2016

Computational Approaches for Binning Metagenomic Reads

Ying Wang
University of Central Florida

COMPUTATIONAL APPROACHES FOR BINNING METAGENOMIC READS

by

YING WANG

M.S. University of Central Florida, 2012
M.S. University of Science and Technology of China, 2010
B.S. HeFei University of Technology, 2007

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in Computer Science
in the Department of Computer Science
at the University of Central Florida
Orlando, Florida

Fall Term
2016

Major Professor: Haiyan Nancy Hu

ABSTRACT

Metagenomics uses sequencing technologies to study genetic sequences from whole microbial communities. Binning metagenomic reads is the most fundamental step in metagenomic studies, which is essential for the understanding of microbial functions, compositions, and interactions in environmental samples. Various taxonomy-dependent and taxonomy-independent approaches have been developed based on information such as sequence similarity, sequence composition, or k-mer frequency. However, there is still room for improvement, and it is still challenging to bin reads from species with similar or low abundance or to bin reads from unknown species.

In this dissertation, we introduce one taxonomy-independent and three taxonomy-dependent approaches to improve the performance of metagenomic reads binning. The taxonomy-independent method called MBBC, bins reads by considering k-mer frequency in reads without reference genomes. The first two taxonomy-dependent methods both bin reads by measuring the similarity of reads to the trained Markov Chains from different taxa. The major difference between these two methods is that the first one selects the potential taxa with the taxonomical decision tree, while the second one, called MBMC, selects potential taxa using ordinary least squares (OLS) method. The third taxonomy-dependent method bins reads by combining the methods of MBMC with clustering Markov chains from the assembled reads. By testing on both simulated and real datasets, these tools showed superior or comparable performance with various the state of the art methods. We anticipate that our tools can significantly improve the accuracy of metagenomic reads binning and thus be widely applied in real environmental samples.

ACKNOWLEDGMENTS

This work is supported by the National Science Foundation [grants 1356524, 1149955, and 1218275].

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 INTRODUCTION

Microbes are ubiquitous, and essential to all life [1, 2]. We humans are composed of over 10^{14} microbial cells, more than the 10^{13} human cells, and the majority of the microbial cells reside in our gastrointestinal tract [3]. Although understanding the human genome is essential to understand the human body, sequencing the genomes of our microbes is also necessary [1]. As more than 99% of organism genomes in the environments are not culturable with conventional approaches, a new field called metagenomics has emerged [4-6].

Metagenomics applies genome sequencing technologies to study whole microbial communities without the need of cultivation, which is different from traditional genomics-based approaches [7-9]. A preliminary step in metagenomics is categorizing microbes in terms of their diversity and abundances, which is essential for the understanding of microbial functions, compositions and interactions in environmental samples [10, 11]. Recently developed Next-Generation Sequencing (NGS) technologies have promoted the development of metagenomic research [12], which enable us to study tens of thousands of genomes simultaneously.

Genome sequences from various species in an environmental sample are randomly cut into short DNA fragments and then sequenced, and these sequences (shotgun reads) represent the compositional properties of original genomes [11, 13, 14]. Because many different species exist in an environmental sample, these mixed shotgun reads need to be clustered into distinct species or Operational Taxonomical Units (OTUs) [14], a process known as metagenomic reads binning. We study metagenomic reads binning instead of assembly since the reads are usually very short and incomplete assembling usually result in incomplete genomic analysis [1]. The metagenomic

reads binning problem becomes difficult in highly complex communities with hundreds of species. To efficiently bin large volumes of shotgun reads efficiently, it is therefore critical to develop computational methods for metagenomic reads binning.

Many computational methods have been developed to infer species information directly from the shotgun reads. Based on whether reference databases are needed, these methods can be briefly categorized into two classes [10]. One category is the taxonomy-dependent methods. These methods assign reads based on the similarity of reads with known reference databases or pre-computed models. Currently, most of the metagenomic binning methods belong to this class. A majority of such methods bin reads by do alignment to reference sequences. The typical alignment methods include BLAST [15], BLAT [16], and the reference sequences are from NCBI (http://www.ncbi.nlm.nih.gov/), UniProt (http://www.uniprot.org/), etc. The reads binning methods that are based on the alignment include MEGAN [17], Sort-ITEMS [18], MLTreeMap [19], pplacer [20] and RAST [21], etc. Besides do alignment straightforward, a recent method Kraken [22], counts the k-mers (k base pair long DNA segments) (k=31) appearance in the reference databases, and assigns reads based on the taxonomical tree. Some other taxonomy-dependent methods utilize compositional properties such as GC content, oligonucleotide usage patterns to compare reads to the sequences or pre-computed models in the reference databases. These compositional properties are believed to be preserved across sufficiently long fragments of a genome and vary among different species [23]. Such composition-based methods include NBC [24], TACOA [25], Phymm [26] etc. Some other methods combine alignment and composition information to bin reads, such methods include SPHINX [27], PhymmBL [26], etc. The other

category is the taxonomy-independent methods. These methods employ the difference of GC content, k-mer frequencies, etc., of different microbes in the environmental samples to bin reads, which include CompostBin [28], TOSS [23], AbundanceBin [29], MetaCluster [30], etc. The differences of k-mer frequencies were widely used in these methods, which is based on the observations that k-mer frequency from reads of a genome is usually linearly proportional to that genome's abundance and sufficiently long k-mers are usually unique in each genome [29, 31].

Despite the existence of many read binning approaches, it is still challenging to bin reads without reference genomes and there is much room for improvement of taxonomy-dependent methods [10]. The taxonomy-independent methods have various problems. Early taxonomy-independent methods could not bin short reads from next generation sequencing technologies [32]. Recently, a few methods [26, 29, 30] have been developed to bin reads, including short reads. For instance, AbundanceBin [29] utilizes the property that k-mers in reads from the same genome have similar frequencies to bin reads. Although these methods have been shown to perform well in certain simulated and experimental datasets, recent studies indicate their limitations [10]. One such limitation is that multiple reads have seldom been considered simultaneously to infer their properties other than k- mer frequency. We infer that properties such as Markov properties shared by a group of reads are likely useful to bin short environmental shotgun reads. Taxonomy-dependent methods also have many problems, and are especially limited by the small number of sequenced microbial genomes, more than 99% of which are still unknown and unstudied [32, 33]. Current approaches such as Kraken [22] are unlikely to bin reads from unknown species.

Because of the above limitations for the approaches in the two categories, we proposed one taxonomy-independent and three taxonomy-dependent methods to improve the performance of metagenomic reads binning. Below are brief descriptions of the tools we have developed.

We developed a novel taxonomy-independent approach called Metagenomic Binning Based on Clustering (MBBC). MBBC first groups reads based on k-mer frequencies within the reads by an expectation maximization (EM) algorithm [34]. The rationale behind this step is that species with different genome coverage usually have different k-mer frequencies and k-mers in reads from the same species often occur similar number of times. Therefore, k-mer frequencies in reads help to separate reads from different species. From the initially grouped reads, MBBC then infers the Markov properties of reads within each group, under the assumption that the majority of reads with similar k-mer frequencies are likely from the same genome and therefore from the same Markov chain. Finally, MBBC iteratively clusters reads based on the learned Markov properties and infers the Markov properties of reads in the same groups until the process converges. Tested on twelve simulated datasets, MBBC reliably clustered reads and determined the species number, genome sizes, and k-mer coverage of each species. The k-mer coverage of a species in this study is the average number of reads covering a random k-mer in the genome of this species, which approximates the genome coverage that is calculated as the sum of the length of all reads from this species divided by the genome length of this species. Tested on multiple real experimental datasets, four of which used 75 base pair long short reads, MBBC performed the same or better than two state-of-the-art taxonomy-independent methods. MBBC is thus a useful method for metagenomic studies. However, MBBC often cannot efficiently deal with reads with low abundances or similar

abundances because MBBC assumes that the differences of k-mers vary among species in the environmental samples. Thus, we proposed three taxonomy-dependent methods which can work on such datasets.

We developed a novel taxonomy-dependent method. This method firstly trains each taxon (a group of populations of organisms that can form a unit) to be an 8-th order Markov chain on each taxonomical level (phylum, class, order, family, genus). Then Markov chains of all the taxa on each taxonomical level were clustered to build a taxonomical decision tree. All reads are assigned to each of the five tree. Based on taxa confidences, the most likely taxa were selected and were regarded as potential taxa that the reads were derived from. Finally, all reads were assigned to Markov chains of these potential taxa. We showed that this method usually finds the real genus for about 92% of the datasets that contain known species, and have higher accuracy than other approaches for datasets that contain unknown species. Although this method performed better than other methods, it also has many problems. First, the structure of this method is complex requiring hierarchical clustering and building five taxonomical decision trees for a large number of taxa. Second, it generates many errors at the clustering stage, so assigning the reads through the tree has low accuracy. Third, because of the clustering errors, we need to set a large cutoff to keep as many taxa as possible. Fourth, the large structure of the tree need huge of memory so that we can only stick to 8-th order Markov chains, while such genome representations can result in worse accuracy than higher order Markov chains.

To make the method more efficient and achieve higher accuracy with higher order Markov chains, we proposed another taxonomy-dependent method called MBMC. Different from all

existing methods, MBMC bins reads by measuring the similarity of reads to the trained 9-th order Markov chains for different taxa, instead of directly comparing reads with known genomic sequences. It first selects potential taxa iteratively with the ordinary least squares method which need much less memory and computations, then it assigns reads to the Markov chains of selected taxa by the relative entropy measurement. By testing on more than 24 simulated and experimental datasets with species of similar abundance, species of low abundance, and/or unknown species, we showed that MBMC reliably grouped reads from different species into separate bins. Compared with four existing approaches, we demonstrated that the performance of MBMC was comparable with existing approaches when binning reads from sequenced species, and superior to existing approaches when binning reads from "unknown" species.

However, we observed that MBMC tended to divide reads from an unknown species into multiple small bins and can't achieve very high accuracy for the datasets that contain unknown species. To better bin the reads from unknown species, we proposed the third taxonomy-dependent method. First, we separate all input reads into two categories according to the similarity of long k-mers with reference genomes, one is from known species; the other is from unknown species. For reads that are from known species, we bin the reads by comparing the long k-mers (k=31) with that in reference genomes. For reads that are from unknown species, we bin reads by clustering the Markov chains from contigs that are obtained from the assembly of these reads. Tested on both simulated and real datasets, our method showed great improvement compared with other methods when the reads are from low abundant and unknown species.

In summary, we have developed one taxonomy-independent and three taxonomy-dependent methods to bin metagenomic reads. Although they showed comparable or better performance than other methods, there are still a lot of works to do. MBBC does not perform well when the genome coverage of different microbial species is small (<2 fold difference), which is a common problem in the taxonomy-independent method because such methods rely on the differences of k-mers on different microbes. Although MBMC performs well for datasets when their genome coverage ratio is small, when the datasets contain unknown species, it can't predict the correct number of species and it usually bins reads from one species into multiple groups. Now our problem is how we know that there exists unknown species in the dataset, how many of them are unknown, and how to bin reads better for these unknown species. We have attempted to address this by using higher taxonomical level information, HMM model, shared long k-mers, etc.

The remaining of this dissertation is organized as follows. Chapter 2, we introduce our proposed MBBC method; chapter 3 is our first taxonomy-dependent method that is based on the taxonomical decision tree; chapter 4 is our proposed effective taxonomy-dependent method MBMC. In chapter 5, we introduce our method that utilizes the clustering of Markov chains to bin reads. The last chapter is the conclusions.

CHAPTER 2 MBBC: AN EFFICIENT APPROACH FOR METAGENOMIC BINNING BASED ON CLUSTERING

2.1 Background

Binning environmental shotgun reads is vital in metagenomic studies [14, 35]. In a metagenomics project, genome sequences of different species from an environment are randomly cut into short DNA fragments and then sequenced [11, 14, 35]. The sequenced DNA fragments are often called reads, and the mixed reads from different species in an environment are thus designated as environmental shotgun reads [14]. Because the information of the species origin of the reads and the relative order of the reads in the genomes is lost during the sequencing process, it is crucial to group the mixed environmental shotgun reads into reads from the same species or operational taxonomical units (OTUs), so called "binning reads" [14]. By binning reads, researchers can identify the number and the abundances of species in the environment, and further understand what functional roles each species plays and how these species work together, which are critical for the study of microbes.

Many computational methods have been developed to bin environmental shotgun reads [36-53]. These methods can be broadly classified into two types. One type is similarity-based [37, 40-42, 48-53], in which one queries the reads in reference databases and utilizes the species origin of the hit sequences in the reference databases to bin reads. The reference databases commonly used include the non-redundant nucleotide database at the National Center for Biotechnology Information (NCBI), Uniprot [54], Pfam [55], etc. The other type of methods is composition-based [36, 39, 43-45], in which the composition information of the reads is used to group reads. The

rationale behind composition-based methods is that reads from different species have different composition properties. For instance, different α -proteobacteria species have GC contents ranging from <30% to >60% [56]. In addition to GC content, the frequency of tetranucleotides in reads and other features are also commonly used as the composition information of reads [43, 57, 58].

Despite the existence of many read-binning methods, there remains much room for improvement [58]. The similarity-based methods are hampered by the limited number of sequenced microbial genomes, more than 99% of which are still unknown and unstudied [59]. The composition-based methods also have various problems. Early composition-based methods cannot bin short reads from next generation sequencing technologies [38, 59]. Recently, a few methods [38, 46, 47] have been developed to bin reads, including the short ones. For instance, AbundanceBin [46] utilizes the property that k-mers (k base pair long DNA segments) in reads from the same genome have similar frequencies to group reads. Although these methods have been shown to perform well in certain simulated and experimental datasets, there remains much room for improvement [58]. For instance, multiple reads have seldom been considered simultaneously to infer their properties other than k-mer frequency, whereas properties such as Markov properties are likely useful to bin short environmental shotgun reads, as demonstrated in the following analyses.

We developed a novel approach called Metagenomic Binning Based on Composition (MBBC). MBBC first groups reads based on the k-mer frequencies in reads by an expectation maximization (EM) algorithm [60]. The rationale behind the read grouping based on k-mer

frequencies step is that species with different genome coverage usually have different k-mer frequencies and k-mers from the same species often occur similar times in reads. Therefore, k-mer frequencies in reads help to separate reads from different species. With the initially grouped reads, MBBC then infers the Markov properties of the reads within each group, under the assumption that the majority of reads with similar k-mer frequencies are likely from the same genome and therefore from the same Markov chain. Finally, MBBC iteratively bins reads based on the learned Markov properties and infers the Markov properties of reads in the same groups until the process converges. Tested on twelve simulated datasets, MBBC reliably binned reads and determined the species number, genome sizes, and k-mer coverage of each species. The k-mer coverage of a species in this study is the average number of reads covering a random k-mer in the genome of this species, which is an approximation of the genome coverage that is calculated as the sum of the length of all reads from this species divided by the genome length of this species. Tested on two real experimental datasets, one of which used 75 base pair long short reads, MBBC performed the same or better than two state-of-the-art composition-based methods [46, 47]. MBBC is thus a useful method for metagenomic studies.

