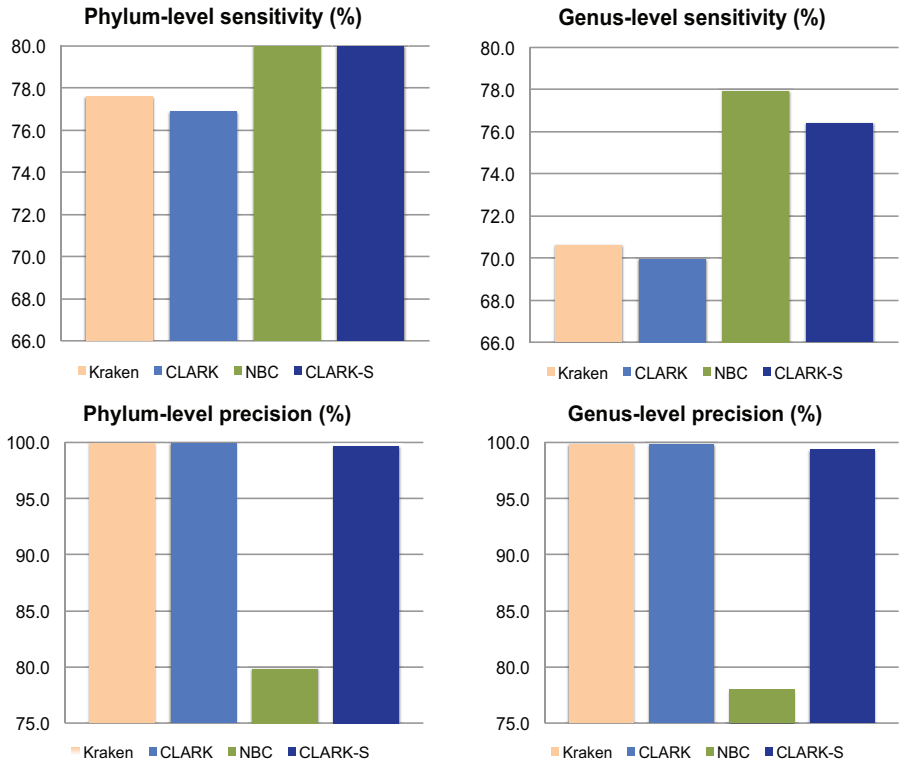

Figure 3.1: Precision and sensitivity of CLARK, NBC, Kraken and CLARK-S on the A1.10.1000 dataset

3.3.4 Real metagenomic samples

In this section, we evaluate the performance of CLARK-S (HC) on a real metagenomic dataset. We have selected the dataset from [Mueller et al., 2015], which is a recently published study on the population dynamics in microbial communities present in surface seawater in Monterey Bay, CA.

This dataset contains 42M reads, and the average read length is 510 bp. We pre-processed the dataset of raw reads using the following trimming steps: (i) we removed the first five bases and kept the following 100 bases using FASTQ Trimmer[1], (ii) we removed reads containing sequencing

[1] http://hannonlab.cshl.edu/fastx_toolkit/index.html

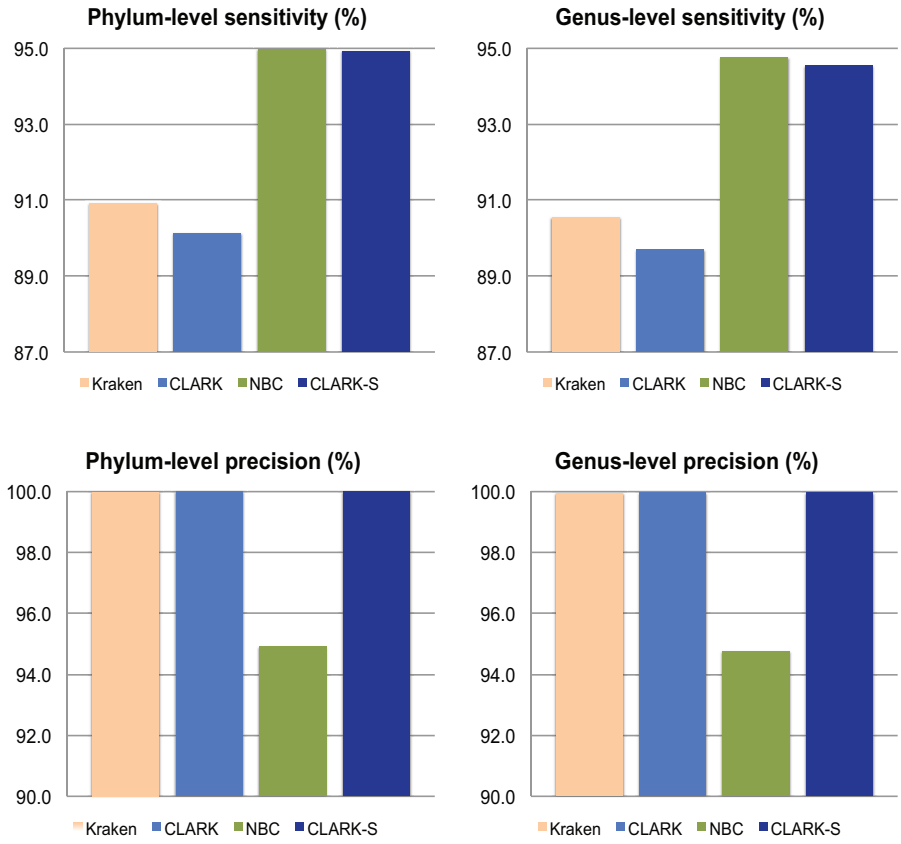

Figure 3.2: Precision and sensitivity of CLARK, NBC, Kraken and CLARK-S on the B1.20.500 dataset

adapters using Scythe[2], (iii) we trimmed the read ends if contained bases with a quality score below 30 and discarded reads containing any Ns using Sickle[3]. The resulting dataset contained 37M short reads.

We classified these 37M short reads using KRAKEN (default) and CLARK-S, using the bacterial/archaeal genomes from NCBI/RefSeq. KRAKEN was able to classify only 1.1 M reads (or 3% of the total). CLARK in its default mode also classifies about 1.1 M reads. However, CLARK-S classifies 20 M reads (or 54% of the total). Among these 20 M classified reads, there are 7 M high

[2]https://github.com/ucdavis-bioinformatics/scythe
[3]https://github.com/ucdavis-bioinformatics/sickle

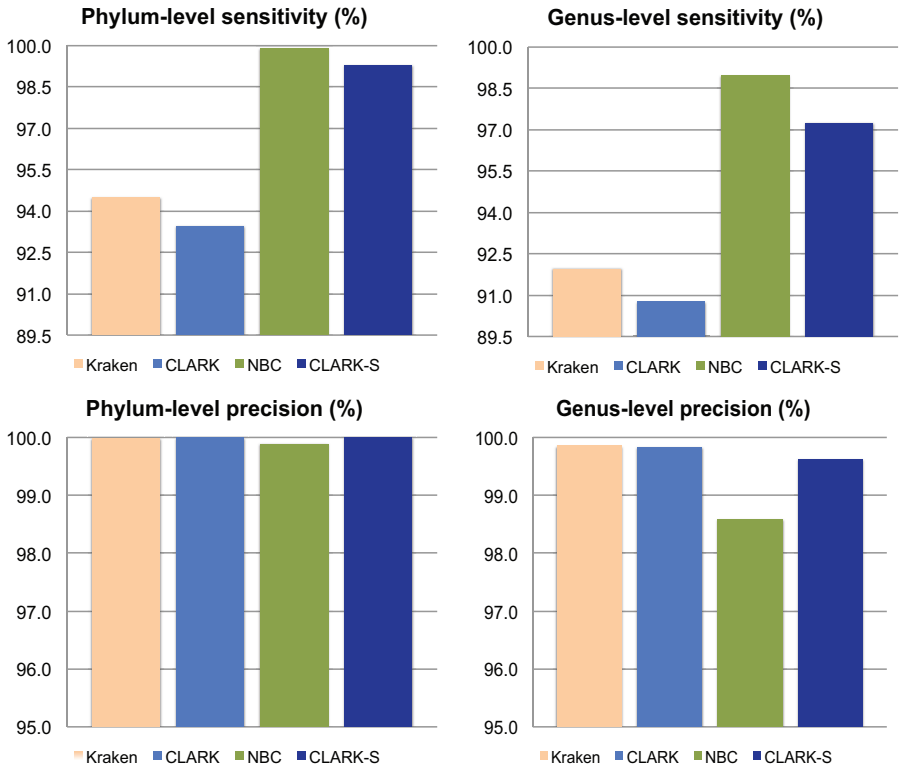

Figure 3.3: Precision and sensitivity of CLARK, NBC, Kraken and CLARK-S on the simBA-5 dataset

confidence assignments (or 19% of the total), which is about 6 times more than KRAKEN.

The fact that KRAKEN assigns only 3% of the reads can be explained by the fact that (i) KRAKEN relies on matching exact k-mer, and (ii) the current database of bacterial/archaeal likely contains only a limited fraction of the bacterial/archaeal diversity in seawater. Seawater metagenomes are likely to contain a high proportion of organisms that are missing in NCBI/RefSeq database because while the marine environment is one of the most biologically diverse on the planet [Felczykowska et al., 2012], the culture in laboratory of bacteria from seawater is difficult [Pace, 2009]. Since CLARK-S allows mismatches on the k-mers, it can identify at least the phylum/genus of unknown organisms.

KRAKEN identified, as dominant phyla, *Proteobacteria* (57%) and *Bacteroides* (27%). This is consistent with results reported in [Mueller et al., 2015], as well as phyla in low-abundance such as *Actinobacteria* (1%) or *Thaumarchaeota* (2%). Within high confidence assignments of CLARK-S, the two dominant phyla are, as expected by estimations from [Mueller et al., 2015], *Proteobacteria* (56%) and *Bacteroides* (32%). Consistently with [Mueller et al., 2015], phyla in low-abundance were correctly identified, for example, *Actinobacteria* (1%) and *Thaumarchaeota* (2%).

Experimental results from KRAKEN and CLARK-S (HC) indicate the expected dominant phyla in the dataset (with the expected abundance for each). While KRAKEN and CLARK-S (HC) are consistent for this dataset, we do notice one significant disagreement. The expected abundance of *Cyanobacteria* is 0–2%, according to [Mueller et al., 2015], but KRAKEN reports 9% and CLARK-S (HC) reports 3%. Such discrepancies can be explained by our pre-processing to create this dataset, however, the estimation by CLARK-S (HC) is more accurate than KRAKEN. As a consequence, CLARK-S was able to assign about 20 times more short reads than KRAKEN, and its high confidence assignments show stronger consistency with expected results than KRAKEN's results.

3.3.5 Time and space complexity

All experiments presented in this study were run on a Dell PowerEdge T710 server (dual Intel Xeon X5660 2.8 Ghz, 12 cores, 192 GB of RAM). NBC's speed is the slowest at 8–9 reads per minute, KRAKEN's speed is 1.8–2M reads per minute, while CLARK (default mode) runs the fastest, at 2.8–3M reads per minute. However, CLARK-S runs slower than CLARK, and classifies about 150–200 thousand reads per minute. While CLARK is the fastest in the default mode, it does not provide the same classification accuracy of NBC or CLARK-S. The fact that CLARK-S

computes spaced k-mers and uses several spaced seeds explains this difference of speed. However, CLARK-S is still several thousand of times faster than NBC.

NBC consumed less than 500MB of RAM, while CLARK and KRAKEN used 70 and 77GB respectively. Finally, CLARK-S used 110GB. This larger RAM usage is due to the multiple databases corresponding to the three spaced seeds. However, this amount remains significantly lower than 160GB, which is the amount needed to build/construct the database of discriminative k-mers.