2.2 Materials and Methods

2.2.1 Two Real Experimental Datasets Retrieved

We used two real experimental datasets. One was the Acid Mine Drainage (AMD) dataset [11] downloaded from ***http://www.ncbi.nlm.nih.gov/books/NBK6860/***. This dataset contained 180,713 single-end reads, with an average read length of 1005 base pairs long. As in the previous study [38], we used the Figaro software [61] to remove the vector sequences in these reads. For

the remaining portions of each read, only the longest contiguous bases whose quality values were >=17 were kept [38]. This filtering resulted in 166,715 reads. These 166,715 reads were then mapped to two dominant species using the MuMmer software [62] with the default parameters. In total, 40499 reads were mapped to the two species and used to test the binning methods.

The other real dataset was the human gut dataset from 15 randomly selected samples and downloaded from ***ftp://public.genomics.org.cn/BGI/gutmeta/High_quality_reads/***. There were 257,158,754 paired-end reads in this dataset, each of which was 75 base pairs long. These reads were mapped to the following three species using the software SOAP 2.21 [63]: *Bacteroides uniformis*, *Alistipes putredinis*, and *Ruminococcusbromii L2-63*. These species were used because they were the most abundant species and/or had more complete genome sequences in the gut dataset. The command used to map reads was ./soap –a <reads_a> -b <reads_b> -D <index.files> -o <PE_output> -2 <SE_output>, which allowed two mismatches and indels during mapping. There were 4,684,098 reads mapped to the three genomes and used to test the metagenomic binning methods.

2.2.2 Twelve Simulated Datasets Generated

To generate simulated datasets, we randomly selected three genera that had more than 20 sequenced species in the NCBI Microbial Genome Database (http://www.ncbi.nlm.nih.gov/genomes/MICROBES/microbial_taxtree.html). The three genera selected were Lactobacillus, Spiroplasma, and Bartonella. Next, from each genus, we randomly selected four species to generate simulated datasets. Note that it is much more challenging to bin reads from species of the same genus than those from different genera. We then generated paired-

end reads using MetaSim [64] for each of the three or four species in a dataset, with the given genome coverage. We specified the read length to be 75 base pairs and simulated the reads with no error or with the empirical error model in MetaSim (~1% error rate).

2.2.3 The Framework of the MBBC Method

We developed a novel method called MBBC (Figure 1).

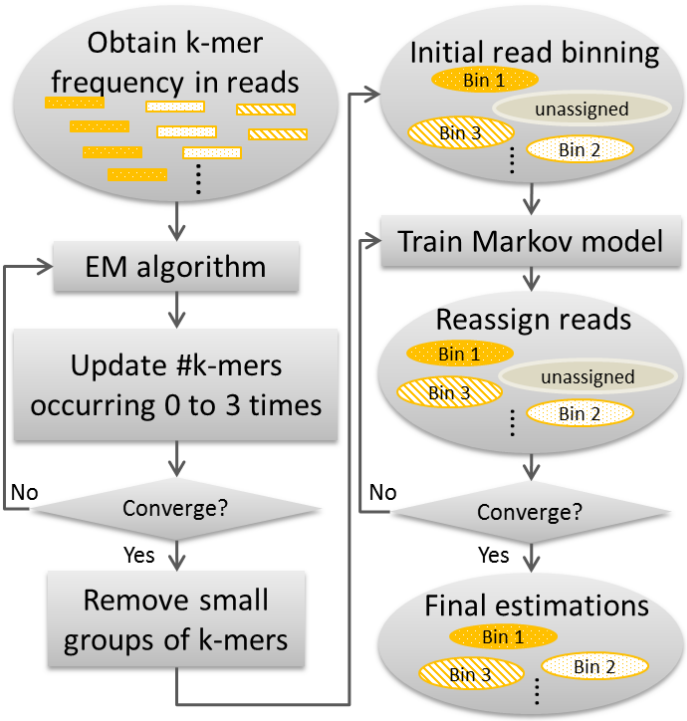

Figure 1 the procedure of read clustering in MBBC. The output on the right from each of the main steps on the left is connected with the corresponding steps.

Our method starts from an EM algorithm to bin k-mers in reads based on their frequencies in reads. The assumption behind the binning is that the frequency of k-mers in reads follows a mixture of Poisson distributions [65]. Next, MBBC iteratively estimates the number of k-mers that occur 0 to 3 times in reads and runs the EM algorithm to estimate the parameters of the mixed

Poisson distributions. The reason to iteratively estimate these numbers is that they are either unobserved or inaccurate and thus affect the estimation of other parameters [46, 47]. Next, MBBC determines the species number and initially bins reads based on the Poisson parameters. MBBC then iteratively models the Markov property of the reads in each group and reassigns reads to groups. Finally, MBBC determines the genome sizes and other metrics based on the assigned reads and the estimated parameters. The details are given in the following.

2.2.4 EM Algorithm for Initial Binning of Reads

We developed an EM algorithm to bin reads based on the frequency of k-mers in reads, where k=16 is chosen so that the chance that a random k-mer occurs multiple times in a microbial genome is small (<1e-5). The underlying assumption of this EM algorithm is that the frequency of k-mers in reads from a microbial species follows a common distribution. Similar to previous studies [46, 65], we use Poisson as the common distribution. Under this assumption, all k-mers in reads from a metagenomics project form the samples of a mixture of Poisson distributions, where the number and the parameters of the Poisson distributions are unknown. EM algorithms are widely used to address mixture problems [65, 66], and therefore applied to initially group reads from different Poisson distributions.

The EM algorithm in MBBC assumes that there are in total *n* different k-mers in reads in a metagenomics project that are from *m* different species, where *m* is unknown. Assume that the frequency of these k-mers in all reads, $x_1, x_2, ..., x_n$, follows a mixture of *m* Poisson distributions with the unknown parameters $\lambda_1, \lambda_2, ..., \lambda_m$. For any i from 1 to *n*, if x_i is from the j-th Poisson

distribution, then $P(x_i = x) = \alpha_j p_j(\lambda_j, x) = \alpha_j \frac{\lambda_j^x}{x!} e^{-\lambda_j}$, where α_j is the probability that a random k-mer is from the j-th distribution and $\sum_{j=1}^{m} \alpha_j = 1$. Intuitively, $\alpha_1, \alpha_2, ..., \alpha_m$ represent the relative species abundance in the environment, and $\lambda_1, \lambda_2, ..., \lambda_m$ represent the k-mer coverage of the species. Because we do not know which distribution x_i is from, we define the missing variable y_i, where $y_i = j$ indicates that x_i is from the j-th Poisson distribution. With the above notations, the log complete likelihood function of the observed data $X = \{x_1, x_2, ..., x_n\}$ and the missing data $Y = \{y_1, y_2, ..., y_n\}$ is $\log(L(\theta; X, Y)) = \sum_{i=1}^{n} \log(\alpha_{y_i} * p_{y_i}(\lambda_{y_i}, x_i))$, where the parameter θ = { $\alpha_1, \alpha_2 \cdots, \alpha_m$; $\lambda_1, \lambda_2, \cdots, \lambda_m$ }. The E-step of the EM algorithm is to calculate Z_{ij}, which is $Z_{ij} = P(y_i = j \mid X, \theta) = \frac{\alpha_j * p_j(\lambda_j, x_i)}{\sum_{r=1}^{m} \alpha_r * p_r(\lambda_r, x_i)}$. The M-step is to estimate the parameters in the following manner: $\alpha_j = \frac{1}{n} \sum_{i=1}^{n} Z_{ij}$, $\lambda_j = \frac{\sum_{i=1}^{n} Z_{ij} x_i}{\sum_{i=1}^{n} Z_{ij}}$.

For a given m, to apply the above EM algorithm, we initialize $\alpha_j = 1/m, \lambda_j = j * 10 + 10$ for j from 1 to m. We then iterate the E-steps and M-steps until the difference between the updated θ and the current θ is small (<1e-5). Finally, we output the current θ = { $\alpha_1, \alpha_2 \cdots, \alpha_m$; $\lambda_1, \lambda_2, \cdots, \lambda_m$ } and assign k-mers to m different groups based on θ.

2.2.5 Estimation of the Species Number

The species number m is unknown and required by the above EM algorithm. To estimate *m*, MBBC initializes *m* as a large number so that the output groups from the EM algorithm contain at least a small group that is too small to serve as a k-mer group from a microbial species. To determine whether an output group is small, MBBC first estimates the number of k-mers that occur x=0, 1, 2, and 3 times, respectively, with the following formula: $\sum_{j=1}^{m} \frac{p_j(\lambda_j, x) \sum_{i=1\&x_i \geq 4}^{n} Z_{ij}}{1 - \sum_{s=0}^{3} p_j(\lambda_j, s)}$. With the estimated number of k-mers that occur x=0, 1, 2, and 3 times, MBBC iteratively runs the EM algorithm using the estimated x_i for i from 0 to 3 and the original x_i for i>3 until the estimated x_i for i<4 do not change. The rationale to iteratively estimate x_i for i<4 is that these x_i are inaccurate because of the existence of low abundance species and sequencing errors [46, 58, 65]. Next, MBBC estimates the genome size represented by each group of k-mers output from the EM algorithm as $\frac{\sum_{i=1}^{n'} Z_{ij} * x_i}{\lambda_j}$, for j from 1 to m, where n' is used to denote that the estimated k-mers that occur fewer than 4 times are used together with other observed k-mers. Finally, MBBC labels groups of k-mers as small groups if their estimated genome sizes are smaller than 400,000, a cutoff that is smaller than the size of the sequenced smallest genome of living organisms [46], and labels other groups as large groups. With the labelled groups, MBBC estimates the species number as the number of the large groups. The α_j for the large groups are normalized so that their sum is

equal to 1. To take the k-mers initially assigned to small groups into account, MBBC then implements one more E-step to calculate Z_{ij} and then updates $\alpha_j = \frac{1}{n}\sum_{i=1}^{n} Z_{ij}$, for i from 1 to n and j from 1 to m.

2.2.6 Initial Read Assignment Based on the Inferred θ

With the inferred θ, MBBC measures the probability that a read belongs to the j-th species as $p_j(\lambda_j, x)$, for j from 1 to m, where x is the median frequency of the k-mers in this read. MBBC then sorts these probabilities from largest to smallest for each read. For a read, if its largest probability minus the second largest probability is larger than a cutoff C (C=0.5), this read will be assigned to the species corresponding to the largest probability. When paired-end reads are used in a project, MBBC assigns the two paired reads to the same species when at least one read can be assigned and there is no conflict between the assignments of the two reads. In this way, MBBC obtains m+1 groups of read, one of which corresponds to the unassigned reads. In case that there are more than 50% of reads unassigned, MBBC reduces C by 0.01 and repeats this process until at least half of the reads in the datasets are assigned to the m groups that correspond to m species.

2.2.7 Final Read Assignment Based on the Markov Property

The final assignment of reads is performed by iteratively inferring a 5-th order Markov chain for each group, except for the group corresponding to unassigned reads, and reassigning reads to each group. The rationale of modelling a group of reads by a Markov chain is that most reads in each of the m groups are likely from the same species and Markov chains are widely used to model the microbial genome sequences [67, 68]. In brief, starting from the initially assigned

reads in a group, MBBC counts 6-mer frequencies in these reads to obtain the transition matrix and the stationary probability of the Markov chain. Next, MBBC scores all reads in this group using the inferred Markov model and obtains the beta percentile of the score distribution. This percentile is used as a cutoff to determine whether a read belongs to a species. The beta used by MBBC in all tested datasets is 10%. MBBC then scores each read with the m trained models and finds the model with the best score for each read. If the best score is larger than the corresponding cutoff, this read is assigned to the species corresponding to the best score. Otherwise, the read is not assigned. With all reads scored and assigned, we have a new set of *m+1* groups of reads and infer the Markov models for the m groups again. This process of inferring the Markov models and assigning reads is iteratively implemented, with the beta decreased to beta/2 after one iteration, until the assigned reads in the m+1 groups do not change. With the final assigned reads, MBBC estimates the genome size of each species using the total number of k-mers in each group divided by the estimated k-mer coverage.

2.2.8 Comparisons with AbundanceBin and MetaCluster 5.0

To run MBBC, we used the following command for each dataset: java –jar –Xmx7g MBBC.jar –i reads_file -m species_number –r read_type, where m was set to be 10, and the read_type=1indicates single-end reads and read_type=0 means paired-end reads. With the known species number *m* as input, we ran AbundanceBin [46] using the command "./abundancebin -input reads_file -bin_num m". We ran MetaCluster 5.0 [47] by the command "./ MetaCluster5_1 reads_file --Species m'" for species with the genome coverage smaller than 6 first and then using the command "./ MetaCluster5_2 reads_file.2 --Species m'' for the species with the genome

coverage larger than 6. When the species number was assumed to be unknown, we ran AbundanceBin and MetaCluster 5.0 using the command "./abundancebin -input reads_file -RECURSIVE_CLASSIFICATION" and "./ MetaCluster5_1 reads_file" followed by "./ MetaCluster5_2 reads_file.2", respectively.

2.3 Results

2.3.1 MBBC Reliably Estimates the Species Number, Genome Sizes, Relative Species Abundances, and k-mer Coverage

We applied MBBC to 12 simulated datasets with the initial species number, *m*, set to be10. These datasets used species from 3 randomly selected genera, from each of which 4 species were randomly selected. We observed that in each dataset, MBBC predicted the exact species number. In all datasets, regardless of whether the genome coverage ratio was larger or smaller than 2, whether there were errors in reads, the predicted genome size, relative species abundance, and k-mer coverage were close to the actual ones.

Figure 2 provided a detailed example of binning reads from four species in the genus of *Spiroplasma* by MBBC (Figure 2). In this example, the genome coverage of the four species was 4, 8, 18, and 32, respectively. MBBC correctly determined the species number. It also reliably predicted the k-mer coverage as 3.34, 6.67, 13.05, and 22.98, respectively, which were close to the actual ones (numbers in the parentheses in Figure 2). The actual k-mer coverage was calculated by counting the number of times the k-mers in a genome covered by reads from this genome. Moreover, MBBC reasonably estimated the genome sizes for the four species (Figure 2).

Initially predicted α, λ										
Species ID	1	2	3	4	5	6	7	8	9	10
α	43.30%	22.97%	11.07%	20.84%	1.16%	0.51%	0.12%	0.03%	0.00%	0.00%
λ	3.88	11.14	16.57	23.61	38.71	51.62	74.22	105.37	158.79	329.53
After updating #k-mers that occur 0 to 3 times										
α	31.59%	16.01%	25.79%	24.33%	1.38%	0.72%	0.14%	0.03%	0.01%	0.00%
λ	3.34	6.67	13.05	22.98	35.61	49.23	72.22	103.45	156.95	328.64

After removing small groups of k-mers				
Genome size	2694580	922216	941606	1121982
α	16.93%	11.55%	23.09%	48.43%
λ	3.34	6.67	13.05	22.98

After iteratively binning reads based on Markov chains : Predicted (real data)				
Genome size	1498994 (1160554)	825923 (945296)	1138156 (1107344)	1212248 (1075140)
α	9.42% (6.98%)	10.35% (11.36%)	27.91% (29.95%)	52.33% (51.70%)
λ	3.34 (3.49)	6.67 (5.83)	13.05 (12.48)	22.98 (20.52)

Figure 2 An example of binning reads from four species in the genus of Spiroplasma by MBBC. α and λ represents the estimated relative species abundance and k-mer coverage, respectively. The real genome sizes, α and λ are listed in the parentheses of the last table in the figure. After updating k-mer occurrences for k-mers occurring fewer than 4 times, the estimated α becomes more accurate. After removing small groups, the estimated species number and α become more accurate.

It is also evident that two steps in the EM algorithm of the MBBC are important for its accuracy (Figure 2). One step is to estimate the number of k-mers occurring 0, 1, 2, and 3 times in reads. After estimating these numbers by iteratively running the EM algorithm, the estimated k-mer coverage becomes much closer to the actual ones. The other step is to remove the small groups of k-mers (the estimated genome sizes corresponding to these groups are smaller than 400,000). By removing these small groups and reassigning k-mers, the estimated species abundance becomes much closer to the actual abundance. These two steps make the EM algorithm in MBBC different

from the one implemented in AbundanceBin [46], which always separates k-mers into two groups, even when reads are from more than two species, and neglects the inaccuracy of the observed numbers of k-mers occurring 0, 1, 2, 3 times in reads.

Figure 2 illustrates the importance of the inferred Markov properties to the accuracy of MBBC as well. It is well known that different microbial genomes often follow different Markov properties [69, 70]. Previous studies, such as [43], have utilized such properties to assign reads of longer than 1000 base pairs in metagenomic studies. Regarding short reads, such as 75 base pairs long reads, it is unlikely to reasonably infer the Markov properties they may have from individual reads. By assuming that most reads grouped by the EM algorithm are likely from one species or OTU, we have reliably inferred the Markov properties that most reads in a group follow and further filtered reads from other species or OTUs. To our knowledge, such a strategy has not been explored before. From Figure 2, it is clear that this strategy significantly improves the accuracy of read binning, as is shown in the generally more accurate estimation of the genome sizes and relative species abundance.

To investigate how the change of genome coverage ratios affects the accuracy of the estimation, we applied MBBC to simulated datasets with all genome coverage ratios larger or smaller than 2, using the first three species in the above example. The above example demonstrated that MBBC reliably estimates the species number, genome sizes, relative species abundance, and k-mer coverage. We noticed that the species number was still accurately predicted even when the genome coverage ratios were smaller than 2 (Table 1). Moreover, as expected, we observed that when the genome coverage ratios were larger than 2, the predicted genome sizes and k-mer

coverage were in general closer to the actual ones than those with genome coverage ratios smaller than 2 (Table 1). In addition, the prediction still agreed well when the genome coverage ratios were smaller than 2. For instance, for the third species (*sps*), the predicted genome size, relative species abundance, and k-mer coverage was 1,139,322 base pairs, 0.5202, and 10.95, respectively, whereas the actual one was 1,107,344 base pairs, 0.5541, and 10.53, respectively (Table 1).

Table 1 Prediction by MBBC on datasets with different genome coverage ratios or species

Datasets	Predicted genome size	Real genome size	Predicted relative abundance	Real relative abundance	Predicted k-mer coverage	Real k-mer coverage
spa4spd8sps18spt32	1498994	1160554	9.42%	6.98%	3.34	3.49
	825923	945296	10.35%	11.36%	6.67	5.83
	1138156	1107344	27.91%	29.95%	13.05	12.48
	1212248	1075140	52.33%	51.70%	22.98	20.52
spa4spd8sps18	1281577	1160554	16.16%	14.45%	3.24	3.49
	921307	945296	22.61%	23.53%	6.31	5.83
	1226752	1107344	61.23%	62.02%	12.83	12.48
spa5spd8sps15	1607360	1160554	27.03%	19.36%	4.03	4.01
	682864	945296	20.95%	25.23%	7.36	5.83
	1139322	1107344	52.02%	55.41%	10.95	10.53
spa5baa8sps15	1463372	1160554	21.50%	16.49%	4.13	4.01
	1318685	1596490	30.49%	36.30%	6.51	5.87
	1250815	1107344	48.01%	47.21%	10.80	10.53

Each species in each dataset is named by the first two letters of their genus name, followed by the first letter from the species name and then the genome coverage. The first dataset is the one used in Figure 1. The predictions are listed in the order of the species names in the dataset name.

We also investigated the performance of MBBC with species from different genera. Intuitively, it should be easier to bin reads from species of different genera than those from the same genus, because the Markov properties of genomes from different genera may be more

different than those from the same genus. When we replaced the second species in the third example above with a species from another genus, we noticed an improvement in the accuracy by MBBC (Table 1). For instance, the estimated k-mer coverage of the replaced species was 6.51, compared with 5.87, the actual k-mer coverage of this species. Conversely, the estimated k-mer coverage of the second species before replacement was 7.36, compared with the actual k-mer coverage of 5.83.

2.3.2 MBBC Reliably Assigns Reads

In addition to estimating species number, genome sizes, and k-mer coverage, another important problem in metagenomic analyses is to group reads from the same species or OTUs together. We investigated how well MBBC binned reads in 12 simulated datasets. We observed that 75% to 91% of reads were correctly binned together, even when there was 1% errors in reads and some genome coverage ratios were smaller than 2. The accuracy of the binned reads was calculated by assuming the species to be the group with the majority of its reads and then counting how many reads were correctly assigned to that species. We also noticed that the accuracy was genus dependent, in that the accuracy of the binned reads for simulated datasets from one genus was always higher than that from another genus, regardless of whether the genome coverage ratios were smaller than 2 or there were errors in reads. In addition, the genome coverage ratios affected the accuracy of read binning, in that the accuracy for datasets from the same genus was always lowest when the ratios were smaller than 2.

To further investigate how the genome coverage ratios affected accuracy, we applied MBBC to datasets with different genome coverage ratios. We used the same datasets listed in

Table 1. As expected, we observed that the accuracy of read binning decreased when the genome coverage ratios decreased. We also noticed that the accuracy was improved with species from different genera, although the genome coverage ratios were still smaller than 2, because of the consideration of the Markov properties of genomes of different species from different genera. For instance, for the last two simulated datasets in Table 1, the accuracy of read binning by MBBC was 85.39%, compared with 82.01%, when species from different genera compared with species from the same genus were used.

2.3.3 MBBC Works Well in Real Datasets

We applied MBBC to two real datasets. One was the AMD dataset [11], in which Sanger reads were used, with an average read length of 707.95 base pairs. MBBC correctly predicted the species number as 2. MBBC also almost perfectly predicted the relative abundances of the two species as 29.1% and 70.9%, versus the actual relative abundance as 29.03% and 70.97%. Moreover, the predicted k-mer coverage of the two species were 4.03 and 8.16, respectively, which were close to the actual coverage (5.14 and 7.35). Overall, the accuracy of read binning by MBBC in this dataset was 94.27%.

The other real dataset we applied MBBC to was a human gut dataset composed of 4,684,098 Illumina reads an average of 75 base pair long from three species. Unexpectedly, MBBC predicted that there were 4 species (Table 2). After scrutinizing the results, we noticed that the majority of reads in both the third and fourth groups were from the same species, the third species. Moreover, the sum of the relative abundance of the third and fourth groups was 72.48%, which was close to the relative abundance of the third species, 69.21%. The other two predicted groups

agreed well with the other their corresponding two real species. For instance, the predicted genome size, relative coverage, and k-mer coverage of the second species were 231555 base pairs, 16.87%, and 10.24, respectively, which concurred well with the real corresponding numbers, 2249085, 16.67%, and 10.24 (Table 2). The accuracy of read binning by MBBC was 74.80% in this dataset, demonstrating that MBBC works well in datasets with long Sanger reads or short Illumina reads.

Table 2 Prediction on the human gut dataset by MBBC

	MBBC Predictions				Real data		
genome size	3524796	2315047	1745685	2274392	NA	2249085	NA
relative abundance	11.25%	16.87%	23.33%	48.55%	14.12%	16.67%	69.21%
k-mer coverage	4.48	10.24	18.78	30	8.28	10.49	18.49

To understand why MBBC did not automatically combine the third and fourth groups into one predicted species, we examined the mapped reads to the genome corresponding to the third species. We noticed that this genome was almost evenly divided into two halves, with coverage of approximately 18 and 30 for the two halves, respectively. Because both halves were longer than the genome size cutoff (400,000), MBBC considered them as two separate genomes. Because the two groups were from the same genome, we also compared the two Markov models learned from the reads from the two halves of the genome. We used the relative entropy to measure the difference of the transition matrix of the two Markov chains. We observed that the relative entropy of the two Markov models was 1.14, which was larger than that of the Markov models of the first two species, with a relative entropy of 0.68. It thus makes sense that MBBC considered them to be two separate species. This result also implies that different compositions in different genome regions may contribute to the different coverage of these regions in genome sequencing.

2.3.4 MBBC Performs Better than AbundanceBin and MetaCluster

We compared MBBC with two widely used composition-based methods, AbundanceBin [46] and MetaCluster 5.0 [47], in the 12 simulated datasets and 2 real datasets mentioned above. Because AbundanceBin was developed for single-end reads, when paired-end reads were used in a dataset, we ran AbundanceBin by treating the two paired-end reads as independent reads. Because MetaCluster runs on paired-end read data, we did not apply it to the AMD dataset that used single-end reads [11]. Overall, MBBC outperformed the two methods in terms of the estimated species number, genome sizes, relative species abundance, k-mer coverage, and binning accuracy. See the following for details.

First, we compared the predicted species number in these 14 datasets. MBBC predicted the right species number in all except one dataset. AbundanceBin and MetaCluster often cannot predict the right species number. Of the 12 simulated datasets, AbundanceBin and MetaCluster correctly predicted the species number in 2 and 0 datasets, respectively. For the AMD dataset, AbundanceBin predicted the correct number of species. For the gut dataset, MetaCluster predicted 512 groups whereas AbundanceBin failed with one bin output. Because the species numbers were not correctly predicted, it was difficult for the two programs to predict other properties of the datasets, such as the genome sizes, the relative abundance, and the k-mer coverage of each species.

Next, we compared the accuracy of the read binning in the 14 datasets (Table 3). Because AbundanceBin and MetaCluster cannot automatically predict the right species number, we specified the known species number as input for the two programs to output the binned reads. In 11 of the 12 simulated datasets, the accuracy of MBBC was better that of the other two methods,

with a median of 14.95% higher accuracy (Table 3). In the only simulated dataset that MBBC did not achieve the highest accuracy, MBBC had an accuracy of 89.09%, slightly less than the best accuracy of 90.44%. AbundanceBin performed better than MetaCluster in all simulated datasets without read errors while MetaCluster performed better than AbundanceBin in simulated datasets with read errors (Table 3). MBBC performed better in terms of estimating the genome sizes, relative species abundance, etc. In the two real datasets, we also observed that MBBC had a higher accuracy than the other two methods. For instance, the accuracy of MBBC in the gut dataset was 74.80%, compared with 52.63% and 71.65% by AbundanceBin and MetaCluster, respectively.

Table 3 Binning accuracy of MBBC, AbundanceBin and MetaCluster

Datasets	MBBC	MetaCluster	AbundanceBin
lag5lar11las24	91.34%	82.93%	64.60%
lag4lar7las12	78.97%	77.66%	39.09%
laa4lag8lar15las30	86.43%	83.49%	50.98%
laa4lag8lar15las30(no errors)	87.13%	85.64%	86.41%
spa4spd9sps18	89.58%	78.68%	63.73%
spa5spd8sps15	82.01%	73.71%	52.44%
spa4spd8sps18spt32	87.35%	72.64%	54.60%
spa4spd8sps18spt32(no errors)	89.09%	74.43%	90.44%
baa3bab7bac15	79.55%	64.83%	61.11%
baa6bab10bac18	75.80%	45.12%	51.13%
baa5bab10bac18bah30	75.71%	34.48%	39.25%
baa5bab10bac18bah30(no errors)	79.90%	45.82%	66.25%
human gut dataset	74.94%	71.65%	52.63%
AMD dataset	94.14%	na	73.42%

Finally, we compared the speed of the three methods to bin reads in the 14 datasets. All comparisons were performed on the same computer with the following configuration: Intel ® Core™ i5-3210M CPU @ 2.50GHz and 8G RAM. The stacked bars in Additional file 4 displayed

the running time of each method on these datasets. We observed that when species number was unknown, the other two methods usually required much more time. When species number was known, MetaCluster was faster (~36.40%) than MBBC, but it only binned reads, and did not predict more parameters, such as genome sizes, relative species abundance, etc. The most time-consuming part of MBBC was the step to update the number of k-mers occurring 0, 1, 2, and 3 times in reads. AbundanceBin was slow even when the species number was known. This update process required more time to converge, which occupied nearly half of the total running time. Given that MBBC can predict more parameters than MetaCluster, runs faster than AbundanceBin, and can automatically and accurately predict the species number, MBBC is a useful tool for metagenomics data analyses.