3.4 Results at the species-level

3.4.1 Introduction

The species-level is the most important taxonomic rank for the analysis of metagenomes, and yet, because of the high similarity between several species (from the same genus or not), the evaluation of the performance of metagenomic classifiers at the species-level requires a higher degree of precaution and confidence than for higher taxonomy rank such as Phylum or Genus.

We evaluated the performance of CLARK-S against the state-of-the-art methods through the use of i) a set of synthetic and real metagenomic samples from various microbial habitats, ii) negative control samples iii) synthetic samples that can enable an accurate evaluation regardless of the fact at the species-level, several targets can be *locally* identical even if they are globally different.

3.4.2 Experimental setup

As said, a recent independent evaluation of several published taxonomic binning methods showed that CLARK and Kraken are the two most accurate tools at the genus and phylum level

[Lindgreen et al., 2016]. Instead of comparing CLARK-*S* to all published binning methods, it is therefore sufficient to compare it against CLARK and Kraken. In collaboration with the Mason group at Cornell, we have compared CLARK-S against seven state-of-the-art reads classifiers (including KRAKEN, LMAT, MEGAN or NBC) and we have showed that CLARK-S outperforms all other tools in the majority of the datasets used.

To guarantee a consistent and fair evaluation, we ran CLARK-*S*, CLARK and Kraken on the same set of reference genomes, namely all microbial genomes in the NCBI/RefSeq database (total of 5,747 species: 1,335 bacteria, 123 archaea and 4,289 viruses). Evaluations were carried out on simulated datasets and real metagenomic data.

We created six synthetic datasets, each representing a distinct microbial habitat and containing reads from the related dominant organisms. We included samples from the human mouth (characterized by 12 dominant species), city parks (48 species), human gut (20 species), household (two datasets, 31 and 21 species) and soil (50 species). A seventh dataset included reads from 525 randomly chosen bacterial/archaeal species. All these datasets are composed of 100bp reads generated by ART [Huang et al., 2012] using the Illumina error model (HiSeq) with default settings. We have chosen the Illumina HiSeq because it is a leading sequencing technology [Levy and Myers, 2016][4] at the time of writing. Since two distinct species i and j can have sequence similarity as high as 98.8% [Stackebrandt and Goebel, 1994], a short read r generated from genome g_i may appear in another genome g_j for a given error rate or number of mismatches. Ignoring he possibility of ambiguity in reads classification is likely to lead to incorrect conclusions on precision and sensitivity. In order to carry out an unbiased evaluation, we created additional datasets (called "unambiguous",

[4] According to authors of [Levy and Myers, 2016], "the Illumina HiSeq X system remains the highest-output platform and the only sequencing technology available that can generate highly accurate data that allow sequencing at the human genome scale at reagent costs under USD 1,000."

see next paragraph for details) in which no read can be mapped to more than one species with the same error rate or number of mismatches. We tested the three tools on fourteen datasets containing a total of 23.5 M reads from 647 species (see Table 3.3). We also added three negative control samples containing short reads that do not exist in any genome in the NCBI/RefSeq database (see next paragraph). We used the precision and sensitivity metrics defined in the previous chapter to evaluate the classification performance.

For experiments on real metagenomes, we chose a large dataset from a recent study on the microbial profile of the NY City subway system, the Gowanus canal and public parks [Afshinnekoo et al., 2015]. We selected twelve datasets containing a total of 105 M reads from various microbial habitat (e.g., bench, garbage can, kiosk, stairway rail, water, etc.), subway stations and riders usage (see Table 3.4). While the ground truth for these data is unknown, the abundance of bacteria, eukaryotes and viruses present in these samples were provided in Afshinnekoo et al., 2015. We trimmed[5] raw reads as it was done in Afshinnekoo et al., 2015 (see Table 3.4), then compared the results of CLARK-*S* with the findings in Afshinnekoo et al., 2015 (see Table 3.5 and 3.6).

Generation of synthetic datasets and negative controls

In this paragraph, we describe how we created the synthetic datasets used for the evaluation of the three tools we tested. To produce synthetic reads we considered organisms reported to be present in real microbial habitats by different published studies. We consider the habitats related to mouth, city parks/medians, gut, indoor and soil (listed below).

- Buc12: As reported in [Franzosa et al., 2015, Human Microbiome Project Consortium , 2012],

[5]Raw reads were trimmed as done in [Afshinnekoo et al., 2015]: the first/last 10 bp each read were removed (reads longer than 100 bp were truncated and the first 100 bp were kept); trimmed reads with more than 10 bp with quality scores less than 20 were removed.

the dominant genus found in the oral cavity is *Streptococcus*. [Franzosa et al., 2015] also reports the presence of the *Haemophilus influenzae*, *Haemophilus parainfluenzae*, *Neisseria subflava* and *Veillonella dispar*. Thus, we chose these four species along with eight species selected from the *Streptococcus* genus.

- CParMed48: Forty-eight species were selected from *Proteobacteria*, *Acidobacteria*, *Bacteroides*, *Actinobacteria* and *Planctomycetes*. These are the dominant phyla reported in [Reese et al., 2015] in city parks and medians in Manhattan.

- Gut20: This dataset contains the twenty species described in the Supplementary Table 1 of [Kuleshov et al., 2016].

- Hous31: Bacteria typically found indoor are *Streptococcaceae*, *Lactobacillaceae*, and *Pseudomonadaceae* (due to human activities), and also *Intrasporangiaceae* and *Rhodobacteraceae* (due to the environment), as reported in [Ruiz-Calderon et al., 2016]. We selected randomly thirty-one species from these microbial families.

- Hous21: We selected twenty-one species from the dominant organisms reported in [Adams et al., 2015] found in the bathroom and kitchen, namely *Propionibacterium acnes*, *Corynebacterium*, *Streptococcus* and *Acinetobacter*.

- Soi50: We selected fifty species from the dominant genera reported in [Fierer et al., 2012], namely *Acidobacteria*, *Actinobacteria*, *Bacteroides*, *Proteobacteria* and *Verrucomicrobia*.

- A seventh dataset called simBA-525 containing reads randomly selected from 525 bacterial/archaeal species was also added.

Datasets generation. We obtained reference genomes from the NCBI/RefSeq database (about 650 billion of nucleotides, containing more than 57,000 genomes distributed in 14,675 species, downloaded on February 9, 2016), then we used the ART read simulator [Huang et al., 2012] to create synthetic reads from the list of species listed above. We ran ART with default quality base profile and error parameters, length 100 bp, and coverage 30x. These seven datasets represent a total of 647 species (see Table 3.3 for statistics on these datasets).

Negative control samples. To generate negative controls, we created three datasets (named "LM", "MH1", "MH2") composed of reads that do not exist in any genomes in the NCBI/RefSeq database (see Table 3.3). To build these datasets, observe that if a DNA fragment of 100 bps contains at least one k-mer that does not appear in any genomes in the full NCBI/RefSeq database then it does not exist in any of these genomes. In other words, if each read contains one unassigned k-mer for the full NCBI/RefSeq database then the read does not map without mismatches (we used $k = 17$). Based on this idea, we generated 10 M 100 bp random reads, using a uniform random distribution for each of the four nucleotides (i.e., each nucleotide has probability 1/4). We also built an index of 17-mers from all genomes in the complete NCBI/RefSeq database. Using this index, we counted the number of unknown 17-mers in each random read. Then, we stored 1 M read that contained at least five unknown 17-mers in dataset "LM", one million read that contained exactly four unknown 17-mers in dataset "MH1", and one million read that contained exactly three unknown 17-mers in dataset "MH2".

Datasets of unambiguously mapped reads. To create datasets of unambiguously mapped for each of these seven datasets, we used the method described next.

Generating datasets of unambiguously mapped reads

In this paragraph, we describe how we identified and removed ambiguously mapped read from the set of reads generated by ART.

Definitions and notations. Given a string x, let $|x|$ denote its length. In the following definitions, we assume that k is a positive integer (length of the k-mers), r is a read, and G is a genome. Given a set of genomes $\{G_1, G_2, \ldots, G_m\}$, a k-mer T is *specific* to G_i if T occurs in G_i (exactly) but T does not occur (exactly) in any other genome G_j, when $j \neq i$ (see [Ounit et al., 2015]). Given a set K of k-mers specific to G, the number of nucleotides of read r covered by at least one k-mer in K is called the *coverage* of r to G which we denote by $cov(r, G)$. Given a position $l \in [1, |G| - |r| + 1]$, we denote by $M(r, G, l)$ the number of mismatches (Hamming distance) between read r and a substring of G of length $|r|$ starting at position l. We denote by $OPT(r, G) = min_{l \in [1,|G|-|r|+1]} M(r, G, l)$, i.e., the minimum number of mismatches for all possible positions of r in G. Given a set of genomes $\{G_1, G_2, \ldots, G_m\}$, read r is *unambiguously mapped* to G_i if and only if for all $j \neq i$ we have that $OPT(r, G_i) < OPT(r, G_j)$. In other words, there is no pair of genomes (G_i, G_j) such that the two optimal alignments of r to G_i and G_j achieves the same number of mismatches.

Lemma 5. *Given a read r and a set of genomes $\{G_1, G_2, \ldots, G_m\}$, if there exists an index $i \in [1, m]$ and a position $l \in [1, |G_i| - |r| + 1]$ such that $\lfloor cov(r, G_i)/k \rfloor > M(r, G_i, l)$ then for all $j \neq i$, we have that $OPT(r, G_j) > OPT(r, G_i)$.*

Proof. By the definition of k-mer specific to a genome: for each non-overlapping block B of k nucleotides that are covered by at least one k-mer specific to G_i in r, there exists at least one mismatch between block B and any block of k nucleotides in G_j where $i \neq j$. Since there is at least

$\lfloor cov(r, G_i)/k \rfloor$ non-overlapping block(s) of k nucleotides covered by at least one k-mer G_i-specific in r, for all $j \neq i$ we have that $OPT(r, G_j) \geq \lfloor cov(r, G_i)/k \rfloor$.

By the hypothesis of the Lemma, there exists $l \in [1, |G_i| - |r| + 1]$ so $\lfloor cov(r, G_i)/k \rfloor > M(r, G_i, l)$. By the definition of OPT, we always have $OPT(r, G_i) \leq M(r, G_i, l)$. Then, for all $j \neq i$, $OPT(r, G_j) \geq \lfloor cov(r, G_i)/k \rfloor$ and $\lfloor cov(r, G_i)/k \rfloor > M(r, G_i, l)$ imply that $OPT(r, G_j) \geq \lfloor cov(r, G_i)/k \rfloor > M(r, G_i, l) \geq OPT(r, G_i)$. Thus, for all $j \neq i$, we have that $OPT(r, G_j) > OPT(r, G_i)$. ■

In other words, if $\lfloor cov(r, G_i)/k \rfloor$ is higher than the number of mismatches between r and G_i then read r is unambiguously mapped to G_i.

Generating unambiguously mapped reads. We used the ART read simulator to create simulated datasets. We considered the species rank, so genomes of the same species were considered together as a unique sequence. We set $k = 19$ to determine sets of k-mers specific to each species (i.e., 14,675 sets), then we created a hash-table to extract all 19-mers from all species and removed all 19-mers that were common to at least one pair of species. To create a dataset of unambiguously mapped reads, we filtered reads as follows. For each species G of a given dataset, and for each read r created, we used the alignment (provided by ART) of r to its reference sequence of origin. We computed the number of mismatches M between r and G, and we estimated the specificity-coverage C of r to G. Using the previous lemma, r was added to the unambiguous variant of the dataset (because it is unambiguously mapped to G) if the value C/k was higher than $M + 1$.

In this step, we addressed the issue of reads that were generated from a genome A but could also occur in another genome B. If a tool assigns those reads to B, should this be considered an incorrect classification? The amount of ambiguity depends not only on the dataset, but also

Table 3.3: Number of reads and species in each synthetic datasets (default and unambiguous) and for the negative controls.