2.4 Discussion and Conclusions

We developed a novel approach called MBBC to bin reads from metagenomics projects. MBBC bins reads by employing two types of read composition properties that have never been considered together previously. Tested on simulated and experimental datasets, we demonstrated that MBBC could reliably determine the species number, genome sizes, relative species abundance, and k-mer coverage. Moreover, MBBC grouped reads from the same species with high accuracy. Compared with two other popular composition-based methods, MBBC performed better in almost every dataset tested, with higher accuracy of read binning in both simulated and real datasets.

The inferred Markov property from the binned reads contributes significantly to the success of MBBC. We demonstrated in the above that the Markov property helped to group reads by

exploring the differences among species and genera. The Markov properties also help MBBC work better with errors in reads. This improvement is because the majority positions in a read from a species still follow the Markov properties, despite the existence of a few error positions.

The comparison of MBBC with AbundanceBin and MetaCluster may be biased by the parameters we used. Except for specifying *m* as the known species numbers, we used the default values of the other parameters in running AbundanceBin and MetaCluster. It is thus possible that the two tools may produce better results with other parameter choices. However, we believe that MBBC should at least behave similarly to or better than the two methods, as the Markov properties of the grouped reads that are important for correctly binning have not been utilized by the two tools.

We suggest users use a large *m* as the initial species number. However, how should one determine this large initial *m*? An economical approach is to start with *m* as a smaller number such as 10. If no small group is discovered by the EM algorithm, one can then increase *m* slightly, such as *m*=15, until the EM algorithm produces small groups. This process will result in the robust binning of the reads.

Two aspects may be considered to further improve the MBBC method. One is the assumption of the Poisson distribution of the frequency of k-mers in reads. The k-mers in reads from one species may not follow a Poisson distribution exactly, and more suitable distributions may be explored. However, from our study on 14 datasets, it seems that this assumption does not substantially affect the prediction. The other aspect is the assumption of the homogeneity of a microbial genome. We previously demonstrated that the third species in the gut dataset is not

homogeneous, which is why MBBC considered it to be two different species. In the future, a better model will be necessary to take the homogeneity of microbial genomes into account when designing composition-based binning methods.

In sum, we developed a novel method for binning metagenomic reads based on composition. This method was demonstrated to reliably predict the species number, genome sizes, relative species abundance, and k-mer coverage. It also displayed a high accuracy in read binning. The free tool implementing the developed method is available at ***http://eecs.ucf.edu/~xiaoman/MBBC/MBBC.html***. Although MBBC showed better performances than other methods in reads binning, it usually can't efficiently deal with reads with low abundances or similar abundances because MBBC is based on the assumption that the differences of k-mers vary among species in the environmental samples. Thus, we proposed a taxonomy-dependent method which can work on such datasets.

CHAPTER 3 BINNING METAGENOMIC READS BASED ON THE TAXONOMICAL DECISION TREE

3.1 Background

Binning metagenomic reads is essential in metagenomic studies. Many computational methods have been developed to bin reads in metagenomic projects. These methods can be broadly classified into two categories. One category is the taxonomy-dependent methods, which compare reads with sequences in public databases or pre-computed models to bin reads. The other category is the taxonomy-independent methods, which employ the difference of GC content, k-mer frequencies, etc., of different microbes in the same environmental samples to bin reads. For instance, AbundanceBin [29] and MBBC [71] utilize the frequency difference of k-mers from different microbes in environmental samples to bin reads, which have been shown to successfully separate reads from species with very different abundance.

Despite the existence of many methods for read binning, it is still challenging to bin reads from metagenomic projects accurately and efficiently. The difficulty commonly arises when the genome coverage of different microbial species is similar which renders it unlikely to separate reads from species with low or similar abundance by taxonomy-independent methods such as AbundanceBin [29] and MBBC [71]. As a result of these limitations, more powerful read binning methods are expected.

We developed a novel taxonomy-dependent approach. Different from all existing methods, this method firstly infers the potential taxa by assigning reads to five taxonomical decision tree, then it bins reads by measuring the similarity of reads to the trained 8-th order Markov chains for

the different potential taxa. The results showed that this method have better accuracy than other popular approaches, and it can bin reads better for reads with similar or low abundances as well as reads from unknown species.

3.2 Materials and Methods

3.2.1. Reference Genomes and Their Representation

We downloaded all 2773 completely sequenced microbial genomes from ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/. We also downloaded their taxonomical information from GLOD (***https://gold.jgi-psf.org/***) [72], including the species, genus, family, order, class, and phylum IDs of these species. Only 2592 genomes had all six-level taxonomical information and were used. In total, we had 32 phyla, 56 classes, 110 orders, 239 families, 608 genus, and 1330 different species (multiple genomes may be from the same species).

We represented each genome by an N-th order Markov chain (N from 5 to 10). We calculated the stationary and transition probabilities of the N-th order Markov chain by counting the N-mer and (N+1)-mer frequencies on both the positive and negative strands of the genome under consideration. A pseudocount, 0.0001, was added to each entry of the stationary and transition probabilities to avoid any entry in the stationary and transition probabilities to be zero. For a taxon with multiple sequenced genomes, such as a phylum, we obtained its stationary and transition probabilities by averaging the corresponding stationary and transition probabilities of the sequenced genomes that belong to this taxon. We chose the 8-th Order Markov chain to model the microbial genomes after comparing with Markov chains of other orders, regarding the memory usage and computations.

3.2.2. Decision Tree

At each of phylum to genus level, in order to largely reduce the calculation of reads assignment to each taxon, we build a decision tree in each level which were based on the Markov chain clustering.

3.2.2.1 Markov Chain Clustering

We first decide the clustering linkage. Each species is represented by an 8-th order Markov chain. The distance of each pair of Markov chains is measured by the relative entropy of both transition and stationary probabilities, that's, the average values calculated from both directions: [D(P||Q)+D(Q||P)]/2, in which, D(P||Q) calculation is shown below, P[i] and Q[i] were the transition or stationary probabilities from each pair of Markov chains.

$$P\ (\vee\ |Q) = \sum_{i} ln\left(\frac{P[i]}{Q[i]}\right) P[i]$$
$$D$$

We used six common linkage strategies to do the hierarchical clustering as shown in Table 4. The optimal decision tree will have the smallest size so that we can assign reads by having less comparisons. Table 4 showed the comparisons between different linkage strategies when using the clustering results from 8-th order Markov Chains for all 2773 species.

Table 4 Comparisons of hierarchal clustering linkage strategies

		WardLinkage	Average Linkage	Centroid Linkage	CompleteLinkage	MedianLinkage	SingleLinkage
number of species in each node, each node has (left_child,right_child), and the nodes in the same level are separated by space	level 1	2025,748	2771,2	2771,2	1615,1158	2771,2	2771,2
	level 2	1115,910 717,31	2770,1 1,1	2770,1 1,1	1613,1 1155,1 1,1	2770,1 1,1	2770,1 1,1
	level 3	1044,71 700,210 177,540 5,26	2769,1	2769,1	1614,1 1156,2	2769,1	2769,1
Tree structure	# nodes	597*2	1040*2	1608*2	594*2	1435*2	1600*2
	tree height	45	883	1600	83	987	1589

From above table, we know that Ward Linkage is the best choice, as it can divide the species more evenly and the tree has the smallest size.

3.2.2.2 Build decision tree

The stationary and transition probabilities for each taxon were obtained by averaging the probabilities of all species in each taxon. The distance between every pair of taxa were calculated using both stationary and transition probabilities. Based on the Ward Linkage, we did hierarchical clustering for these taxa, that's, initially each taxon is a cluster, the closest two clusters were merged into one new cluster first, then among the left clusters and this new cluster, the closest two clusters were merged into one new cluster. At last, these taxa were merged into one cluster, which is the root. So from the root to leaf, we recorded the taxa that were grouped together in each node. In this way, one binary tree was built based on the clustering results. In each tree, after we have

the taxa information in each node, we calculated the Markov chain in each node by averaging the transition and stationary probabilities for all the taxa in this node. Then, we build a decision tree with four attributes [left child, right child, Node ID, Node Markov chain] in each node.

When we assign one random read, it will be input into each tree from root to leaf. The assignment scores for both ends and their reverse complement were calculated by sum the log of both stationary and transition probabilities for one read. The read was assigned to the node with higher score. The score was calculated in this way: score1=score in one end; score11=score in this end's reverse complement; score2=score in another end; score22=score in this end's reverse complement; The score for this read=max[(score1+score22)/2,(score11+score2)/2]. In each tree, one path is determined by the larger assignment scores, and the leaf IDs will be recorded in each of five tree.

3.2.3. Confidences

3.2.3.1. Random Reads Used to Calculate Confidences:

The random reads were generated for each of five tree as following step: For each taxon, we selected at most 10 species based on the relative entropy between Markov chains. From the closet to farthest species for one taxon, evenly, at most 10 species were selected. For each of these species, all possible 75-bp segments (single-end reads) from genome sequences were considered as their reads. In this way, we can generate large number of reads for each tree, the number of reads ranges from 557413571 to 5442512809. These reads were assigned to each of the decision tree. Then we can calculate the confidences based on the accuracy of reads assignment.

3.2.3.2. Confidence Calculations:

For each node in each tree, we can obtain the assigned number of reads as well as real number of reads. To calculate the sensitivity, precision for each node, we first need to calculate the true positive (TP), which is the number of correctly assigned reads; the false negative (FN), which is the number of reads that belong to this node but were assigned inaccurately; the false positive (FP), which is the number of reads that did not belong to this node but were assigned to this node. Then sensitivity=(TP)/(TP+FN), precision=(TP)/(TP+FP). We also calculate F1 score which considers both sensitivity and precision, and F1 score=2TP(2TP+FP+FN).

3.2.4. Metagenomic Binning Based on the Decision Tree

Our method firstly assigns reads to five decision tree (phylum tree, class tree, order tree, family tree and genus tree). By using certain strategy, we select the limited number of genus candidates out of all 608 genera based on the reads assignment results from five decision tree. When species number m is known and unknown, we assign all the reads to the obtained m high confident genus. The flowchart of this method is shown in Figure 3.

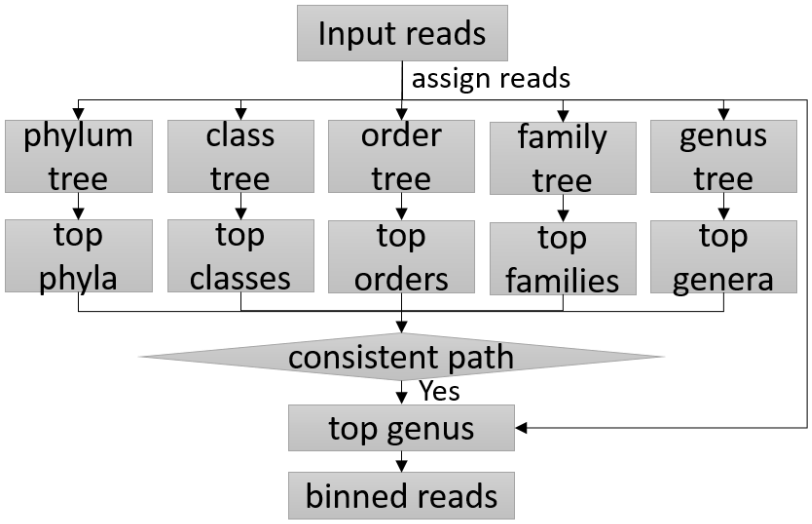

Figure 3 the flowchart of reads binning using taxonomical decision tree

3.2.4.1 Get the genus candidates

Given five decision trees which were built based on the 8-th order Markov Chain, we can assign reads to each of them. After we assigned reads to each of five tree, we will have the number of assigned reads for each node in each tree.

For individual tree, we only consider the leaves, each of which represent one taxon. We can sort the taxa based on three values (#assigned reads), (# assigned reads)*precision, and values calculated based on the binomial distribution. We calculate the values based on the binomial distribution in the following way: For each assigned taxon, we have a path from root to leaf; through this path we can calculate the probability that among the predicted real reads, how many were correctly assigned to this taxon: Pr=precision1*precision2*.... Then for each taxon, we can calculate the value 1-pbinom(n-1,N,Pr), in which, n (=# assigned reads * precision) is the number

of predicted real reads in one taxon, N is the total number of reads. We only keep the taxa that have value<1e-05. In this way, we can discard many taxa that reads were not reliably assigned to.

For species that were known to the tree, the real taxon in the assigned groups which is based on the values of binomial distribution will in general have higher rank. For example, in the genus tree, Figure 4 showed the real genus rank for the three values when we only consider the species with real genus rank (based on(# assigned reads)) larger than 10 in our simulated datasets.

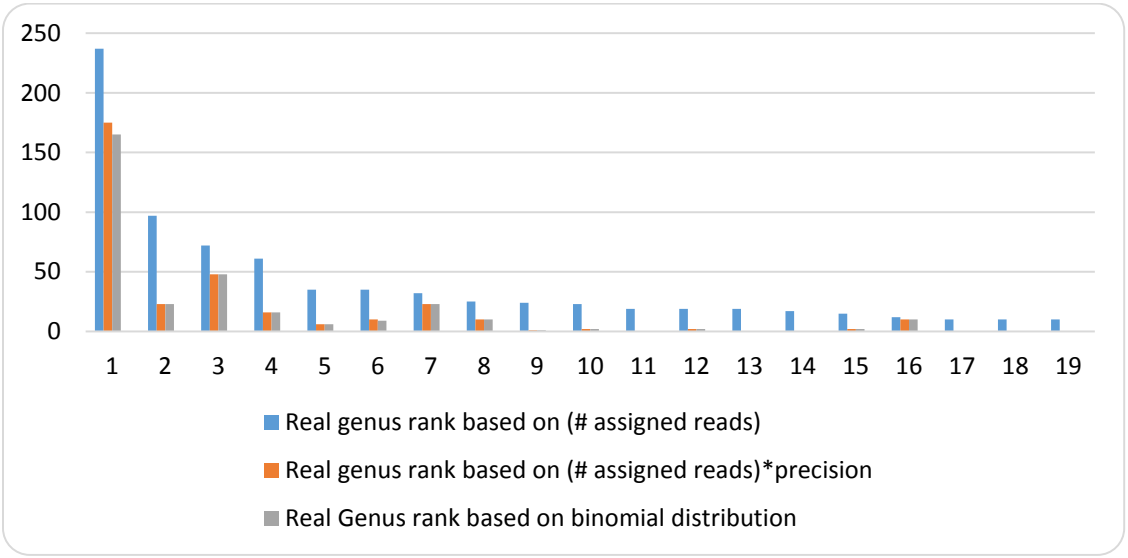

Figure 4 Genus ranks based on three values

From phylum to genus tree, with known species classification information, we can iteratively get the consistent path such that each taxon in the path satisfies the value of binomial distribution 1-pbinom(n-1,N,Pr) <1e-05. Consistent paths mean that the taxon in higher level contain the taxon in lower level. In the genus tree, we will finally get all possible high confident genus. To reduce the number of high confident genus further, these high confidence genus will be sorted by the value (# assigned reads)*precision, and we only keep the top 20 Genus. As in some

cases the real taxa will be discarded due to the inaccurate calculation of Pr, we also combine the Genus from top K(K=10) Genus based on the (#assigned reads).