Synthetic datasets	Buc12	CParMed48	Gut20	Hou31	Hou21	Soi50	simBA-525
Species	12	48	20	31	21	50	525
Reads (default)	600,000	1,200,000	500,000	775,000	525,000	2,500,000	5,666,143
Reads (unambiguous)	600,000	1,200,000	500,000	750,000	500,000	2,500,000	5,727,654

Synthetic datasets	HM1	HM2	LM
Species	0	0	0
Reads	1,000,000	1,000,000	1,000,000

Table 3.4: Metadata of the selected real samples from [Afshinnekoo et al., 2015]: Sample ID, number of raw reads, number of reads after trimming, object swabbed, location of the sample, borough name, and the number of weekly riders in 2013.

Sample ID	Raw reads	Trimmed reads	Object swabbed	Location	Borough	Weekly riders
GC01	29,282,945	28,739,916	Water Sample	Gowanus Canal	Brooklyn	NA
P00090	3,161,196	3,085,871	Stairway rail	Times Sq-42 St/42 St	Manhattan	197,696
P00302	12,206,080	11,700,388	Bench	59 St-Columbus Circle	Manhattan	72,236
P00306	7,536,640	7,194,993	Kiosk	34 St-Penn Station	Manhattan	90,042
P00454	7,872,512	7,555,783	Bench	Fulton St	Manhattan	64,461
P00589	3,129,344	3,015,949	Turnstile	Broadway-Lafayette St/Bleecker St	Manhattan	38,799
P00720	6,833,000	6,536,830	Bench	Franklin St	Manhattan	5,825
P00945	7,530,914	7,257,415	Bench	Forest Av	Queens	4,103
P01041	1,171,456	1,160,282	Bench	Van Siclen Av	Brooklyn	2,974
P01136	6,417,114	6,220,889	Garbage Can	Jefferson St	Brooklyn	6,612
P01270	17,072,185	16,471,331	Seats	F Train	Brooklyn	NA
P01324	2,686,976	2,594,672	Garbage Can	Whitlock Av	Bronx	1,685

on the set of reference genomes used to classify. This ambiguity introduces reference-dependent bias that can affects precision and sensitivity. While we are aware that these datasets might not be considered realistic, removing ambiguous reads allow us to have an unambiguous ground truth that allows to compare across tools without bias.

3.4.3 Experimental Results

Synthetic samples

Observe in Table 3.8 that the sensitivity achieved by CLARK-*S* on the fourteen simulated datasets is consistently higher than other tools, while maintaining high precision (the increase in sensitivity is even higher on unambiguous datasets).

Observe that the increase of sensitivity is sometimes followed by a decrease in precision, especially on the default datasets. However, several facts should be taken into consideration. Observe that while increasing precision is relatively easy (it is sufficient to be very strict in assigning a read), increasing sensitivity is much harder. That's why here we focus on the sensitivity because this metric is much more difficult to maximize than precision. We show that CLARK-S achieves higher sensitivity than CLARK or KRAKEN sometimes at the cost of a slightly lower precision (because in order to achieve a higher sensitivity, CLARK-S must assign more reads than the other tools, which can lead to more wrong assignments). The fact that CLARK-S achieves higher sensitivity than CLARK or KRAKEN implies that it can classify reads that the other tools are unable to identify. We hypothesize that the reads that CLARK and KRAKEN fail to identify are likely to be ambiguous reads (i.e., they can be assigned to multiple species). High sequence similarity can often exists between two species (more than 98% similarity [Mende et al., 2013]). This is why we have designed the datasets of unambiguously mapped reads. As shown in Table 3.8, CLARK-S outperforms CLARK and KRAKEN in both sensitivity and precision for several unambiguous datasets.

Also, note that CLARK-*S* did not classify any reads from the negative control samples as expected. Table 3.9 shows that CLARK-*S* classifies about 200 thousand short reads per minute

(using one CPU), while CLARK classifies about 3.5 M short reads per minute. If one can take advantage of eight cores, CLARK-*S* classifies about 1 M short read per minute, which is sufficiently fast to process large metagenomic datasets in few minutes. Finally, consistently with our previous experimental results at the Genus and Phylum level, CLARK-S requires more time to build the database than CLARK or Kraken, and its RAM usage is comparable to the other tools.

Real samples

Observe in Table 3.9 that CLARK-*S* classifies more reads than CLARK or Kraken. On average, CLARK-*S* classifies 10% more reads than Kraken, and 27% more reads than CLARK. Table 3.6 indicates the reads count assigned by each tool to each species listed in Afshinnekoo et al., 2015 and present in the database. CLARK-*S* achieves consistently the highest agreement with Afshinnekoo et al., 2015 on all samples. For instance, in P00589 and P00720, CLARK-*S* detected the presence of the virus *Enterobacter* phage HK97 but CLARK/KRAKEN did not; in sample P01136, CLARK-*S* detected *Brucella ovis* but CLARK/KRAKEN failed to do so.

In general, CLARK-S identified more relevant organisms than the other tested tools. A recent independent study [Thompson et al., 2017] showed that CLARK-S classifies more reads and detects more relevant organisms than other standard tools such as GRAFTM[6] or KRAKEN.

3.5 Conclusion

In this chapter, we have introduced for the concept of discriminative spaced k-mers for the classification problem of short metagenomic reads. To the best of our knowledge, CLARK-

[6]GraftM is a reads classifier available at `https://github.com/geronimp/graftM`. GRAFTM classifies reads based on HMM profiles and a reference phylogeny.

S is the only metagenome classifier using (multiple) discriminative spaced k-mers. Our extensive experiments on several realistic metagenomic samples show that i) CLARK-S can be as precise as (or more precise than) KRAKEN and as sensitive as NBC, ii) while CLARK-S is slower than CLARK because its uses multiple spaced seeds, it is still faster than NBC by several order of magnitude and it can classify a million short reads per minute with 8 CPU.

Finally, in the context of real metagenomic data (from seawater samples to urban samples), we showed that CLARK-S can classify with high accuracy a much higher proportion of short reads than CLARK or KRAKEN. This finding was corroborated by an independent study [Thompson et al., 2017] that showed that CLARK-S can detect more relevant organisms than other classification tools.

Table 3.5: List of species detected in [Afshinnekoo et al., 2015] which are also present in the database (i.e., bacteria/archaea/viruses genomes from NCBI/RefSeq) for each of the twelve samples.

Sample ID	Species in [Afshinnekoo et al., 2015] and present in the default NCBI/RefSeq database (bacteria/archaea/viruses)
GC01	*Bifidobacterium adolescentis, Bifidobacterium longum, Desulfobacterium autotrophicum, Erwinia billingiae, Eubacterium eligens, Eubacterium rectale, Methanocorpusculum labreanum, Parabacteroides distasonis*
P00090	*Acinetobacter baumannii, Cronobacter turicensis, Enterobacter cloacae, Enterococcus casseliflavus, Enterococcus faecalis, Klebsiella pneumoniae, Lysinibacillus sphaericus, Macrococcus caseolyticus, Micrococcus luteus, Pseudomonas putida, Pseudomonas stutzeri, Stenotrophomonas maltophilia, Streptococcus suis*
P00302	*Achromobacter xylosoxidans, Acinetobacter baumannii, Bacillus megaterium, Dickeya dadantii, Enterobacter cloacae, Enterococcus casseliflavus, Enterococcus faecalis, Enterococcus faecium, Enterococcus hirae, Finegoldia magna, Klebsiella pneumoniae, Lactococcus lactis, Leuconostoc mesenteroides, Lysinibacillus sphaericus, Micrococcus luteus, Propionibacterium acidipropionici, Propionibacterium acnes, Pseudomonas putida, Pseudomonas stutzeri, Staphylococcus epidermidis, Staphylococcus haemolyticus, Stenotrophomonas maltophili*
P00306	*Acinetobacter baumannii, Acinetobacter oleivorans, Enterobacter cloacae, Enterobacteria phage IME10, Enterococcus casseliflavus, Enterococcus faecium, Klebsiella pneumoniae, Propionibacterium acnes, Pseudomonas stutzeri, Stenotrophomonas maltophilia*
P00454	*Acinetobacter baumannii, Chlorobium phaeobacteroides, Enterobacter cloacae, Enterococcus casseliflavus, Enterococcus mundtii, Klebsiella pneumoniae, Lysinibacillus sphaericus, Pseudomonas stutzeri, Solibacillus silvestris, Stenotrophomonas maltophilia*
P00589	*Acinetobacter baumannii, Enterobacter cloacae, Enterobacteria phage HK97, Enterococcus casseliflavus, Lactococcus lactis, Pseudomonas putida, Pseudomonas stutzeri, Streptococcus suis*
P00720	*Corynebacterium variabile, Enterobacter cloacae, Enterobacteria phage HK97, Enterococcus casseliflavus, Lactococcus lactis, Leuconostoc citreum, Lysinibacillus sphaericus, Pseudomonas stutzeri, Stenotrophomonas maltophilia*
P00945	*Bacillus megaterium, Enterobacter cloacae, Enterococcus faecalis, Enterococcus faecium, Lysinibacillus sphaericus, Pseudomonas putida, Pseudomonas stutzeri, Stenotrophomonas maltophilia, Stenotrophomonas phage phiSMA7*
P01041	*Enterobacter cloacae, Enterobacteria phage HK97, Enterococcus casseliflavus, Enterococcus faecalis, Pseudomonas stutzeri, Stenotrophomonas maltophilia*
P01136	*Brucella ovis, Corynebacterium variabile, Enterobacter cloacae, Enterobacteria phage HK97, Enterococcus casseliflavus, Leuconostoc mesenteroides, Pseudomonas putida, Pseudomonas stutzeri, Stenotrophomonas maltophilia, Streptococcus suis*
P01270	*Achromobacter xylosoxidans, Enterobacter cloacae, Enterococcus casseliflavus, Enterococcus faecalis, Enterococcus faecium, Enterococcus hirae, Lactococcus lactis, Lysinibacillus sphaericus, Propionibacterium acnes, Pseudomonas putida, Pseudomonas stutzeri, Stenotrophomonas maltophilia*
P01324	*Cronobacter sakazakii, Enterobacter cloacae, Enterobacteria phage HK97, Enterococcus casseliflavus, Enterococcus faecium, Escherichia coli, Klebsiella pneumoniae, Kocuria rhizophila, Lactococcus lactis, Leuconostoc mesenteroides, Micrococcus luteus, Pseudomonas stutzeri, Rhodopseudomonas palustris, Stenotrophomonas maltophilia, Stenotrophomonas phage phiSMA7, Streptococcus parauberis, Streptococcus suis, Streptococcus thermophilus*

Table 3.6: Column A lists the reads count reported by KRAKEN, CLARK, and CLARK-S on the species listed in Table 3.5. For each species, a count is reported as a triplet (KRAKEN, CLARK, CLARK-S). Column B reports the agreement rate between [Afshinnekoo et al., 2015] and results reported by KRAKEN (first line), CLARK (second line), and CLARK-S (third line), in this order. For example, for the sample GC01, the agreement rate between KRAKEN and [Afshinnekoo et al., 2015] was 75% because KRAKEN detected the presence of 6 species out of the 8 species reported in [Afshinnekoo et al., 2015]. Column C reports the percentage of species for which CLARK-S reports a higher reads count than both KRAKEN and CLARK. For example, for the sample P00090, CLARK-S reports a higher number of reads count than both KRAKEN and CLARK for 12 species out of 13 (i.e., 92.3%).

Sample ID	A	B	C
GC01	*Bifidobacterium adolescentis (1238, 1218, 1307), Bifidobacterium longum (1106, 1093, 1217),* *Desulfobacterium autotrophicum (88171, 84690, 142189), Erwinia billingiae (8774, 8651, 9443),* *Eubacterium eligens (0, 0, 0), Eubacterium rectale (0, 0, 0),* *Methanocorpusculum labreanum (429, 400, 1091), Parabacteroides distasonis (1028, 1011, 1340)*	75% 75% 75%	100%
P00090	*Acinetobacter baumannii (8482, 8143, 14783), Cronobacter turicensis (2108, 2078, 1471),* *Enterobacter cloacae (44220, 41877, 64974), Enterococcus casseliflavus (14731, 14535, 16365),* *Enterococcus faecalis (2481, 2472, 2563), Klebsiella pneumoniae (49647, 49011, 49772),* *Lysinibacillus sphaericus (4, 4, 11), Macrococcus caseolyticus (1904, 1891, 2110),* *Micrococcus luteus (2686, 2646, 2990), Pseudomonas putida (8944, 8405, 12327),* *Pseudomonas stutzeri (1243301, 1228384, 1349618),* *Stenotrophomonas maltophilia (15162, 14732, 19712), Streptococcus suis (26495, 25484, 41016)*	100% 100% 100%	92.3%
P00302	*Achromobacter xylosoxidans (417007, 396787, 798804), Acinetobacter baumannii (53782, 51650, 84481),* *Bacillus megaterium (1291, 1263, 1619), Dickeya dadantii (8574, 8893, 6470),* *Enterobacter cloacae (328816, 303503, 497288), Enterococcus casseliflavus (9735, 9517, 12275),* *Enterococcus faecalis (20903, 20844, 21109), Enterococcus faecium (773, 757, 1045),* *Enterococcus hirae (1506, 1500, 1557), Finegoldia magna (314, 305, 505),* *Klebsiella pneumoniae (32826, 30878, 31901), Lactococcus lactis (911, 873, 1483),* *Leuconostoc mesenteroides (1890, 1853, 1965), Lysinibacillus sphaericus (1, 1, 1),* *Micrococcus luteus (781, 785, 879), Propionibacterium acidipropionici (379, 385, 413),* *Propionibacterium acnes (770, 767, 812), Pseudomonas putida (3493, 3452, 4770),* *Pseudomonas stutzeri (987112, 980445, 1011820), Staphylococcus epidermidis (661, 650, 771),* *Staphylococcus haemolyticus (1066, 1028, 1320), Stenotrophomonas maltophilia (50279, 48597, 72008)*	100% 100% 100%	86.4%
P00306	*Acinetobacter baumannii (540511, 520987, 731225), Acinetobacter oleivorans (67230, 66304, 72904),* *Enterobacter cloacae (171685, 159913, 272355), Enterobacteria phage IME10 (0, 0, 0),* *Enterococcus casseliflavus (54313, 53029, 67794), Enterococcus faecium (2675, 2649, 2910),* *Klebsiella pneumoniae (20732, 19474, 22448), Propionibacterium acnes (931, 925, 948),* *Pseudomonas stutzeri (533478, 525799, 585020),* *Stenotrophomonas maltophilia (564888, 560201, 586129)*	90% 90% 90%	100%
P00454	*Acinetobacter baumannii (46223, 45761, 48612), Chlorobium phaeobacteroides (1, 1, 147),* *Enterobacter cloacae (21652, 20137, 32217), Enterococcus casseliflavus (6931, 6852, 7405),* *Enterococcus mundtii (1112, 1101, 1151), Klebsiella pneumoniae (22895, 22507, 22950),* *Lysinibacillus sphaericus (1, 1, 3), Pseudomonas stutzeri (4711283, 4652107, 5004594),* *Solibacillus silvestris (2555, 2407, 4990), Stenotrophomonas maltophilia (43004, 41930, 53308)*	100% 100% 100%	100%
P00589	*Acinetobacter baumannii (7513, 7362, 9684), Enterobacter cloacae (2471, 2380, 3334),* *Enterobacteria phage HK97 (0, 0, 10), Enterococcus casseliflavus (11906, 11742, 13533),* *Lactococcus lactis (1743, 1699, 2578), Pseudomonas putida (6062, 5822, 8554),* *Pseudomonas stutzeri (777233, 765277, 850289), Streptococcus suis (8506, 8201, 13373)*	87.5% 87.5% 100%	100%

Table 3.7: **(Cont'd)** Column A lists the reads count reported by KRAKEN, CLARK, and CLARK-S on the species listed in Table 3.5. For each species, a count is reported as a triplet (KRAKEN, CLARK, CLARK-S). Column B reports the agreement rate between [Afshinnekoo et al., 2015] and results reported by KRAKEN (first line), CLARK (second line), and CLARK-S (third line), in this order. For example, for the sample GC01, the agreement rate between KRAKEN and [Afshinnekoo et al., 2015] was 75% because KRAKEN detected the presence of 6 species out of the 8 species reported in [Afshinnekoo et al., 2015]. Column C reports the percentage of species for which CLARK-S reports a higher reads count than both KRAKEN and CLARK. For example, for the sample P00090, CLARK-S reports a higher number of reads count than both KRAKEN and CLARK for 12 species out of 13 (i.e., 92.3%).

P00720	*Corynebacterium variabile (1302, 1262, 1487), Enterobacter cloacae (82530, 75880, 125426), Enterobacteria phage HK97 (0, 0, 48), Enterococcus casseliflavus (25280, 25059, 26621), Lactococcus lactis (2437, 2430, 2614), Leuconostoc citreum (498, 496, 511), Lysinibacillus sphaericus (26, 25, 49), Pseudomonas stutzeri (2738041, 2698911, 2989300), Stenotrophomonas maltophilia (516748, 501500, 671902)*	88.9% 88.9% 100%	100%
P00945	*Bacillus megaterium (760, 754, 771), Enterobacter cloacae (44780, 41433, 69336), Enterococcus faecalis (8984, 8954, 9128), Enterococcus faecium (1219, 1217, 1278), Lysinibacillus sphaericus (2, 0, 2), Pseudomonas putida (2505, 2340, 2920), Pseudomonas stutzeri (4149, 4157, 4849), Stenotrophomonas maltophilia (1258848, 1230418, 1589727), Stenotrophomonas phage phiSMA7 (397, 391, 637)*	100% 88.9% 100%	100%
P01041	*Enterobacter cloacae (13726, 12754, 20206), Enterobacteria phage HK97 (0, 0, 11), Enterococcus casseliflavus (5196, 5082, 6395), Enterococcus faecalis (2571, 2567, 2607), Pseudomonas stutzeri (611583, 608607, 626318), Stenotrophomonas maltophilia (58910, 58591, 60892)*	83.3% 83.3% 100%	100%
P01136	*Brucella ovis (0, 0, 12), Corynebacterium variabile (974, 965, 1005), Enterobacter cloacae (41486, 38925, 60976), Enterobacteria phage HK97 (0, 0, 16), Enterococcus casseliflavus (8871, 8783, 9460), Leuconostoc mesenteroides (896, 886, 909), Pseudomonas putida (49887, 47305, 56607), Pseudomonas stutzeri (1140608, 1101902, 1627874), Stenotrophomonas maltophilia (6588, 6425, 9192), Streptococcus suis (7045, 6768, 10659)*	80% 80% 100%	100%
P01270	*Achromobacter xylosoxidans (9129, 9013, 10142), Enterobacter cloacae (464185, 438737, 712806), Enterococcus casseliflavus (204915, 203223, 215280), Enterococcus faecalis (454647, 453560, 458843), Enterococcus faecium (5058, 4972, 6434), Enterococcus hirae (7299, 7264, 7588), Lactococcus lactis (2155, 2119, 2684), Lysinibacillus sphaericus (7, 6, 12), Propionibacterium acnes (341, 366, 351), Pseudomonas putida (1722194, 1623230, 3097829), Pseudomonas stutzeri (3177433, 3126518, 3511417), Stenotrophomonas maltophilia (1281605, 1248952, 1619141)*	100% 100% 100%	91.7%
P01324	*Cronobacter sakazakii (4237, 4016, 4891), Enterobacter cloacae (15067, 13986, 22082), Enterobacteria phage HK97 (0, 0, 2), Enterococcus casseliflavus (4685, 4553, 6638), Enterococcus faecium (533, 514, 783), Escherichia coli (2797, 2694, 4119), Klebsiella pneumoniae (2859, 2702, 3091), Kocuria rhizophila (84, 70, 178), Lactococcus lactis (1088, 1071, 1322), Leuconostoc mesenteroides (1042, 1036, 1089), Micrococcus luteus (162, 166, 173), Pseudomonas stutzeri (323280, 319408, 343408), Rhodopseudomonas palustris (370, 354, 422), Stenotrophomonas maltophilia (72640, 70301, 105826), Stenotrophomonas phage phiSMA7 (2, 2, 4), Streptococcus parauberis (1477, 1473, 1526), Streptococcus suis (378, 359, 582), Streptococcus thermophiles (369, 367, 389)*	94.4% 94.4% 100%	100%

Table 3.8: Precision and sensitivity for KRAKEN, CLARK, and CLARK-S on the synthetic datasets (default, unambiguous). The highest value for precision and sensitivity are indicated in bold. The second table reports the count of classified reads for KRAKEN, CLARK and CLARK-S for the negative controls.

Synthetic datasets	KRAKEN		CLARK		CLARK-S	
Default	*Precision*	*Sensitivity*	*Precision*	*Sensitivity*	*Precision*	*Sensitivity*
Buc12	93.43%	69.42%	**93.61%**	69.05%	90.36%	**71.38%**
CParMed48	99.08%	92.31%	**99.09%**	92.18%	99.08%	**93.15%**
Gut20	99.21%	82.45%	**99.24%**	82.23%	98.19%	**86.06%**
Hou31	94.25%	83.46%	**94.30%**	83.30%	93.94%	**84.32%**
Hou21	98.66%	87.00%	**98.72%**	86.81%	98.51%	**88.30%**
Soi50	99.49%	92.48%	**99.51%**	92.37%	99.32%	**93.51%**
simBA-525	91.17%	57.57%	**91.27%**	57.19%	87.50%	**58.53%**
Unambiguous						
Buc12	95.02%	73.18%	**95.26%**	72.82%	92.67%	**75.61%**
CParMed48	99.50%	94.07%	99.51%	93.91%	**99.64%**	**95.18%**
Gut20	98.87%	84.82%	**98.92%**	84.60%	98.68%	**86.06%**
Hou31	97.26%	87.57%	**97.36%**	87.45%	97.09%	**88.21%**
Hou21	99.16%	87.12%	99.19%	86.88%	**99.27%**	**89.23%**
Soi50	99.49%	92.96%	**99.51%**	92.86%	99.44%	**93.66%**
simBA-525	98.57%	88.75%	**98.69%**	88.63%	98.43%	**89.20%**

Negative control	KRAKEN	CLARK	CLARK-S
MH1	0	0	0
MH2	0	0	0
LM	0	0	0

Table 3.9: Classification speed of KRAKEN, CLARK and CLARK-S on the synthetic datasets (default and unambiguous), the negative control samples and the real samples. CLARK and KRAKEN were run with default settings (i.e., 31-mers), and, for KRAKEN, the database was loaded with the option "–preload" to assure the highest speed. Each tool was run three times to smooth I/O and cache issues (the reported numbers are the best values). The values are in thousands of read per minute. Values in bold are the highest for each dataset.