At last, we combine the high confident Genus and top K Genus together and get the Genus candidates. In all of our datasets, we need about 20 Genus candidates averagely.

3.2.4.2 Predict the number of species and group the reads

After we have the genus candidates, we re-assign all the reads to these genus candidates.

When the species number m is known, we rank these genus candidates by the number of assigned reads. We obtained the top m Genus (m is the known species number), and the assigned reads from other genera will be re-assigned to these top m genus. In this way, all reads will be assigned to the top m genus.

When the species number m is unknown. Iteratively, we obtained the top Genus from the Genus candidates that satisfy the value 1-pbinom(n-1,N,Pr) <1e-05, and also each of these Genus contains more than 5% of totally reads; the assigned reads from other Genus candidates will be re-assigned to these top Genus until the predicted number of species was not changed. In the application of binomial distribution, N is the total number of reads, n is the number of assigned reads of one node, the probability Pr is the probability that one random reads be accurately assigned to one node. Here we used the F1 score which is a weighted average of both precision and sensitivity. Because we can't calculate confidences so precisely, we also require that these genera contain more than 5% of totally reads to compensate such errors.

3.2.5 Simulated and Experimental Datasets:

For each species, the illumina paired-end reads were simulated using MetaSim software with read length=75bp and empirical error model. The genome coverage was set to be 3 or 1.

We generated datasets with low abundance ratio and low abundance. Each of our simulated datasets contain reads from three different species which were in the same phylum, class, order or family, and the genome coverage ratio for each dataset is 1:1:1. In each level, we have five datasets, and all species were known to the three. Using the same species that have genome coverage 3, we also generated datasets, each of which contain extremely low abundance species, the genome coverage for each species is smaller than 1(=0.5).

We generated five datasets which contain reads from species that were unknown to all the tree. Each dataset contains three randomly selected species.

For experimental datasets, we randomly selected two datasets from the Human Microbiome Project (http://hmpdacc.org/HMSCP/): SRS017080 and SRS013705. These datasets contained mapped paired-end reads. For each dataset, we randomly selected three species. The other simplified real dataset was the human gut dataset from 15 randomly selected samples and downloaded from ftp://public.genomics.org.cn/BGI/gutmeta/High_quality_reads/. There were 257,158,754 paired-end reads in this dataset, each of which was 75 base pairs long. These reads were mapped to the following three species using the software SOAP: Bacteroides uniformis, Alistipes putredinis, and Ruminococcusbromii L2-63. There were 4,684,098 reads mapped to the three genomes and used to test the methods.

3.2.6 Comparisons with AbundanceBin, MetaCluster 5.0, MEGAN5 and Kraken

We ran our program by using known and unknown species numbers.

With the known species number m as input, we ran AbundanceBin using the command "./abundancebin -input reads_file -bin_num m".

We ran MetaCluster 5.0 by using the command "./ MetaCluster5_2 reads_file.2 --Species m''. Because the genome coverage (=3) were very low for all simulated species, we only run the second command of MetaCluster 5.0, which is used to process low-abundance species.

Besides doing comparisons with taxonomy-independent methods, we also compared with two taxonomy-dependent methods by using the same reference genomes. To compare with these two methods fairly, we used the same reference genomes as ours. MEGAN5 is an alignment-based method. We ran MEGAN5, by firstly mapped the reads to the 2592 reference genomes using BLASTN, and then ran it by inputting mapped reads. The other method Kraken, is a composition-based method. We firstly generated our own reference database using those 2592 genomes. Because of the limited memory computing environment, we used the same command as they generated their "MiniKraken" database. To increase Kraken's accuracy, we concatenated the paired-end reads together with a single "N" between the sequences as suggested by Kraken. With our own databases and concatenated reads, we run Kraken to get the reads assignment.

3.3 Results

3.3.1 8-th order Markov Chain is Sufficient to Bin Reads and Represents the Corresponding Genome Sequences

One genome sequence can be represented by an N-th order Markov chain. With thousands of known genome sequences, we would like to know which order of Markov chain can better represent a genome sequence. We randomly generated 10 datasets, each has three species which were from the same Genus. For each of the 10 datasets, the paired-end reads with coverage 3 from the three species were assigned to the three Markov Chains trained from the three species in the dataset. The read assignment accuracy is the number of correctly assigned reads divided by the total number of reads from the three species.
From the comparisons, we know that 8-th order Markov Chain will have relatively high assignment accuracy and it will result in highest increment (12.11%) in assignment accuracy compared with other Markov Chains. Higher order Markov Chain will not only have not much increment on the assignment accuracy compared with 8-th order Markov Chain, but also will generate huge number of transition probabilities (4^10 or more) which will make the calculation time-consuming, especially in our decision tree.

3.3.2 Our Method Can Divide the Reads Well When Species Were Known and is Comparable or Better than Other Four Methods

We generated datasets with low genome coverage ratio and low abundance. Each of our simulated datasets contains reads from three different species which were in the same Phylum, Class, Order or Family, and the genome coverage ratio for each dataset is 1:1:1, and the genome

coverage for each species is 3. In each level, we have five datasets, and all species were known to the tree.

We have 608 genus, instead of assigning reads to so many Genus directly, we assigned reads to five decision tree (phylum tree, class tree, order tree, family tree and genus tree). By using certain strategy, we obtained limited number of genus from all 608 genus. The number of Genus candidates for each dataset were shown below:

23, 23, 24, 24, 23, 22, 28, 25, 20, 22, 24, 22, 25, 21, 21, 23, 27, 25, 21, 24

23 out of 25 datasets' candidates contain the real species. So in general we need about 23 genus candidates. From these genus candidates, we selected certain number of high confident genus when the species number m is known or unknown, then all reads were assigned to m genus.

The results of our method when m is known and unknown were shown in Table 5. Our method works well when species were known. Besides dataset '2_2' and '3_4', the Genus candidates of other datasets contain real species. But from the results of these two datasets, we can see that although the genus candidates did not contain real species (one real species was missing), the accuracy of '2_2' was also high. When m is unknown, our program can predict the correct number of species for most of the datasets. From the accuracy of the reads assignments, we can see that reads from higher level tend to be divided better than reads from lower level, that's because Markov Chains from higher level were more distinguishable than Markov Chains from lower level.

For AbundanceBin, when input m=3, and when we consider that the species with the largest assigned reads in each bin can represent that bin, the three output bins of AbundanceBin can only represent less than three species for all datasets, while the grouped reads from our method

and MetaCluster can usually represent exactly three species for each dataset. So AbundanceBin cannot work well on these datasets as the genome coverage ratio were 1:1:1 for all these datasets.

Both MEGAN5 and Kraken are taxonomic dependent method. MEGAN5 is an alignment-based method, Kraken is a composition-based method. Both of these methods used a lowest common ancestor algorithm to assign reads, one applies on the mapped reads, the other applies on the exact alignment of k-mers. As the reads will usually be mapped to more than three species, when we calculated the assignment accuracy, we only consider the groups that contain more than 5% of totally reads. Both of these two methods rely on the reference databases, thus it has high accuracy when the reference sequences were known. Our method is comparable with both MEGAN5 and Kraken regarding that they resulted in more than three clusters for more datasets and their methods rely on the sequence alignment, which tends to have high accuracy.

Table 5 Comparisons of accuracy on 25 simulated datasets (decision tree method)

Notes	dataset	Accuracy (ours when m is known)	Accuracy (ours when m is unknown) [m]	MetaCluster	Abundancebin	MEGAN [m]	Kraken [m]
species from same Phylum	1_1	97.63%	97.63%[3]	55.26%	48.31%	87.77%[6]	90.84%[5]
	1_2	96.99%	96.99%[3]	56.60%	49.36%	95.87%[3]	95.87%[3]
	1_3	97.72%	97.72%[3]	61.30%	49.57%	95.92%[4]	96.04%[3]
	1_4	97.19%	97.19%[3]	58.80%	46.87%	98.76%[3]	96.88%[4]
	1_5	96.07%	96.07%[3]	56.65%	44.81%	98.93%[3]	96.84%[3]
species from same Class	2_1	97.32%	97.32%[3]	59.05%	38.79%	93.82%[5]	93.26%[4]
	2_2	96.22%	96.15%[4]	57.30%	41.06%	98.71%[3]	96.75%[3]
	2_3	93.65%	93.65%[3]	52.81%	42.28%	96.72%[3]	91.53%[4]
	2_4	92.91%	92.91%[3]	58.11%	40.43%	99.16%[3]	97.02%[3]
	2_5	91.76%	91.76%[3]	63.63%	37.70%	98.73%[3]	97.07%[3]
species from same Order	3_1	92.98%	92.98%[3]	68.03%	40.18%	96.38%[3]	94.70%[3]
	3_2	92.30%	91.75%[4]	56.16%	55.43%	99.12%[4]	91.69%[4]
	3_3	95.22%	95.22%[3]	61.97%	35.35%	96.04%[3]	95.56%[3]
	3_4	76.21%	75.65%[4]	60.31%	57.79%	97.43%[3]	96.86%[3]
	3_5	89.42%	89.42%[3]	57.90%	47.01%	96.83%[4]	94.68%[4]
species from same family	4_1	95.87%	95.87%[3]	59.24%	38.81%	99.23%[3]	96.78%[3]
	4_2	90.15%	90.15%[3]	59.68%	38.17%	99.22%[3]	96.86%[3]
	4_3	77.27%	77.27%[3]	50.46%	39.57%	94.88%[3]	92.36%[4]
	4_4	92.68%	92.68%[3]	58.61%	42.47%	98.85%[3]	97.00%[3]
	4_5	92.29%	92.29%[3]	60.73%	52.16%	94.80%[3]	96.96%[3]

The simulated datasets tested above all have abundance ratio 1:1:1, and genome coverage 3, our method has good performances in such low abundance ratio datasets. We also generated datasets with extremely low abundance species for the same species used above, each dataset contain three species and the genome coverage of each species is less than 1, only 0.5. The results showed that our method is comparable with MEGAN5, and better than other three methods. MetaCluter have no results for these datasets. AbundanceBin usually group most of the reads into only one bin even we set the input number of bins to be 3. Both MEGAN and Kraken have similar results as above, which means that taxonomic-dependent method will work for datasets with

extremely low abundance species. Our method also has similar results as above. Thus our method can generally group reads without the needs to consider their low abundances or low abundance ratio.

3.3.3 Our Method Works Better than Other Four Methods on Real Datasets

We did experimental on the two HMP real datasets SRS017080 and SRS013705 and one human gut datasets. In each of the dataset, one to two species were known to the tree.

The reads assignment accuracy of our method and MetaCluster, AbundanceBin when m is known were in Table 6. Our method has higher accuracy than other two methods, especially when two of three species were known to the tree. When two of three species were unknown to the tree, our method can also divide the reads into three groups which contain three main species, while grouped reads from other two methods sometimes can't represent three main species. And because of the low genome coverage and low abundance ratio, other methods sometimes can't work.

MEGAN5 has worse results as many reads from unknown species were not mapped to the reference genomes. The same is true of Kraken, it also has worse results.

Table 6 Comparisons of accuracy on real datasets (decision tree method)

	Accuracy (ours when m is known)	Accuracy (ours when m is unknown) [predicted m]	MetaCluster (m is known)	Abundancebin (m is known)	MEGAN (alignment-based method) [m]	Kraken (composition-based method) [m]
HMP: SRS017080	92.40 %	92.40%[3]	74.39% (mainly contain two species)	75.84% (mainly contain two species)	56.19% [2]	44.08% [2]
HMP: SRS013705	82.04 %	73.52%[3]	32.28% (mainly contain two species)	71.38%	28.07% [2]	15.33% [1]
Human gut: baalrub	76.03 %	76.25%[5]	71.77%	69.68% (mainly contain two species)	14.72% [1]	11.38% [1]

3.3.4 Our Method Have Better Performance than Other Four Methods When All Species were Unknown

We generated five datasets which contain reads from species that were unknown to the tree. Each dataset contain three randomly selected species. Also the genome coverage is 3, and the genome coverage ratio is 1:1:1 for each dataset.

When all the species were unknown to the tree, it has better performances than MetaCluster and AbundanceBin as shown in Table 7. The grouped reads of AbundanceBin also mainly contain less than three species for all datasets.

Both MEGAN5 and Karken can hardly work for all these datasets as most of the reads were not mapped to the reference genomes. When we calculated the assignment accuracy, we only consider the groups that contain more than 5% of totally reads.

Table 7 Comparisons of accuracy on datasets with unknown species (decision tree method)

dataset	Accuracy (ours when m is known)	Accuracy (ours when m is unknown) [m]	MetaCluster (m is known)	Abundancebin (m is known)	MEGAN (alignment-based method) [m]	Kraken (composition-based method) [m]
1	67.27%	69.03%[6]	63.34%	39.77%	0.00% [0]	0.00% [0]
2	70.45%	70.45%[3]	62.69%	39.68%	0.00% [0]	0.00% [0]
3	72.12%	75.50%[6]	59.70%	36.57%	0.00% [0]	0.00% [0]
4	68.14%	71.13%[6]	57.56%	35.02%	0.00% [0]	0.00% [0]
5	67.17%	81.70%[6]	57.76%	37.34%	5.97% [1]	0.00% [0]

3.4 Discussion and Conclusions

We developed a novel taxonomy-dependent method. This method obtains the potential taxa through five taxonomical decision tree. We showed that this method can usually find the real genus for most of the datasets that contain known species, and have higher accuracy than other approaches for datasets that contain unknown species. Although this method showed better performance than other methods. It has many problems. Firstly, the structure of this method is complex, we need to do hierarchical clustering and build five taxonomical decision tree. Secondly, it will generate many errors when doing clustering, so assigning the reads through the tree will not

be that accurate, and the most assigned taxa may not be the real taxa. Thirdly, because of the errors when doing clustering, we need to set a little large cutoff to keep as many as possible taxa. In this way, one to two genera in the datasets cannot still be kept. Fourthly, the large structure of the tree need huge of memory so that we can only stick to 8-th order Markov chains, but such genome representations can result in less accuracy than higher order Markov chains. In order to overcome the above limitations, we proposed another taxonomy-dependent method called MBMC.

CHAPTER 4 MBMC: AN EFFECTIVE TAXONOMY-DEPENDENT APPROACH FOR BINNING METAGENOMIC READS

4.1 Background

Binning reads is one of the most crucial steps in metagenomic data analyses [1]. A typical metagenomic dataset consists of millions or billions of short sequenced DNA segments called reads [73]. These reads usually originate from different microbial species and are mixed together during sequencing. Binning reads is the process of grouping reads from individual species or operational taxonomical units (OTUs) together [8] and is therefore critical for the understanding of the composition and functions of microbes in environmental samples [11, 14, 74].

Many computational methods have been developed to bin reads in metagenomic projects. These methods can be broadly classified into two categories. One category is the taxonomy-dependent methods [17-22, 26, 27, 43, 53, 75], which compare reads with sequences in public databases to group reads and determine which known species are present. The other category is the taxonomy-independent methods [25, 28-30, 71], which employ the difference of GC content, k-mer frequencies, etc., of different microbes in the same samples to bin reads. For instance, AbundanceBin [29] and MBBC [71] utilize the frequency difference of k-mers from different microbes in environmental samples to bin reads, which have been shown to successfully separate reads from species with very different abundance.

Despite the existence of dozens of methods for read binning, it is still challenging to bin reads from metagenomic projects accurately and efficiently. The difficulty stems from the fact that the majority of microbial species are still not sequenced, therefore making read binning from these

species problematic by the developed taxonomy-dependent approaches [10, 33]. The difficulty also commonly arises when the genome coverage of different microbial species is similar (<2 fold difference), which renders it unlikely to separate reads from species with similar abundance by taxonomy-independent methods such as AbundanceBin [29] and MetaCluster [30]. As a result of these limitations, more powerful read binning methods are required.

In this study, we developed a novel taxonomy-dependent approach called MBMC. Different from all existing methods, MBMC bins reads by measuring the similarity of reads to the trained 9-th order Markov chains for different taxa, instead of directly comparing reads with known genomic sequences. We showed that MBMC reliably determined the species number and binned reads with an average accuracy of more than 91% when tested on 10 simulated datasets with similar species abundance. Additionally, we demonstrated that MBMC reliably binned reads for both known and unknown species when tested on 10 additional simulated datasets and 4 experimental datasets. Compared with four existing approaches, MBMC demonstrated comparable or better performance than existing approaches on reads from known species, and far superior performance when dealing with reads from "unknown" species.

4.2 Materials and Methods

4.2.1 Known Species Used and Their Representation

We used the same taxa mentioned in Section 3.2.1. But here we chose the 9-th Order Markov chain to model the microbial genomes in MBMC after comparing with Markov chains of other orders.

4.2.2 Simulated and Experimental Datasets

To compare the different order of Markov chains, we generated 10 random datasets. In each dataset, paired-end reads were simulated with MetaSim [64] to cover the genomes of three species from the same genus three times. In order to determine whether the 8-th order or the 9-th order Markov chains should be used, we generated 25 groups of species, each of which contained three species. We then generated 50 simulated datasets using the 25 groups of species, with the genome coverage as either 3 or 0.5.

To compare MBMC with existing methods, we simulated 10 more complicated datasets. At each taxonomical level (phylum, class, order, family or genus), two datasets were generated. The first one contained five species from the same taxon with the corresponding genome coverage as 1, 1, 1, 2, and 2. The second one contained six species from the same taxon with the corresponding genome coverage as 2, 3, 4, 5, 6, and 7. Species from the same taxonomical level with similar coverage in these datasets made it challenging to bin reads by taxonomy-independent methods [29, 71]. For each species, Illumina paired-end reads were simulated using MetaSim [64] with the read length as 75 base pairs and with the empirical error model in MetaSim. The 75 base pairs long reads were used because many benchmark datasets contained reads of such a length, and it was much more challenging to bin shorter reads than to bin longer reads [30, 76]. We thus reported our study on datasets with 75 base pairs long reads in the following. It was also worth mentioning that we simulated datasets with longer reads as well, and in general, MBMC indeed worked even better on datasets with longer reads.

To study whether MBMC and other methods could work on datasets composed of reads from "unknown" species, we generated additional 10 simulated datasets. Each dataset consisted of reads from three species with the coverage for each genome as 3 or 0.5. The three species were randomly selected from the 181 of the 2773 completed sequenced microbial genomes that did not have complete taxonomical information and were not used to train MBMC.

For experimental datasets, we randomly selected two datasets from the Human Microbiome Project (HMP, http://hmpdacc.org/HMSCP/), SRS017080 and SRS013705. These two datasets contained mapped paired-end reads. We selected four species for the dataset SRS017080, only one of which was known to MBMC and the other three of which were unknown species. However, these unknown species in the dataset SRS017080 had known sequenced genomes from the same species but different strains. For the dataset SRS013705, we selected five species, one of which was known and the remaining four unknown. One of these four species had a known genome from a different strain. Moreover, we generated a HMP mock dataset that contained single-end reads from SRR172902 and SRR172903. This mock dataset consisted of reads from 22 microbial genomes. All reads were mapped to the 22 genomes by SOAPdenovo2 [77] with the default parameters. To be consistent with our species abundance cutoff in the default version of MBMC, 5%, we selected all top abundant species that contained more than 5% of total reads in this HMP mock dataset. There were six such species. Reads from these six species accounted for 71.62% of total mapped reads. In addition, we used a human gut dataset from 15 randomly selected samples from ftp://public.genomics.org.cn/BGI/gutmeta/High_quality_reads/ [78]. There were 257,158,754 paired-end reads in this dataset, each of which was 75 base pairs

long. These reads were mapped to the following three species using the software SOAPdenovo2 [77], *Bacteroides uniformis*, *Alistipes putredinis*, and *Ruminococcusbromii L2-63*, because they were the most abundant species and/or had more complete genome sequences in the gut datasets [71, 78]. In total, 4,684,098 reads were mapped to the three genomes and used to test MBMC and other methods.

4.2.3 MBMC: A Novel Taxonomy-Dependent Approach to Bin Metagenomic Reads

We developed MBMC, a novel taxonomy-dependent approach, to bin metagenomic reads. For a given dataset, MBMC determines and selects potential taxa with the ordinary least squares (OLS) method, and then assigns reads to the selected taxa by the relative entropy measurement. The details are in the following.

To determine potential taxa, MBMC models all input reads by a mixture of 9-th order Markov chains. That is, the frequency of 9-mers and 10-mers in all input reads and their reverse complement reads is counted to calculate the stationary and the transition probabilities of the Markov chain mixture. Correspondingly, MBMC models each taxon at every taxonomical level by a 9-th order Markov chain. MBMC then applies the following OLS to identify potential taxa:

$$y = \mathbf{X}\beta + \epsilon$$

where, $y = \begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix}$, $X = \begin{pmatrix} X_1^T \\ X_2^T \\ \dots \\ X_n^T \end{pmatrix} = \begin{pmatrix} x_{11} \dots x_{1p} \\ x_{21} \dots x_{2p} \\ \dots \\ x_{n1} \dots x_{np} \end{pmatrix}$, $\beta = \begin{pmatrix} \beta_1 \\ \beta_2 \\ \dots \\ \beta_p \end{pmatrix}$, $\epsilon = \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \dots \\ \epsilon_p \end{pmatrix}$

In the above formula, n=4^{10}; y and each column of X denote the transition probabilities of the input reads and each of the p taxa at current taxonomical level under consideration, respectively;

β is the unknown parameter that approximates the relative abundance of reads from a taxon. The unknown parameter β can be estimated by OLS as

$$\hat{\beta} = (X^T X)^{-1} X^T y$$

MBMC only keeps taxa whose relative abundance β is larger than a pre-specified cutoff, say 0.01 or 0.05. Since there are much more taxa at lower levels such as at the species level than those at higher levels such as at the phylum level, to speed up the prediction process, MBMC iteratively chooses potential taxa from high levels to low levels (Figure 5). In brief, MBMC first infers potential phyla. There are 32 different phyla in total and X is thus initially a 4^{10} by 32 matrix. By the above OLS, MBMC keeps potential phyla whose corresponding relative abundance β_i>=0.01. In addition to these K_1 phyla with β_i>=0.01, MBMC also keeps the minimum number of taxa such that their cumulative sum of the relative abundance will be larger than 0.5. That is, MBMC selects the smallest number of taxa, say K_2, so that $\beta_1 + \beta_2 + ... + \beta_{K_2} \geq 0.5$. This is to deal with unknown species that may exist in the datasets. When certain reads are from an unknown species, the estimated relative abundances of this species will be small (e.g. <0.01), because reads from an unknown species will often be assigned to several closest "neighbours" of known taxa. Then, totally K potential phyla were kept, in which $K = union(K_1, K_2)$. Next, MBMC infers potential classes. Note that although there are 56 classes in our training datasets,

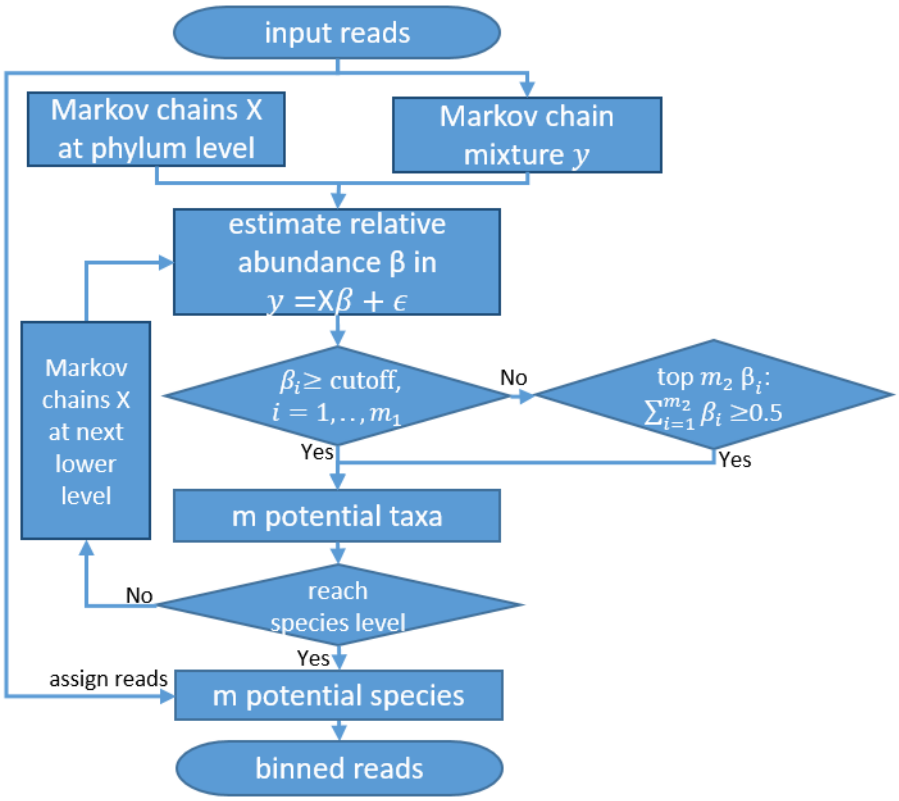

Figure 5 Flowchart of MBMC.

The cutoff to select potential taxa was defined as 0.01 at the phylum to genus levels and 0.05 at the species level.

X in the above OLS only considers classes that belong to the selected potential phyla and thus p may be much smaller than 56. Next, MBMC similarly predicts potential orders, families and genera, respectively. Finally, MBMC similarly predicts potential species by requiring β_i>=0.05 and $\beta_1 + \beta_2 + ... + \beta_{K_2} \geq 0.5$ to obtain potential species. The use of such a stringent requirement enables more accurate predictions of species in our studies.

With the K potential species, MBMC assigns reads to the potential species, by calculating the similarity score of a read to the Markov chains that represent the selected potential species. For

a given potential species, the similarity score of a single-end read $a_1a_2 \ldots a_n$ is the corresponding stationary probability times the corresponding transition probabilities. That is, for a given species, its score is equal to

$$S(a_1a_2 \ldots a_9) * T(a_{10}|a_1a_2 \ldots a_9) * T(a_{11}|a_2a_3 \ldots a_{10}) * \ldots * T(a_n|a_{n-9}a_{n-8} \ldots a_{n-1})$$

in which, a_i, i=1,..,n is the nucleotide at the i-th position of this read, n is the length of the read, S and T are the transition probability and stationary probability of the corresponding Markov chian for this potential species, respectively. For a paired-end read, its score will be the larger of the following two scores, (a+b')/2 and (a'+b)/2, where a, a', b and b' are score of one end of this read, score of the reverse complement of this end, score of the other end of the read, and score of the reverse complement of the other end, respectively.

4.2.4 Comparisons with Other Methods

We compared MBMC with two taxonomy-independent methods, AbundanceBin [29] and MetaCluster 5.1 [30]. We ran MBMC without the information of the number of species present. We ran AbundanceBin and MetaCluster with the actual species number m as input, since they hardly predicted the correct species number. We ran AbundanceBin using the command "./abundancebin –inpu t reads_file -bin_num m". We ran MetaCluster by using the command "./MetaCluster5_2 reads_file --Species m''. Because the genome coverage was low for all simulated species, we only ran the second command of MetaCluster 5.1, which was used to process low-abundance species.

We also compared MBMC with two taxonomy-dependent methods, MEGAN5 [17] and Kraken [22]. It must be noted that we used the same 2592 genomes to train MEGAN5 and Kraken.

MEGAN5 is an alignment-based method. We ran MEGAN5, by firstly mapping reads to the 2592 reference genomes using BLASTN, and then ran it by inputting mapped reads to its GUI mode program. Kraken is a composition-based method. We firstly generated a reference database using the 2592 genomes. Because of the limited memory-computing environment, we used the same command as they generated their "MiniKraken" database. To increase Kraken's accuracy, we concatenated paired-end reads together with a single "N" between the sequences, as suggested by Kraken. With the built reference database and concatenated reads, we ran Kraken by the command "./kraken --preload --db my_own_db reads_file >>output_file".

4.3 Results

4.3.1 The 9-th Markov Chain Models Are Effective in Representing Microbial Genomes.

We represented each completed microbial genome by an N-th order Markov chain, where N was from 5 to 10. Intuitively, the higher order a Markov chain has, the better it represents a given microbial genome. However, a Markov chain with a higher order has at least 4 times more parameters, which demands higher memory storage and more computation time. Therefore, it is necessary to investigate which order gives more accurate read assignment with reasonable memory usage.