Default	KRAKEN (1 CPU)	CLARK (1 CPU)	CLARK-S (1 CPU)	CLARK-S (8 CPUs)
Buc12	2,206.0	**4,839.5**	214.4	1,220.8
CParMed48	2,060.9	**3,691.4**	204.3	913.6
Gut20	1,792.6	**3,369.5**	196.1	1,077.8
Hou31	2,111.6	**3,465.5**	201.4	1,067.7
Hou21	2,011.5	**3,308.9**	199.2	1,124.6
Soi50	2,008.6	**3,193.3**	169.5	1,074.7
simBA-525	1,955.7	**3,194.5**	203.1	1,092.5
Unambiguous				
Buc12	2,307.8	**4,160.5**	217.7	1,101.5
CParMed48	2,299.3	**4,057.7**	201.3	874.1
Gut20	2,028.0	**2,954.0**	134.3	1,083.7
Hou31	2,109.3	**3,912.9**	142.0	964.0
Hou21	2,057.8	**3,801.1**	157.8	1,003.8
Soi50	2,131.6	**2,868.9**	141.4	1,024.7
simBA-525	1,936.1	**3,359.0**	141.7	1,076.3

Negative control	KRAKEN (1 CPU)	CLARK (1 CPU)	CLARK-S (1 CPU)	CLARK-S (8 CPUs)
HM1	1,924.7	**2,619.1**	146.2	1,033.1
HM2	1,901.6	**2,932.1**	131.9	937.9
LM	2,145.8	**2,654.2**	134.2	957.3

Sample ID	KRAKEN (1 CPU)	CLARK (1 CPU)	CLARK-S (1 CPU)	CLARK-S (8 CPUs)
GC01	2,572.8	**3,142.3**	290.7	1,315.9
P00090	2,543.3	**2,587.7**	230.7	1,355.7
P00302	2,310.9	**3,330.3**	326.7	1,432.1
P00306	2,596.5	**3,553.6**	332.5	1,428.1
P00454	2,709.9	**3,668.7**	364.7	1,569.5
P00589	2,805.0	**4,929.9**	312.2	1,373.8
P00720	2,457.0	**5,203.0**	312.2	1,545.8
P00945	2,683.1	**4,758.7**	324.2	1,390.9
P01041	2,311.6	**4,348.5**	313.9	1,381.2
P01136	2,643.1	**4,893.1**	315.0	1,371.2
P01270	2,390.5	**3,548.8**	341.8	1,531.8
P01324	2,660.8	**3,513.6**	320.1	1,363.9

Table 3.10: Assignment rate (i.e., ratio in percent between the number of assigned/classified reads and the total number of reads) on real samples for KRAKEN, CLARK and CLARK-S. Values in bold are the highest.

Sample ID	KRAKEN	CLARK	CLARK-S
GC01	1.74%	1.36%	**2.55%**
P00090	54.22%	49.59%	**56.16%**
P00302	29.07%	23.70%	**29.89%**
P00306	39.37%	33.82%	**40.47%**
P00454	70.02%	66.37%	**71.50%**
P00589	31.84%	29.46%	**34.24%**
P00720	59.49%	55.59%	**64.35%**
P00945	26.26%	23.21%	**35.65%**
P01041	**67.87%**	50.28%	64.35%
P01136	31.01%	26.36%	**35.65%**
P01270	**65.20%**	50.28%	64.35%
P01324	**27.65%**	23.29%	27.23%

Chapter 4

Predicting microbial profiles by spatial locality

4.1 Introduction

In this final chapter, we focus on a microbiome of major importance: the urban microbiome. Indeed, more than half of the human population (54%) live in cities, and by 2050 66% of the human population will live in urban areas [U. N. Report, 2014]. New York City (NYC) stands as a striking example of a city with high very density and human-environment interactions. Its population is more than 8.2 M, and its subway system is one of the busiest in the world (1.7 billion riders per year commuting through 466 stations spread over 252 miles) [APTA Ridership Report, 2014]. This large urban system is an ideal framework to study disease transmissions (e.g., by disease outbreaks or bioterrorism acts). Since contaminated surfaces in public transport systems can propagate diseases [Otter and French, 2009], a constant/continuous biosurveillance of the NYC subway

system would insure the safety of riders and allow adaptive/fast actions in case of outbreaks.

The culture-independent sequencing and analysis of all DNA recovered from a sample embodied in the metagenomics discipline is revolutionizing the analysis of environmental samples. Unlike traditional procedures, which require performing multiple targeted assays each looking for a specific pathogen or organism, laboratories can use a single sequencing based test that is able to identify all microorganisms in a sample without the need for culture [Handelsman, 2004]. Fast sequencers are able to run in few hours and have a relatively low cost [Quick et al., 2015]. Thus, with the introduction of fast and mobile sequencing instruments (e.g., MinION by Oxford Nanopore Technologies) as well as fast and accurate metagenome analysis tools (e.g., CLARK [Ounit et al., 2015]), we can envision a real-time city-scale biosurveillance: at each of the N sites of interest (e.g., bus/subway stations) in a dense city (e.g., NYC), technicians collect environmental samples, perform sequencing (in laboratory or directly on site with a mobile sequencer) and send the results to a secure database accessed by public health authorities. Authorities can monitor the microbial composition across the city, track abnormal profile changes, alert targeted populations in case of outbreaks.

Such a "constant city-scale bio-surveillance" would have tremendous benefits for the health of individuals, but it has several challenges. First, at the time of writing, the cost is prohibitive. Collecting and sequencing a sample cost about $150. For a weekly bio-surveillance at a city-scale (e.g., NYC) at least a thousand of samples are needed [Afshinnekoo et al., 2015], which brings the total cost to more than $7.8 million per year. Second, it is very likely that some of the data will be missing and contaminated. Samples may be misplaced, lost, contaminated, or wrongly annotated. Consequently, the monitoring can be incorrect or incomplete. Even if a robot was col-

lecting samples and carrying out the sequencing, there is a non-zero probability of a robot to break down or being damaged.

Here we address the problem of the missing/contaminated data at a site of interest s and propose an efficient computational solution to recover the presence of microbes at s. To the best of our knowledge, there is no solution for this problem in the public literature. First, in the context of NYC subway system we demonstrate that there exist a strong correlation or high similarity of the microbial population between subway stations close of each others in data collected in [Afshinnekoo et al., 2015]. Second, we propose a Bayesian method to determine accurately the microbes present in a subway station given the microbial composition of subways stations surrounding it.

Such a model can alleviate to the missing information due to data contamination/loss, and help to minimize the cost of bio-surveillance by metagenomics at a city-scale with limited loss of detection. Finally, it can also be a solution to analyze quickly and at a high-level a city-scale microbiome without the need to process all samples at once but rather thanks to a fraction of it.

4.2 Statistical Method

4.2.1 Data collection

The dataset in [Afshinnekoo et al., 2015] contains 1,457 sequenced samples, representing a total of 4.882 B reads (a cumulative length of 1.367 10^{12} bp, or 2.988 TB in disk space). The average length of the reads is 280 bp. Samples were collected at all open subway stations and all subway lines of the NYC subway system, but also the Staten Island Railway, the Gowanus canal and public parks. Areas swabbed include various objects (e.g., garbage can, bench, water, turnstile, kiosk, stairway rail or water) [Afshinnekoo et al., 2015]. Each sample was annotated by several

attributes (e.g., sample ID, station name, GPS coordinates of the swabbed object, etc.)

4.2.2 Taxonomic classification

In [Afshinnekoo et al., 2015], the authors used MegaBLAST [Zhang et al., 2000], BWA [Li and Durbin, 2009], MEGAN and MetaPhlAn2 in order to classify sequences against a database of reference sequences. Here we used CLARK-S (v1.2.3) because it can classify more reads and with higher accuracy than other state-of-the-art tools as described previously. We compared the taxonomic classification of CLARK-S against the findings from the original study [Afshinnekoo et al., 2015] and found that the CLARK-S's results are consistent with them. The top bacterial species (and viruses) detected by CLARK-S match the list of top species detected in [Afshinnekoo et al., 2015].

4.2.3 Post-processing of CLARK-S results

Classification results were post-processed in order to filter low-confidence assignment. CLARK-S provides statistics for each classified read, namely a confidence score and a gamma score (see previously chapter). A low confidence score means that the read may be mapped to several species (ambiguous read) and a low gamma score indicates that the read was classified with low evidence.

A cut-off of 0.75 on the confidence score is sufficient to select high confidence assignments [Ounit et al., 2015]. However, it is critical to set a high cut-off for the gamma score to insure that reads were assigned with sufficient evidence. Based on distribution of the gamma score, we decided to filter out reads that had a confidence score lower than 0.75 or with a gamma score lower than 0.50. We obtain 1.363 B classified reads (27.9% of the total).

4.2.4 Taxonomic analysis of the samples

We have analyzed the microbial composition at the sample level and subway station level. Using the metadata of each sample (indicating the sample ID-subway station associations) provided by [Afshinnekoo et al., 2015], the microbial composition was identified for each sample. Then, for each subway station swabbed with T samples, we computed for each species i its *abundance ratio*, which is the ratio between the number of reads classified as species i in all T samples and the total number of classified reads in all T samples.

Table 4.1 and 4.2 report the top dominant bacteria and viruses identified by CLARK-S through all the samples, for different minimum abundance ratio thresholds. In Table 4.1, the top three dominant bacteria *Pseudomonas stutzeri*, *Stenotrophomonas maltophilia* and *Enterobacter cloacae* are in this order also the top three dominant bacteria reported in [Afshinnekoo et al., 2015]. In addition, the presence of the viruses *Enterobacteria phage phiX174*, *Enterobacteria phage mEp235* and *Erwinia phage ENT90* is consistent with the top ten viruses/phages reported in the [Afshinnekoo et al., 2015]. Observe also the concomitant presence of bacteriophages with their bacterial hosts, which suggests a consistency in the microbial profile as also observed in [Afshinnekoo et al., 2015]. However, several species detected in [Afshinnekoo et al., 2015] as dominant are missed by CLARK-S (e.g., *Acinetobacter radioresistans*, *Acinetobacter nosocomialis* or *Lysinibacillus sphaericus*). Similarly, several viruses were not detected by CLARK-S. In both cases, the reason is due to the fact the reference sequence of these organisms was not included in the CLARK-S database. We can make similar observations in Table 4.2 although fewer viruses were detected because of a higher abundance threshold. These results show that CLARK-S was

Table 4.1: Summary of the ten most dominant bacterial species (Top) and viral species (Bottom) detected by CLARK-S with an abundance ratio higher than 0.1%. For each species, we reported the related rank, the number of samples in which the species is detected, the species names, and the corresponding NCBI taxonomy ID.

Rank	Number of samples	Bacteria name (species)	NCBI Taxonomy ID
1	991	*Pseudomonas stutzeri*	316
2	785	*Stenotrophomonas maltophilia*	40324
3	665	*Enterobacter cloacae*	550
4	630	*Acinetobacter baumannii*	470
5	517	*Pseudomonas aeruginosa*	287
6	511	*Pseudomonas putida*	303
7	483	*Escherichia coli*	562
8	473	*Enterococcus casseliflavus*	37734
9	447	*Salmonella enterica*	28901
10	433	*Klebsiella pneumoniae*	573

Rank	Number of samples	Virus name (species)	NCBI Taxonomy ID
79	45	*Enterobacteria phage phiX174 sensu lato*	374840
96	33	*Escherichia phage HK639*	906669
151	17	*Enterobacteria phage mEp235*	1147150
169	13	*Salmonella phage SSU5*	1177632
189	11	*Enterococcus phage EF62phi*	977801
230	7	*Erwinia phage ENT90*	947843
242	6	*Enterobacterial phage mEp390*	1147158
267	5	*Staphylococcus phage phiRS7*	1403390
341	3	*Human endogenous retrovirus K*	45617
346	3	*Stenotrophomonas phage S1*	573591

able to retrieve all dominant microbes found in [Afshinnekoo et al., 2015].

4.2.5 Bayesian inference model

In this section, we first introduce some notations and then present the probabilistic model. Each subway station has a GPS coordinate. In order to define the distance between two subway stations, we assume that stations of the NYC subway system are on a 2D plane and that the latitude and longitude coordinates can be viewed as the standard (x, y) coordinates. Since the NYC subway

Table 4.2: Summary of the ten most dominant bacterial species (Top) and viral species (Bottom) detected by CLARK-S with an abundance ratio higher than 1%. For each species, we reported the related rank, the number of samples in which the species is detected, the species names, and the corresponding NCBI taxonomy ID.

Rank	Number of samples	Virus name (species)	NCBI Taxonomy ID
1	809	*Pseudomonas stutzeri*	316
2	574	*Stenotrophomonas maltophilia*	40324
3	430	*Enterobacter cloacae*	550
4	310	*Acinetobacter baumannii*	470
5	231	*Enterococcus casseliflavus*	37734
6	160	*Klebsiella pneumoniae*	573
7	157	*Pseudomonas putida*	303
8	133	*Enterococcus faecalis*	1351
9	127	*Escherichia coli*	562
10	119	*Exiguobacterium sp. MH3*	1399115

Rank	Number of samples	Virus name (species)	NCBI Taxonomy ID
50	17	*Enterobacteria phage phiX174 sensu lato*	374840
168	1	*Enterococcus phage EF62phi*	977801
180	1	*Escherichia phage phAPEC8*	1229753
184	1	*Erwinia phage ENT90*	947843
186	1	*Salmonella phage SSU5*	1177632
192	1	*Escherichia phage HK639*	906669

system spans an area of few miles in both direction (i.e., North-South and East-West), the distance between two stations is approximatively equivalent to the Euclidean distance.

We use ρ to denote a distance radius. Let s be a subway station and let $N(s, \rho)$ be the set of subway stations located within ρ distance from s. In other words, $N(s, \rho)$ is the neighborhood of s in a radius ρ. The number of elements in the neighborhood is $|N(s, \rho)|$. We define the "abundance ratio" of a species i for a given station s (that was swabbed and from which M reads were sequenced), as the number of reads assigned to i by CLARK-S divided by the total number of reads assigned by CLARK-S. A species i is present in the microbiome of station s (and we write $i \in s$) if the abundance ratio of i in s is higher than some predefined threshold (e.g., 1% or 0.1%).

Similarly, we can define the presence of a species in the neighborhood of a station. We say that the species i is present in the neighborhood of s (and we write $i \in \gamma N(s, \rho)$) if and only if i is present in at least $\gamma \times |N(s, \rho)|$ stations of $N(s, \rho)$, where γ is a value between 0 and 1.

Bayesian model

For a given station s, a distance ρ, a species i and $\gamma \in [0, 1]$, let us formulate probabilities $P(i \in s | i \in \gamma N(s, \rho))$ and $P(i \in s | i \notin \gamma N(s, \rho))$ using the Bayesian theorem:

$$P(i \in s | i \in \gamma N(s, \rho)) = \frac{P(i \in \gamma N(s, \rho) | i \in s) \times P(i \in s)}{P(i \in \gamma N(s, \rho) | i \in s) \times P(i \in s) + P(i \in \gamma N(s, \rho) | i \notin s) \times P(i \notin s)} \quad (4.1)$$

$$P(i \in s | i \notin \gamma N(s, \rho)) = \frac{P(i \notin \gamma N(s, \rho) | i \in s) \times P(i \in s)}{P(i \notin \gamma N(s, \rho) | i \in s) \times P(i \in s) + P(i \notin \gamma N(s, \rho) | i \notin s) \times P(i \notin s)} \quad (4.2)$$

Because probabilities $P(i \in s)$ and $P(i \in \gamma N(s, \rho) | i \in s)$ (as well as $P(i \notin s)$ and $P(i \in \gamma N(s, \rho) | i \notin s)$) can be easily precomputed from a history of observations, $P(i \in s | i \in \gamma N(s, \rho))$ can be quickly estimated. Similarly, we can estimate $P(i \in s | i \notin \gamma N(s, \rho))$.

To infer the microbial composition of the station s, we use thresholds $0 \leq \theta_1 \leq 1$ and $0 \leq \theta_2 \leq 1$ for $P(i \in s | i \in \gamma N(s, \rho))$ and $P(i \in s | i \notin \gamma N(s, \rho))$, respectively. If $P(i \in s | i \in \gamma N(s, \rho)) > \theta_1$ then we consider species i present in s. Similarly, if $P(i \in s | i \notin \gamma N(s, \rho)) > \theta_2$ then we consider species i present in s.

4.2.6 Training and testing

Data cleaning

We removed samples that were partially annotated (e.g., missing GPS coordinates, or unknown associations between sample ID and stations), as well as any non-subway stations (e.g., no canal and parks). The final dataset contained 1,064 samples covering 412 subway stations.

Cross-fold validation

In order to evaluate the performance of the model, we applied a five-fold cross validation. We divided the set of stations into five subsets. We used four subsets for inferring the probabilities $P(i \in s)$, $P(i \notin s)$, $P(i \in \gamma N(s,\rho)|i \in s)$, $P(i \in \gamma N(s,\rho)|i \notin s)$ and $P(i \notin \gamma N(s,\rho)|i \in s)$ and $P(i \notin \gamma N(s,\rho)|i \notin s)$ and we tested the model on the remaining fifth subset. We repeated this procedure four more times, each time using a different subset for testing. During these five evaluations, we computed True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN). Based on TP, TN, FP and FN, we computed precision $\frac{TP}{TP+FP}$, recall $\frac{TP}{TP+FN}$ and accuracy $\frac{TP+TN}{TP+TN+FP+FN}$. While the accuracy provides an overall performance of the predictor, the precision indicates how often predictions are correct and the recall measures the fraction of species correctly predicted. We also tried 10-fold cross validation but we did not observe a significant change in precision/recall/accuracy. We also trained and tested the model on several thresholds of abundance ratio (see Table 4.3).

4.3 Experimental Results

In order to analyze the spatial locality of the microbiome of the NYC subway system, we analyzed all pairs of stations (i.e., 84,666 pairs from 412 stations) and for each pair, we compared the microbial composition computed by CLARK-S between the two stations. To compare two microbial composition, we computed the Pearson correlation coefficient on the log value of the abundance ratios. High correlation values (≥ 0.9) but also negative/null correlation values (≤ -0.2) were found in the analysis of these pairs.

In Figure 4.1, we show the dependencies between the pairwise distance and the corresponding correlation intervals. For each correlation interval we plotted the first quartile, average and third quartile of all associated pairwise distance. The figure clearly shows that stations closer to each other have higher correlation of the microbial composition than stations that are far away from each other.

4.3.1 Evaluation of the model

The performance of our model depends on parameters ρ, γ, θ_1, θ_2. In order to optimize the performance, we conducted a grid search of the parameter space. We considered $\rho = [0.005, 0.02]$ with step 0.001, $\gamma = [0, 1]$ with step 0.05, $\theta_1 = [0, 1]$ with step 0.05, $\theta_2 = [0, 1]$ with step 0.05.

For each choice of ρ, γ, θ_1, θ_2 we carried out the five-fold cross validation, computed precision, recall and accuracy and selected the parameters values producing the highest performance. We decided to maximize the sum of the three metrics (i.e., precision+recall+accuracy), with the constraint that each metric had to be at least higher than 0.50 to avoid trivial solutions, e.g., precision=1.0, accuracy=1.0, and recall=0.0.

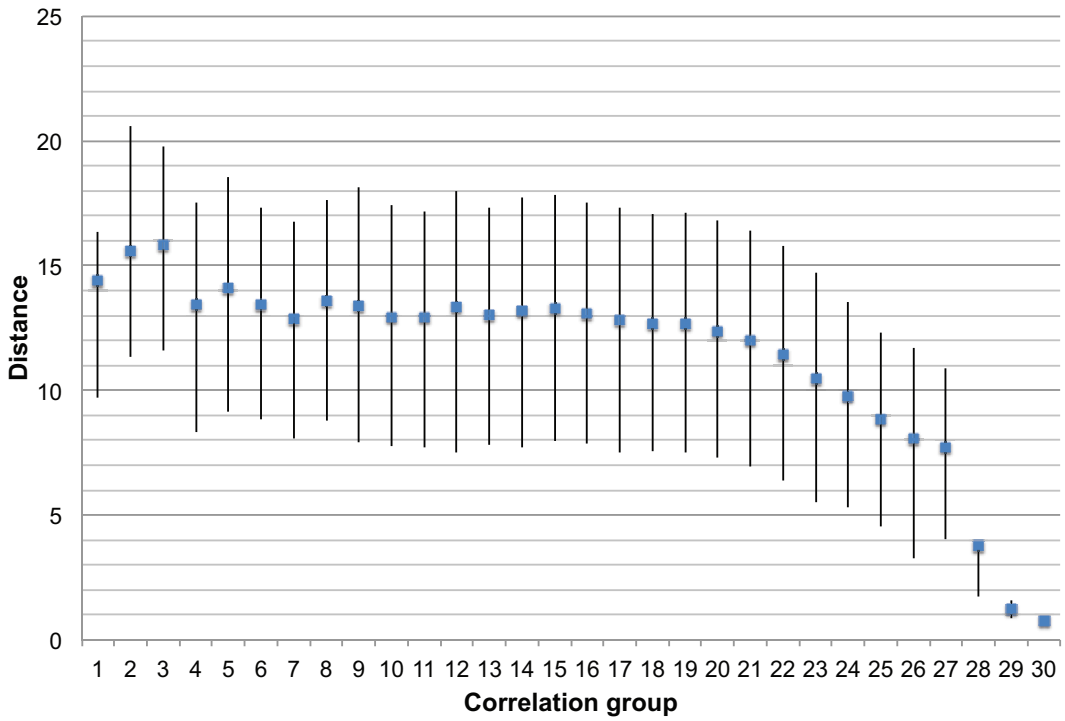

Figure 4.1: Dependencies between the pairwise distance of subway stations and the Pearson correlation coefficient of the microbial composition of corresponding pair of stations. The x-axis represents intervals of Pearson correlation values. The y-axis represents the pairwise distance between stations. For each correlation group, the first quartile, average and third quartile of all related pairwise distance are plotted.

In Table 4.3, we present the best performance over the parameter space for different threshold value for the abundance ratio. Observe that for a threshold of 1%, the model can achieve high accuracy and high precision.

4.3.2 Perspective for biosurveillance

In the context of a city-scale biosurveillance, any missing data due to samples being contaminated or lost can prevent a complete analysis and overview of the dynamics of targeted microbiomes. A predictive model such the one we presented can help to dealing with missing information.

A strategy to reduce the high cost of biosurveillance would be to sample, sequence and analyze a carefully chosen subset of stations based on their proximity and then use a predictive model (trained on previous observations) to determine the microbial composition of the missing stations.

4.3.3 Running time and database

As explained in the method section, CLARK-S was run for the taxonomic classification of samples. We used the default database of bacterial, archaeal and viruses genomes from NCBI/RefSeq (a total of 5,747 species). The 4.882 billion raw reads were classified, at the species rank, with 8 CPU, in only 6.2 days on a Dell PowerEdge T710 server (dual Intel Xeon X5660 2.8 GHz, 12 cores and 192 GB of RAM). The RAM usage for the classification was about 110 GB.

Once the microbial composition is obtained, our probabilistic model can be trained in few minutes, as it only requires to compute distances and perform look-ups of organisms. The computational complexity of our model is $O(N^2)$ where N is the number of stations/samples.

Table 4.3: Performance of the Bayesian model for several threshold of abundance ratio. For each threshold of the abundance ratio, a five-fold cross-validation was performed to train and test the model, and estimate the precision, recall and accuracy. For each threshold, the values of the model parameters ρ, μ, γ, θ_1 and θ_2 that allow the highest classification performance are reported.

Abundance threshold (%)	ρ	μ	γ	θ_1	θ_2	Precision (%)	Recall (%)	Accuracy (%)
1.0	0.013	5	0.90	0.20	0.45	77.7	53.6	85.8
0.9	0.013	5	0.90	0.20	0.45	78.5	51.6	85.5
0.8	0.013	5	0.55	0.35	0.45	74.9	55.1	85.7
0.7	0.014	6	0.50	0.35	0.45	73.4	55.5	84.7
0.6	0.013	5	0.55	0.30	0.5	75.5	50.3	86.4
0.5	0.015	6	0.60	0.15	0.35	68.9	55.4	86.6
0.4	0.015	6	0.55	0.05	0.35	69.6	50.2	87.1
0.3	0.016	6	0.50	0.25	0.25	51.7	63.9	82.7
0.2	0.012	7	0.65	0.30	0.3	51.5	73.0	73.8
0.1	0.012	5	0.70	0.30	0.4	64.1	50.1	82.7

4.4 Conclusion

We have presented a probabilistic model capable of predicting microorganisms present in a given subway station using the known microbial composition of the surrounding stations. Using real data from the NYC subway system, our model was able to achieve high accuracy (83% and 86% for an abundance threshold of 0.1% and 1% respectively), and high precision (64% and 78% for an abundance threshold of 0.1% and 1% respectively). However, the relatively low recall (50% and 54% for an abundance threshold of 0.1% and 1% respectively) is a weakness of the approach.