To see which order of Markov chains was better, we studied read assignment in 10 random datasets. In each dataset, paired-end reads were simulated with MetaSim [64] to cover the genomes of three species from the same genus three times. The genomes of the corresponding three species were represented by N-th order Markov chains, where N was from 5 to 10. For a given Markov chain order, a read was assigned to one of the three Markov chains for which this read had the

largest similarity score. The read assignment accuracy in a dataset was defined as the number of correctly assigned reads divided by the total number of reads from the three species. Figure 6 showed the read assignment accuracy in each dataset when using different orders of Markov chains. From the 5^{th} to 10^{th} order Markov chains, the average accuracy increased. The largest increment, 19.05%, happened when the order was changed from 7 to 8. The second largest increment, 10.57%, occurred from the order 8 to the order 9. The increment from the 9-th order to the 10-th order was 2.78%, which together with the above observations suggested that the 8-th or the 9-th order Markov chains likely be the most effective Markov chains to represent a microbial genome.

Figure 6 Comparisons of read assignment accuracy using different orders of Markov Chains.

To compare the representation of the 8-th Markov chains and the 9-th order Markov chains, we tested them on 25 simulated datasets. Here although reads in each dataset were still from three randomly chosen species, every read was scored with K instead of 3 Markov chains to calculate the similarity scores of the read to the Markov chains, and to determine the assigned species. Here

K was the number of potential species predicted by MBMC. The assignment accuracy was correspondingly defined as the percentage of reads that were correctly assigned to the correct species. From the accuracy comparison of the representation of the 8-th and the 9-th order Markov chains, it was evident that the accuracy was improved by 2.9% for the 9-th order Markov Chains. For one of the datasets, dataset 3_5, because of the inaccuracy representation of the genome sequences by the 8-th order Markov Chains, the obtained potential species even did not include the real ones. The assignment accuracy was only 71.40% for the 8-th order chain on this dataset, while the 9-th order Markov Chains achieved more than 89% accuracy on every dataset. We therefore chose the 9-th order Markov chain representation in MBMC in the following analyses.

It was worth mentioning that the 9-th order Markov chains representing species from the same taxa were in general more similar to each other than those representing species from different taxa. For instance, the 9-th order Markov chains representing species from the same genus, *Spiroplasma*, had an average distance of 212147, while the 9-th order Markov chains representing species from *Spiroplasma* and from any other genus (such as Genus *Aquifex*) had an average much larger distance (such as 246549). The distance of two Markov chains was calculated as the sum of the absolute difference of the corresponding entries in the two transition matrices. The more different the taxonomical information of two species was, in general, the larger distance the two corresponding Markov chains had. This implied that the Markov chain representation of an un-sequenced species may be approximated by the Markov chain representation of a sequenced species that had similar high-level taxa as this un-sequenced species. In other words, one may

group reads from un-sequenced species by using the Markov representation of sequenced species, which was exactly what we discovered in Sections 4.3.3 and 4.3.4.

4.3.2 MBMC Reliably Predicted the Species Number and Accurately Grouped Reads in Simulated Datasets

With the 9-th order Markov chain representation of microbial genomes, we developed a novel taxonomy-dependent binning method called MBMC (Material and Methods). To show whether MBMC reliably binned reads on more complicated datasets, we simulated 10 datasets with MetaSim [64] and tested MBMC on these datasets. At each taxonomical level (phylum, class, order, family or genus), two datasets were simulated. The first dataset contained five species from the same taxon with the corresponding genome coverage as 1, 1, 1, 2, and 2, respectively; and the second one contained six species from the same taxon with the corresponding genome coverage as 2, 3, 4, 5, 6, and 7, respectively. Species with similar low coverage from the same taxon made it challenging to bin reads [29, 71].

MBMC performed well on these datasets (Table 8). In all 8 out of 10 datasets, MBMC correctly predicted the actual species number. The binning accuracy in a dataset, which was defined as the percentage of correctly binned reads divided by the total number of reads, varied from 69.39% to 99.04%, with an average accuracy as 91.71%. It was also evident that the binning accuracy was relatively higher at high taxonomical levels and lower at low taxonomical levels. For instance, the average binning accuracy at the phylum level was 96.61% while that was 81.38% at the genus level (Table 8).

Table 8 Comparisons of accuracy on 10 simulated datasets (MBMC)

dataset[species #]	MBMC [m]	MetaCluster	Abundancebin	MEGAN5 [m]	Kraken [m]
1_1[5]	96.96%[5]	42.56%	na	94.19%[4]	92.00%[4]
1_2[6]	96.25%[6]	91.36%	na	97.02%[6]	90.57%[5]
2_1[5]	99.04%[5]	42.76%	34.48%	88.37%[5]	92.60%[5]
2_2[6]	95.26%[6]	89.38%	33.50%	97.71%[6]	89.16%[6]
3_1[5]	89.11%[5]	36.19%	29.55%	99.01%[5]	94.59%[5]
3_2[6]	94.37%[7]	87.20%	38.50%	93.25%[7]	93.32%[6]
4_1[5]	97.36%[5]	41.57%	28.07%	97.39%[5]	96.27%[5]
4_2[6]	86.01%[5]	90.42%	28.06%	98.82%[6]	96.43%[6]
5_1[5]	93.36%[5]	40.78%	28.28%	82.02%[6]	91.68%[5]
5_2[6]	69.39%[6]	51.02%	na	89.55%[6]	90.50%[7]

At each taxonomical level (phylum, class, order, family and genus), two datasets were generated. The first one contained five species from the same taxon with the corresponding genome coverage as 1, 1, 1, 2, and 2. The second one contained six species from the same taxon with the corresponding genome coverage as 2, 3, 4, 5, 6, and 7.

We noticed that for the last dataset, the dataset 5_2, although MBMC correctly predicted the species number, it had a low accuracy. This was caused by the similarity among the six Markov chains corresponding to these species. For instance, the smallest distance between two Markov chains in the dataset 5_2 was 102014, which was much smaller than that in all other datasets (111182 in the dataset 3_2). Recall that the distance of two Markov chains was calculated as the sum of the absolute difference of the corresponding entries in the two transition matrices. It was worth pointing out that the average distance of two 9-th order Markov chains for two species from the same genus was 160615, which was much larger than the smallest distance of Markov chains in the dataset 5_2.

We also applied four existing methods to the same 10 simulated datasets (Material and Methods). The two taxonomy-independent methods, AbundanceBin and MetaCluster, could not predict the correct species number in any dataset. Therefore, we specified the actual species number in each dataset when we ran them. Even with the actual species number as input, on average, the binning accuracy of AbundanceBin and MetaCluster was 31.49% and 61.32%, respectively (Table 8). We observed that at every taxonomical level, the accuracy of MetaCluster was much higher in the second dataset than that in the first one. This may be because the coverage in the first dataset was smaller than that in the second dataset, and MetaCluster could not work well with species of low coverage. This also implied that MBMC worked well for datasets with low-abundance species. The binning accuracy of the two taxonomy-dependent methods, MEGAN5 and Kraken, was high because the species in these datasets were known to the two methods (Table 8). In addition, as reads were usually mapped to many different species by the two taxonomy-dependent methods, when we calculated their binning accuracy, we considered only groups containing more than 5% of total reads and assumed groups with most reads from a species as the correctly predicted groups for that species. Even with such a treatment, both MEGAN5 and Kraken incorrectly predicted the species number in three datasets. Overall, MBMC had a slightly lower average accuracy than MEGNA5 and Kraken at lower taxonomical levels (order, family, genus), but higher accuracy in higher taxonomical levels (phylum, class), suggesting that the Markov chains for species from different higher level taxa were more different than those from lower level taxa.

4.3.3 MBMC Worked Well on Datasets with Unknown Species

In contrast to the existing taxonomy-dependent approaches, MBMC compares reads with the Markov representation of the sequenced microbial genomes, instead of the microbial genome sequences themselves. We hypothesize that this unique aspect may enable MBMC to reliably bin reads from unknown species. The rationale was that an unknown species may be from the same phylum, class, order, family or genus as certain sequenced species, whose Markov representation may thus aid the accurate binning of reads from this unknown species. To test this hypothesis, we applied MBMC to 10 additional datasets. Each dataset was composed of reads from three randomly selected "unknown" species that were not used to train MBMC. The genome coverage was set as 3 or 0.5 for each species in each dataset.

The binning accuracy of MBMC together with other four existing methods was summarized in Table 9. MBMC had an average accuracy of 70.30% on these datasets, which was at least 10.13% higher than any of the four methods. Among the four methods, the two taxonomy-independent methods had much higher accuracy than the two taxonomy-dependent methods. We again input the actual species number for the two taxonomy-independent methods to improve their accuracy. As to the two taxonomy-dependent methods, they were barely able to bin any read in any dataset (Table 9), because reads from unknown species unlikely satisfied the required high similarity to the genome sequences used to train the two methods. For instance, Kraken did not group more than 5% of the total reads to any bin, although three such groups of reads existed in each of the ten datasets. It was also evident that although MetaCluster performed better than other

three methods on datasets with high coverage (coverage=3), it could not work on any dataset with coverage as 0.5.

Table 9 Comparisons on datasets with unknown species.(MBMC)

dataset[species #]	MBMC [m]	MetaCluster	Abundancebin	MEGAN5 [m]	Kraken [m]
0_1	65.39%[12]	63.60%	39.71%	0.00%[0]	0.00%[0]
0_1*	65.47%[13]	na	38.76%	0.00%[0]	0.00%[0]
0_2	68.55%[11]	60.16%	39.66%	0.00%[0]	0.00%[0]
0_2*	68.05%[12]	na	38.68%	0.00%[0]	0.00%[0]
0_3	79.15%[13]	61.56%	36.59%	0.00%[0]	0.00%[0]
0_3*	77.23%[12]	na	36.25%	0.00%[0]	0.00%[0]
0_4	63.87%[15]	58.38%	35.00%	0.00%[0]	0.00%[0]
0_4*	64.09%[15]	na	35.04%	0.00%[0]	0.00%[0]
0_5	75.02%[15]	57.15%	37.34%	0.00%[0]	0.00%[0]
0_5*	76.22%[14]	na	37.51%	0.00%[0]	0.00%[0]

Each dataset contained three unknown species with the corresponding genome coverage as either 3, 3, 3 or 0.5, 0.5 and 0.5 (datasets marked with *). Here m represented the number of the predicted species, and "na" meant no meaningful output.

We also noticed that MBMC did not accurately predict the actual species number in these datasets (Table 9). This was because multiple genome sequences used to train MBMC may be similar to the genome sequences of an unknown species. Reads from the unknown species may thus be divided into several groups, each of which may be more similar to one of the known genomes and predicted as a species. We scrutinized our prediction in these ten datasets and confirmed that this was the case. For instance, in the ninth dataset, five predicted species, *Bacillus amyloliquefaciens, Bacillus licheniformis, Geobacillus_Y412MC61, Solibacillus silvestris* and *Paenibacillus Y412MC10*, shared the same order with the actual "unknown" species

Exiguobacterium sibiricum. These five predicted species should be combined into one predicted species, as 70.91% of reads from them were from the actual species *Exiguobacterium sibiricum*, which was not used to train MBMC. In the same dataset, another four predicted species, *Sinorhizobium fredii, Agrobacterium H13 3, Agrobacterium fabrum* and *Rhizobium leguminosarum* shared the same family as the actual "unknown" species *Rhizobium sp. IRBG74*. Again, these four predicted species should be combined into one predicted species, as 92.13% of reads from them were from the actual species *Rhizobium sp IRBG74.*

A valid question was thus how to tell whether there existed unknown species in a dataset. We observed that the predicted abundance of each bin was relatively small in these datasets. For instance, in the fifth dataset, at the species level, the largest relative abundance was only 0.041. Moreover, as we mentioned above, we also observed that several genome sequences corresponding to these small bins were quite similar to each other. In practice, one may take the above two observations into account when interpreting the predictions by MBMC.

4.3.4 MBMC Performed Much Better than Other Methods on Experimental Datasets

To assess the practical application of MBMC, we studied the performance of MBMC on four experimental datasets: two HMP real datasets SRS017080 and SRS013705, one HMP mock dataset, and one human gut dataset. Two HMP real datasets and one human gut dataset were used because such datasets were widely used in assessing other metagenomics tools [30, 79-81].The HMP mock dataset was used because it was different from other three datasets and contained only single-end read. In each dataset, at least one species was known.

The read binning accuracy of MBMC together with that of other four existing methods was shown in Table 10. Overall, MBMC had a higher binning accuracy than other methods in three of the four datasets, where there existed unknown species. In these three datasets, MBMC in general binned reads into more groups, with several groups corresponding to the same unknown species. MEGAN5 and Kraken had relatively poor predictions when unknown species existed in the datasets, as many reads from the unknown species were not mapped to the reference genomes. For instance, in the dataset SRS013705, MBMC had more than 28.78% higher accuracy than MEGAN5 and Kraken. It was worth pointing out that MBMC had a similar accuracy as MEGAN5 and Kraken in the dataset SRS017080, because the three unknown species in this dataset had sequenced strains and MEGAN5 and Kraken were able to directly map a large number of reads to these sequenced known strains. However, even in this dataset, MBMC had a more than 8.07% higher accuracy than MEGAN5 and Kraken, supporting the advantage of comparing reads with the Markov representation of the genome sequences instead of the actual genome sequences themselves. As to the remaining dataset with only known species, the HMP mock dataset, MBMC correctly predicted five of the six species. Kraken had a low accuracy again due to the fact that this dataset contained single-end reads of 75 base pairs long, which were too short for Kraken to perform well. MetaCluster and AbundanceBin did not work out on this mock dataset, due to the low genome coverage and low abundance ratio of species. MEGAN had a higher accuracy in this mock dataset with single-end reads.

Table 10 again demonstrated the strength of MBMC from two aspects. One aspect is that although MBMC is a taxonomy-dependent method, it works well to bin reads from unknown

species. For instance, in the human gut dataset, which had two unknown species, MBMC achieved at least 11.39% higher accuracy than other methods. Second, MBMC works well when species abundance is low or the relative abundance among species is low. In both cases, the existing taxonomy-independent methods have difficulties in binning reads, as illustrated in the HMP mock dataset.