These preliminary findings suggest for the first time that it is possible to design a predictive model for the microbial composition of geographically-related samples. While our model was trained on static data, we expect that the performance would improve by including time-dependent observation. The authors of [Afshinnekoo et al., 2015] analyzed the temporal evolution of the microbiome for one station (Penn Station) and observed that some organisms are detected constantly and others with fluctuation over time (likely due to variation of temperature). We speculate that this phenomenon can also be observed in several other subway stations, and hypothesize that the low recall of our model could be attributed to an out-of-synch sampling.

We understand the technical challenges of collecting metagenomic samples for the entire NYC subway at multiple time points, considering the entire NYC microbiome swabbing in [Afshinnekoo et al., 2015] took about 18 months. However, our model could be easily extended to take into account temporal dependencies.

Chapter 5

Conclusions

Within the past decade, the improvement of the sequencing technologies has enabled major discoveries in molecular biology, but it also has created several computational challenges in the analysis pipeline. In metagenomics, faster, more accurate and more efficient algorithms are needed to analyze environmental or clinical samples that are sequenced more and more quickly. In genomics, fast and efficient algorithms are required to process massive amount of data from large, complex and repetitive genomes.

This dissertation presents new computational methods for solving the problem of supervised sequence classification for metagenomics and genomics applications. We have designed and implemented a family of software tools (i.e., CLARK, CLARK-l and CLARK-S) that are accurate, efficient and faster than previously published methods. Our tools are already considered state-of-the-art and currently used by several research teams world-wide. We have also described a new statistical approach for pathogen detection and biosurveillance at a city-scale, which allows one to infer/inpute the presence of microbes in order to compensate the loss/contamination of data/samples.

Recent progress in sequencing techniques allow to foresee future computational challenges. For example, quantum tunneling is expected to allow a much higher output than any existing sequencing platform [Di Ventra and Taniguchi, 2016]. If hundreds of samples could be sequenced in only few hours then much faster sample analysis methods and more efficient storage/compression solutions for these data would be needed.

Recently it has been shown that DNA sequencing in space has several advantages compared to sequencing on the ground [Castro-Wallace et al., 2016]. If sequencing was routinely performed on the International Space Station, efficient and inexpensive solutions for data transfer between ground and space would be crucial to process larger and larger samples.

Constant innovation in sequencing technologies will create continuing pressure on computational methodologies to keep the pace in the analysis of genomic and metagenomic data. Creativity and ingenuity will be needed for solving new computational problems.

Bibliography

The National Microbiome Initiative. *The White House Office Of Science and Technology*, 2016. URL https://www.whitehouse.gov/blog/2016/05/13/announcing-national-microbiome-initiative.

R. I. Adams, A. C. Bateman, H. M. Bik, and J. F. Meadow. Microbiota of the indoor environment: a meta-analysis. *Microbiome*, 3(1):1, 2015.

E. Afshinnekoo, C. Meydan, et al. Geospatial resolution of human and bacterial diversity with city-scale metagenomics. *Cell Systems*, 1(1):72–87, 2015.

H. K. Allen, L. A. Moe, J. Rodbumrer, A. Gaarder, and J. Handelsman. Functional metagenomics reveals diverse β-lactamases in a remote alaskan soil. *The ISME journal*, 3(2):243–251, 2009.

S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.

S. K. Ames, D. A. Hysom, S. N. Gardner, G. S. Lloyd, M. B. Gokhale, and J. E. Allen. Scalable metagenomic taxonomy classification using a reference genome database. *Bioinformatics*, 29(18): 2253–2260, 2013.

M. A. Antonio, S. E. Hawes, and S. L. Hillier. The identification of vaginal lactobacillus species and the demographic and microbiologic characteristics of women colonized by these species. *Journal of Infectious Diseases*, 180(6):1950–1956, 1999.

E. Bao, T. Jiang, I. Kaloshian, and T. Girke. Seed: efficient clustering of next-generation sequences. *Bioinformatics*, 27(18):2502–2509, 2011.

A. L. Bazinet and M. P. Cummings. A comparative evaluation of sequence classification programs. *BMC bioinformatics*, 13(1):92, 2012.

D. A. Benson, M. Cavanaugh, K. Clark, I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, and E. W. Sayers. Genbank. *Nucleic Acids Research*, page gks1195, 2012.

A. Brady and S. Salzberg. PhymmBL expanded: confidence scores, custom databases, parallelization and more. *Nature Methods*, 8(5):367–367, 2011.

D. G. Brown, M. Li, and B. Ma. A tutorial of recent developments in the seeding of local alignment. *Journal of bioinformatics and computational biology*, 2(04):819–842, 2004.

B. Buchfink, C. Xie, and D. H. Huson. Fast and sensitive protein alignment using diamond. *Nature methods*, 12(1):59–60, 2015.

S. Burkhardt and J. Kärkkäinen. Better filtering with gapped q-grams. In *Annual Symposium on Combinatorial Pattern Matching*, pages 73–85. Springer, 2001.

S. L. Castro-Wallace, C. Y. Chiu, K. K. John, S. E. Stahl, K. H. Rubins, A. B. McIntyre, J. P. Dworkin, M. L. Lupisella, D. J. Smith, D. J. Botkin, et al. Nanopore dna sequencing and genome assembly on the international space station. *bioRxiv*, page 077651, 2016.

K. P. Choi, F. Zeng, and L. Zhang. Good spaced seeds for homology search. In *Bioinformatics and Bioengineering, 2004. BIBE 2004. Proceedings. Fourth IEEE Symposium on*, pages 379–386. IEEE, 2004.

T. J. Close, S. Wanamaker, M. L. Roose, and M. Lyon. *HarvEST*. Springer, 2007.

T. J. Close, P. R. Bhat, S. Lonardi, Y. Wu, N. Rostoks, L. Ramsay, A. Druka, N. Stein, J. T. Svensson, S. Wanamaker, et al. Development and implementation of high-throughput SNP genotyping in barley. *BMC genomics*, 10(1):582, 2009.

P. E. Compeau, P. A. Pevzner, and G. Tesler. How to apply de Bruijn graphs to genome assembly. *Nature biotechnology*, 29(11):987–991, 2011.

H. M. P. Consortium et al. A framework for human microbiome research. *Nature*, 486(7402): 215–221, 2012.

M. I. Consortium et al. The metagenomics and metadesign of the subways and urban biomes (metasub) international consortium inaugural meeting report. *Microbiome*, 4(1):1–14, 2016.

M. Di Ventra and M. Taniguchi. Decoding dna, rna and peptides with quantum tunnelling. *Nature nanotechnology*, 11(2):117–126, 2016.

J. Doležel, J. Vrána, J. Šafář, J. Bartoš, M. Kubaláková, and H. Šimková. Chromosomes in the flow to simplify genome analysis. *Functional & integrative genomics*, 12(3):397–416, 2012.

M. Escalona, S. Rocha, and D. Posada. A comparison of tools for the simulation of genomic next-generation sequencing data. *Nature Reviews Genetics*, 17(8):459–469, 2016.

A. Felczykowska, S. K. Bloch, B. Nejman-Falenczyk, and S. Baranska. Metagenomic approach in the investigation of new bioactive compounds in the marine environment. *Acta Biochim Pol*, 59: 501–505, 2012.

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. *Proceedings of the National Academy of Sciences*, 109(52):21390–21395, 2012.

E. A. Franzosa, K. Huang, J. F. Meadow, D. Gevers, K. P. Lemon, B. J. Bohannan, and C. Huttenhower. Identifying personal microbiomes using metagenomic codes. *Proceedings of the National Academy of Sciences*, 112(22):E2930–E2938, 2015.

V. Galata, C. Backes, C. C. Laczny, G. Hemmrich-Stanisak, H. Li, L. Smoot, A. E. Posch, S. Schmolke, M. Bischoff, L. von Müller, et al. Comparing genome versus proteome-based identification of clinical bacterial isolates. *Briefings in Bioinformatics*, page bbw122, 2016.

J. Gardy, N. J. Loman, and A. Rambaut. Real-time digital pathogen surveillancethe time is now. *Genome Biology*, 16(1), 2015.

S. Goodwin, J. D. McPherson, and W. R. McCombie. Coming of age: ten years of next-generation sequencing technologies. *Nature Reviews Genetics*, 17(6):333–351, 2016.

L. Hahn, C.-A. Leimeister, R. Ounit, S. Lonardi, and B. Morgenstern. rasbhari: Optimizing spaced seeds for database searching, read mapping and alignment-free sequence comparison. *PLOS Computational Biology*, 12(10):e1005107, oct 2016. doi: 10.1371/journal.pcbi.1005107. URL `http://dx.doi.org/10.1371/journal.pcbi.1005107`.

J. Handelsman. Metagenomics: application of genomics to uncultured microorganisms. *Microbiology and molecular biology reviews*, 68(4):669–685, 2004.

E. Y. Harris, R. Ounit, and S. Lonardi. Brat-nova: Fast and accurate mapping of bi-sulfite-treated reads. *Bioinformatics*, page btw226, 2016.

T. Hsu, R. Joice, J. Vallarino, G. Abu-Ali, E. M. Hartmann, A. Shafquat, C. DuLong, C. Baranowski, D. Gevers, J. L. Green, et al. Urban transit system microbial communities differ by surface type and interaction with humans and the environment. *mSystems*, 1(3):e00018–16, 2016.

W. Huang, L. Li, et al. ART: a next-generation sequencing read simulator. *Bioinformatics*, 28(4): 593–594, 2012.

Human Microbiome Project Consortium . Structure, function and diversity of the healthy human microbiome. *Nature*, 486(7402):207–214, 2012.

D. H. Huson, A. F. Auch, J. Qi, and S. C. Schuster. Megan analysis of metagenomic data. *Genome research*, 17(3):377–386, 2007.

D. H. Huson, S. Mitra, H.-J. Ruscheweyh, N. Weber, and S. C. Schuster. Integrative analysis of environmental sequences using megan4. *Genome research*, 21(9):1552–1560, 2011.

R. W. Hyman, M. Fukushima, L. Diamond, J. Kumm, L. C. Giudice, and R. W. Davis. Microbes on the human vaginal epithelium. *Proceedings of the National Academy of Sciences*, 102(22): 7952–7957, 2005.

L. Ilie and S. Ilie. Multiple spaced seeds for homology search. *Bioinformatics*, 23(22):2969–2977, 2007.

L. Ilie, S. Ilie, and A. M. Bigvand. Speed: fast computation of sensitive spaced seeds. *Bioinformatics*, 27(17):2433–2434, 2011.

B. Jia, A. R. Raphenya, B. Alcock, N. Waglechner, P. Guo, K. K. Tsang, B. A. Lago, B. M. Dave, S. Pereira, A. N. Sharma, et al. Card 2017: expansion and model-centric curation of the comprehensive antibiotic resistance database. *Nucleic acids research*, 45(D1):D566–D573, 2017.

K. E. Jones, N. G. Patel, M. A. Levy, A. Storeygard, D. Balk, J. L. Gittleman, and P. Daszak. Global trends in emerging infectious diseases. *Nature*, 451(7181):990–993, 2008.

W. J. Kent. Blatthe blast-like alignment tool. *Genome research*, 12(4):656–664, 2002.

V. Kuleshov, C. Jiang, W. Zhou, F. Jahanbani, S. Batzoglou, and M. Snyder. Synthetic long-read sequencing reveals intraspecies diversity in the human microbiome. *Nature biotechnology*, 34(1): 64–69, 2016.

C.-A. Leimeister, M. Boden, S. Horwege, S. Lindner, and B. Morgenstern. Fast alignment-free sequence comparison using spaced-word frequencies. *Bioinformatics*, page btu177, 2014.

S. E. Levy and R. M. Myers. Advancements in next-generation sequencing. *Annual review of genomics and human genetics*, 17:95–115, 2016.

R. E. Ley, D. A. Peterson, and J. I. Gordon. Ecological and evolutionary forces shaping microbial diversity in the human intestine. *Cell*, 124(4):837–848, 2006.

H. Li and R. Durbin. Fast and accurate short read alignment with burrows–wheeler transform. *Bioinformatics*, 25(14):1754–1760, 2009.

M. Li, B. Ma, D. Kisman, and J. Tromp. Patternhunter ii: Highly sensitive and fast homology search. *Journal of bioinformatics and computational biology*, 2(03):417–439, 2004.

M. Li, B. Ma, and L. Zhang. Superiority and complexity of the spaced seeds. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pages 444–453. Society for Industrial and Applied Mathematics, 2006.

S. Lindgreen, K. L. Adair, and P. P. Gardner. An evaluation of the accuracy and speed of metagenome analysis tools. *Scientific reports*, 6, 2016.

B. Liu, T. Gibbons, M. Ghodsi, T. Treangen, and M. Pop. Accurate and fast estimation of taxonomic profiles from metagenomic shotgun sequences. *BMC genomics*, 12(Suppl 2):S4, 2011.

S. Lonardi, D. Duma, M. Alpert, F. Cordero, M. Beccuti, P. R. Bhat, Y. Wu, G. Ciardo, B. Alsaihati, Y. Ma, et al. Combinatorial pooling enables selective sequencing of the barley gene space. *PLoS computational biology*, 9(4):e1003010, 2013.

R. Luo, B. Liu, Y. Xie, Z. Li, W. Huang, J. Yuan, G. He, Y. Chen, Q. Pan, Y. Liu, et al. SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler. *Gigascience*, 1 (1):18, 2012.

B. Ma, J. Tromp, and M. Li. Patternhunter: faster and more sensitive homology search. *Bioinformatics*, 18(3):440–445, 2002.

T. Magoc, S. Pabinger, S. Canzar, X. Liu, Q. Su, D. Puiu, L. J. Tallon, and S. L. Salzberg. GAGE-B: an evaluation of genome assemblers for bacterial organisms. *Bioinformatics*, 29(14):1718–1725, 2013.

G. Marçais and C. Kingsford. A fast, lock-free approach for efficient parallel counting of occurrences of k-mers. *Bioinformatics*, 27(6):764–770, 2011.

J. L. Martínez. Antibiotics and antibiotic resistance genes in natural environments. *Science*, 321 (5887):365–367, 2008.

A. C. Martiny, J. B. Martiny, C. Weihe, A. Field, and J. C. Ellis. Functional metagenomics reveals previously unrecognized diversity of antibiotic resistance genes in gulls. *Frontiers in microbiology*, 2, 2011.

K. Mavromatis, N. Ivanova, K. Barry, H. Shapiro, E. Goltsman, A. C. McHardy, I. Rigoutsos, A. Salamov, F. Korzeniewski, M. Land, et al. Use of simulated data sets to evaluate the fidelity of metagenomic processing methods. *Nature methods*, 4(6):495–500, 2007.

A. G. McArthur and G. D. Wright. Bioinformatics of antimicrobial resistance in the age of molecular epidemiology. *Current opinion in microbiology*, 27:45–50, 2015.

A. G. McArthur, N. Waglechner, F. Nizam, A. Yan, M. A. Azad, A. J. Baylay, K. Bhullar, M. J. Canova, G. De Pascale, L. Ejim, et al. The comprehensive antibiotic resistance database. *Antimicrobial agents and chemotherapy*, 57(7):3348–3357, 2013.

D. R. Mende, S. Sunagawa, G. Zeller, and P. Bork. Accurate and universal delineation of prokaryotic species. *Nature methods*, 10(9):881–884, 2013.

R. R. Miller, V. Montoya, J. L. Gardy, D. M. Patrick, and P. Tang. Metagenomics for pathogen detection in public health. *Genome Med*, 5(9):81, 2013.

B. Morgenstern, B. Zhu, S. Horwege, and C. A. Leimeister. Estimating evolutionary distances between genomic sequences from spaced-word matches. *Algorithms for Molecular Biology*, 10 (1):5, 2015.

R. S. Mueller, S. Bryson, B. Kieft, Z. Li, J. Pett-Ridge, F. Chavez, R. L. Hettich, C. Pan, and X. Mayali. Metagenome sequencing of a coastal marine microbial community from monterey bay, california. *Genome announcements*, 3(2):e00341–15, 2015.

M. Muñoz-Amatriaín, S. Lonardi, M. Luo, K. Madishetty, J. T. Svensson, M. J. Moscou, S. Wanamaker, T. Jiang, A. Kleinhofs, G. J. Muehlbauer, et al. Sequencing of 15 622 gene-bearing bacs clarifies the gene-dense regions of the barley genome. *The Plant Journal*, 84(1):216–227, 2015.

A. B. Nicholas, J. B. Thissen, V. Y. Fofanov, J. E. Allen, M. Rojas, G. Golovko, Y. Fofanov, H. Koshinsky, and C. J. Jaing. Metagenomic analysis of the airborne environment in urban spaces. *Microbial ecology*, 69(2):346–355, 2015.

J. Otter and G. French. Bacterial contamination on touch surfaces in the public transport system and in public areas of a hospital in london. *Letters in applied microbiology*, 49(6):803–805, 2009.

R. Ounit and S. Lonardi. Higher classification accuracy of short metagenomic reads by discriminative spaced k-mers. In *Algorithms in Bioinformatics*, pages 286–295. Springer, 2015.

R. Ounit and S. Lonardi. Higher classification sensitivity of short metagenomic reads with CLARK-S. *Bioinformatics*, 32(24):3823–3825, aug 2016. doi: 10.1093/bioinformatics/btw542. URL `http://dx.doi.org/10.1093/bioinformatics/btw542`.

R. Ounit, S. Wanamaker, T. J. Close, and S. Lonardi. Clark: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers. *BMC genomics*, 16(1):1, 2015.

N. R. Pace. Mapping the tree of life: progress and prospects. *Microbiology and Molecular Biology Reviews*, 73(4):565–576, 2009.

K. R. Patil, P. Haider, P. B. Pope, P. J. Turnbaugh, M. Morrison, T. Scheffer, and A. C. McHardy. Taxonomic metagenome sequence assignment with structured output models. *Nature methods*, 8 (3):191–192, 2011.

J. Qin, R. Li, J. Raes, M. Arumugam, K. S. Burgdorf, C. Manichanh, T. Nielsen, N. Pons, F. Levenez, T. Yamada, et al. A human gut microbial gene catalogue established by metagenomic sequencing. *nature*, 464(7285):59–65, 2010.

J. Quick, P. Ashton, S. Calus, C. Chatt, S. Gossain, J. Hawker, S. Nair, K. Neal, K. Nye, T. Peters, et al. Rapid draft sequencing and real-time nanopore sequencing in a hospital outbreak of salmonella. *Genome Biol*, 16(114.2015):10–1186, 2015.

J. Quick, N. J. Loman, S. Duraffour, J. T. Simpson, E. Severi, L. Cowley, J. A. Bore, R. Koundouno, G. Dudas, A. Mikhail, et al. Real-time, portable genome sequencing for ebola surveillance. *Nature*, 530(7589):228–232, 2016.

A. T. Reese, A. Savage, E. Youngsteadt, K. L. McGuire, A. Koling, O. Watkins, S. D. Frank, and R. R. Dunn. Urban stress is associated with variation in microbial species compositionbut not richnessin manhattan. *The ISME journal*, 2015.

G. L. Rosen, E. R. Reichenberger, and A. M. Rosenfeld. NBC: the naive bayes classification tool webserver for taxonomic classification of metagenomic reads. *Bioinformatics*, 27(1):127–129, 2011.

J. F. Ruiz-Calderon, H. Cavallin, S. J. Song, A. Novoselac, L. R. Pericchi, J. N. Hernandez, R. Rios, O. H. Branch, H. Pereira, L. C. Paulino, et al. Walls talk: Microbial biogeography of homes spanning urbanization. *Science advances*, 2(2):e1501061, 2016.

H. S. Said, W. Suda, S. Nakagome, H. Chinen, K. Oshima, S. Kim, R. Kimura, A. Iraha, H. Ishida, J. Fujita, et al. Dysbiosis of salivary microbiota in inflammatory bowel disease and its association with oral immunological biomarkers. *DNA research*, page dst037, 2013.

D. Schulte, R. Ariyadasa, B. Shi, D. Fleury, C. Saski, M. Atkins, C.-C. Wu, A. Graner, P. Langridge, N. Stein, et al. BAC library resources for map-based cloning and physical map construction in barley (Hordeum vulgare L.). *Bmc Genomics*, 12(1):247, 2011.

N. Segata, L. Waldron, A. Ballarini, V. Narasimhan, O. Jousson, and C. Huttenhower. Metagenomic microbial community profiling using unique clade-specific marker genes. *Nature methods*, 9(8):811–814, 2012.

H. M. Seth-Smith, S. R. Harris, R. J. Skilton, F. M. Radebe, D. Golparian, E. Shipitsyna, P. T. Duy, P. Scott, L. T. Cutcliffe, C. ONeill, et al. Whole-genome sequences of chlamydia trachomatis directly from clinical samples without culture. *Genome research*, 23(5):855–866, 2013.

E. Stackebrandt and B. Goebel. Taxonomic note: a place for DNA-DNA reassociation and 16s rRNA sequence analysis in the present species definition in bacteriology. *Journal of Systematic and Evolutionary Microbiology*, 44(4):846–849, 1994.

S. Sunagawa, D. R. Mende, G. Zeller, F. Izquierdo-Carrasco, S. A. Berger, J. R. Kultima, L. P. Coelho, M. Arumugam, J. Tap, H. B. Nielsen, et al. Metagenomic species profiling using universal phylogenetic marker genes. *Nature Methods*, 10(12):1196–1199, 2013.

L. R. Thompson, G. J. Williams, M. F. Haroon, A. Shibl, P. Larsen, J. Shorenstein, R. Knight, and U. Stingl. Metagenomic covariation along densely sampled environmental gradients in the red sea. *The ISME journal*, 11(1):138–151, 2017. URL `http://dx.doi.org/10.1038/ismej.2016.99`.

D. T. Truong, E. A. Franzosa, T. L. Tickle, M. Scholz, G. Weingart, E. Pasolli, A. Tett, C. Huttenhower, and N. Segata. Metaphlan2 for enhanced metagenomic taxonomic profiling. *Nature methods*, 12(10):902–903, 2015.

P. Vaishampayan, C. Moissl-Eichinger, R. Pukall, P. Schumann, C. Spröer, A. Augustus, A. H. Roberts, G. Namba, J. Cisneros, T. Salmassi, et al. Description of tersicoccus phoenicis gen. nov., sp. nov. isolated from spacecraft assembly clean room environments. *International journal of systematic and evolutionary microbiology*, 63(7):2463–2471, 2013.

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, et al. Environmental genome shotgun sequencing of the sargasso sea. *science*, 304(5667):66–74, 2004.

J. Whipps, K. Lewis, and R. Cooke. Mycoparasitism and plant disease control. *Fungi in biological control systems, Manchester University Press, Manchester, United Kingdom*, pages 161–187, 1988.

D. Wood and S. Salzberg. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biol*, 15(3):R46, 2014.

Z. Zhang, S. Schwartz, L. Wagner, and W. Miller. A greedy algorithm for aligning dna sequences. *Journal of Computational biology*, 7(1-2):203–214, 2000.