Table 10 Comparisons of accuracy on the real datasets. (MBMC)

dataset	MBMC [m]	MetaCluster	Abundancebin	MEGAN5 [m]	Kraken [m]
HMP: SRS017080 (2[3])	83.00%[2]	68.66%	52.01%	71.13%[2]	53.00%[3]
HMP: SRS013705 (2[3])	78.89%[5]	68.95%	50.36%	37.54%[3]	18.14%[2]
HMP mock (6[6])	77.57%[5]	na	na	81.79% [6]	62.52%[6]
Human gut (1[3])	86.88%[8]	71.77%	69.68%	14.72%[1]	11.38%[1]

m is the number of predicted species. The parentheses in each dataset ID includes the number of known species followed by the number of species in the dataset.

4.4 Discussion and Conclusions

We developed a novel taxonomy-dependent approach called MBMC to bin metagenomic reads. In contrast to existing taxonomy-dependent approaches, MBMC bins reads by measuring how well reads are modelled by the learned Markov chains from the sequenced microbial genomes, instead of comparing each read with each sequenced microbial genome directly. Such a unique characteristic enables MBMC to reliably group reads from both known and unknown species. Tested on 10 simulated datasets with only known species, we showed that MBMC had a more than

91% of accuracy on average, which was comparable with the performance of current popular taxonomy-dependent approaches on known species. Tested on 10 additional simulated datasets containing unknown species, we showed that MBMC had a more than 70% of accuracy on average, which was much better than current popular approaches. Tested on 4 experimental datasets, we demonstrated that MBMC outperformed the available methods, especially when there were unknown species.

It is worth mentioning that MBMC has a reasonable speed. It was at least 4 times faster than MEGAN (including BLAST time) and was faster than many taxonomy-dependent methods, such as NBC [24], PhymmBL[26]. The number of potential species in a dataset affects the speed of MBMC. Since MBMC likely predicts more species when there exist unknown species in a dataset, its speed will be slower in datasets containing unknown species than in similar datasets with only known species.

We used only 2592 completed microbial genomes to train MBMC. We could also include draft microbial genomes for the training of MBMC. In the future, with more microbes sequenced, their sequences could be used to train MBMC as well. A critical question remaining is whether the training with more genomes will make the binning accuracy of MBMC decrease instead of increasing. It is possible that the accuracy will decrease because more Markov chains from these genomes may be more similar to each other. To address this question, we randomly selected 1000 draft microbial genome sequences from ftp://ftp.ncbi.nlm.nih.gov/genomes/Bacteria/), 955 of which with all six level taxonomical information were combined with the default 2592 genomes to train MBMC. We then applied the trained MBMC to the simulated datasets mentioned above.

We found that the accuracy was almost the same with the accuracy when using 2592 genomes in MBMC. As mentioned in Kraken, there exist contaminant and adapter sequences in certain draft genomes [22]. Here training with draft genomes did not affect the accuracy of MBMC much. This suggests that MBMC is robust for contaminant sequences. It also implies that MBMC may become even better with more training genome sequences.

We implicitly employed the tree structure of taxa in MBMC. Alternatively, we could explicitly utilize the taxonomical trees to assign reads. For instance, we could build taxonomical trees at each taxonomical level, and assign reads to each tree and then combine trees at all levels to make final decisions about which species a read belongs to. Alternatively, one could use only the species tree to assign reads directly. We tested these two alternative approaches and found neither was as time-efficient as the top-down strategy used in MBMC.

MBMC performed better in binning reads from "unknown" species than existing approaches. We observed that MBMC tended to divide reads from an unknown species into multiple small bins. How to combine these small bins together to reconstruct a unique bin for an unknown species seems challenging and worth further investigation. In addition, although MBMC showed superior performance than existing approaches on simulated and real experimental datasets with unknown species, in practice, the low abundance of many unknown species in metagenomic projects is still challenging for existing methods including MBMC to figure out how to bin reads. It is thus important to be able to deal with species with even lower abundance than the current MBMC can deal with. In the next section, we proposed a reads binning method that can better binning reads from low abundance species and unknown species.

CHAPTER 5 BINNING METAGENOMIC READS BASED ON CLUSTERING OF MARKOV CHAINS

5.1 Background

MBMC has comparable performances when binning reads from known species than existing approaches and better performances when binning reads from unknown species. We observed that MBMC tended to divide reads from an unknown species into multiple small bins. How to combine these small bins together to reconstruct a unique bin for an unknown species seems challenging and worth further investigation. In addition, although MBMC showed superior performance than existing approaches on simulated and real experimental datasets with unknown species, in practice, the low abundance of many unknown species in metagenomic projects is still challenging for existing methods including MBMC to figure out how to bin reads. It is thus important to be able to deal with unknown species with even lower abundance than the current MBMC can deal with.

In this study, we improve the method of MBMC. We separate all input reads into two categories according to the similarity of long k-mers with reference genomes, one is from known species; the other is from unknown species. For reads that are from known species, we bin the reads by comparing the long k-mers (k=31) with that in reference genomes. For reads that are from unknown species, we bin reads by clustering the Markov chains from contigs that are obtained from the assembly of these reads. Tested on both simulated and real datasets, our method showed great improvement compared with other methods when the reads are from low abundant unknown species.

5.2 Materials and Methods

5.2.1 Known Species Used and Their Representation

We used the same taxa mentioned in Section 4.2.1.

5.2.2 Simulated and Experimental Datasets

We generated four simulated datasets. Each dataset consisted of reads from three species with the coverage for each genome as 8. The three species were randomly selected from the 181 of 2773 completed sequenced microbial genomes that did not have complete taxonomical information.

We used the same human gut dataset mentioned in section 4.2, which contains three species, and only one of them was known.

5.2.3 Binning Reads Based on the Clustering of Markov Chains

Here we only talk about how to bin reads from unknown species, which is the procedure shown in the right part of the following figure.

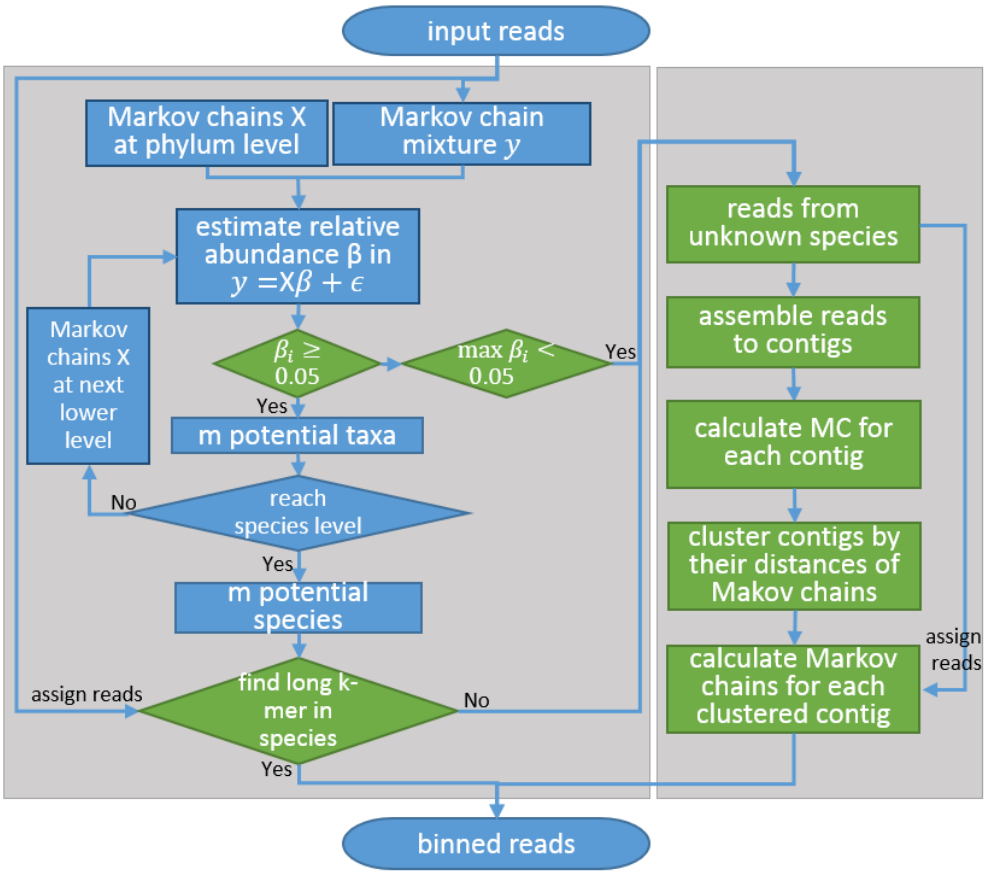

Figure 7 Flowchart of binning metagenomic reads based on clustering of Markov chains

The procedure is as following:

At the beginning, we estimate the relative abundance of input reads from phylum to species level. If the maximum of the relative abundance smaller than 0.05, then all reads will be from unknown species. All input reads will be assigned to the species with β>=0.05. Based on the existence of long k-mers (k=31) in potential species, there will be a group of reads that may be from unknown species.

For the reads that are from unknown species, we first assemble reads to contigs. The widely used tool Velvet [82] is used to assemble these reads with k-mer (k=41). The output contigs whose

length were larger than 500bp will be kept to do the next analysis. This is to ensure that we can have enough sequences to train the Markov chains. Then we calculate the Markov chains for each of contig, because the length of contigs is short and we only care about the similarity of Markov chains, here we used lower order Markov chains (e.g. 4-th order). To do clustering, we first need to calculate the distances of Markov chains. For every pair of Markov chains, we will calculate its distance as the average difference of each transition probability. We will cluster Markov chains from all contigs based their distances using average linkage. The tree will be cut based on the distance cutoffs. Here the cutoff is defined as 0.7*(distance value at root). The contigs at the leaves will be clustered if their parents' distances<cutoff. Here we only consider clusters that contain more than 100 contigs. The number of clusters will be the number of predicted unknown species. Each set of merged contigs (cluster) represents a group of sequences that may be from the same species. After that, in order to bin reads, we calculate the Markov chains for each of merged contigs, and we used 7-th order Markov chains in order to have a higher binning accuracy. Because not all reads can be directly mapped to these clustered contigs, we used lower order Markov chains to grasp information in short sequences. Those unassigned reads will be assigned to one of the merged contigs based on the similarity of the Markov chains.

5.3 Results

We tested out method using both simulated and real datasets. Each simulated dataset contains three species with the same genome coverage 8. The real human gut dataset contains three species, only one of them is known. Table 11 showed the binning accuracy for these six datasets. Here we only

used at most the top 5000 longer contigs to do the contig clustering in order to reduce the computational cost. Because Kraken has similar performance with MEGAN, so here we only showed the results for one of them. Overall, our method has better or comparable performance or MBMC and other methods when the species are unknown.

Table 11 Binning accuracy

datasets (#species)	accuracy [#species]	#contigs used (# all contigs)	MBMC [m]	MetaCluster	Abundancebin	MEGAN5 [m]
0_1[3]	89.27%[4]	3424(3424)	65.39%[12]	63.60%	39.71%	0.00%[0]
0_2[3]	95.3%[3]	4772(4772)	68.55%[11]	60.16%	39.66%	0.00%[0]
0_3[3]	95.04%[6]	5000(6046)	79.15%[13]	61.56%	36.59%	0.00%[0]
0_4[3]	94.7%[4]	5000(6940)	63.87%[15]	58.38%	35.00%	0.00%[0]
human gut[3]	85.87%[3]	3159(3159)	86.88%[8]	71.77%	69.68%	14.72%[1]

5.4 Discussion and Conclusions

We developed a metagenomic binning method that can deal with unknown species efficiently. This method first separates all input reads into two categories according to the similarity of long k-mers with reference genomes, one is from known species; the other is from unknown species. For reads that are from known species, we bin the reads by comparing the long k-mers (k=31) with that in reference genomes. For reads that are from unknown species, we bin reads by clustering the Markov chains from contigs that are obtained from the assembly of these

reads. Tested on both simulated and real datasets, our method showed great improvement compared with other methods when the reads are from low abundant unknown species.

Although this method showed better performance for datasets with unknown species, there are still some problems with it. First, the essential step of this method is to do reads assembling, how to assemble reads efficiently is still an open problem. Second, instead of using Markov chain, how to build a model to better represent a group of reads also need further investigation. Third, due to the limitation of the computers, we can't do clustering for very large distance matrix, so how to do clustering for large data is also a problem.

CHAPTER 6: CONCLUSIONS

Binning metagenomic reads is one of the fundamental steps in metagenomic studies. Current methods usually cannot work well when the genome coverage is small or when the datasets contain unknown species. To address these problems, we developed one taxonomy-independent and three taxonomy-dependent methods to bin metagenomic reads.

The taxonomy-independent method called MBBC, bins reads based on the difference of k-mer frequency distributions from different species without the reference genomes. This approach utilizes the k-mer frequency distributions and the Markov property to bin the reads. The first two taxonomy-dependent methods both bin reads by measuring the similarity of reads to the trained Markov chains from different taxa. The taxonomical decision tree method builds five taxonomical decision trees and then assigns reads to each of the decision trees. MBMC uses the linear regression model to model the relationship of input reads with known reference genomes, and selects potential taxa using the OLS method. The third taxonomy-dependent method bins reads by combining the methods of MBMC with clustering the Markov chains from assembled reads so that it can work on datasets that contain both known and unknown species. It first divides the input reads into known or unknown categories based on the sufficiently long k-mers. Then for reads from unknown species, it assigns the reads to the Markov chains trained from assembled reads. By testing on both simulated and real datasets, these tools showed superior or comparable performance with various state of the art methods.

Although they showed comparable or better performance than other methods, there are still a lot of works to do. MBBC does not perform well when the genome coverage of different

microbial species is small (<2-fold difference) or the reads abundance is low, which is a common problem in the taxonomy-independent methods. Because such methods are always based on the assumption that different species have different genome coverage. The taxonomical decision tree method which is a taxonomical-dependent method performs well for datasets when their genome coverage ratio is small. However, this method is not efficient and accurate since it needs to do clustering to build decision tree and needs to assign reads to these five decision trees. Each step will generate certain errors. Although MBMC performs better than taxonomical decision tree method and is much more efficient and accurate when the datasets contain unknown species, they cannot predict the correct number of species and they usually bin reads from one unknown species into multiple small groups. Besides this, the key problem is how we know that there exist unknown species in a dataset, how many of them are unknown, and how to bin reads better for these unknown species. The last method was proposed to address most of these problems. It can achieve higher accuracy when datasets contain both known and unknown species, but there exist other problems in this method. For example, the essential step of this method is to do reads assembling, while how to assemble reads efficiently is still an open problem. Another problem is that we cannot do clustering for very large distance matrix due to the limited computer resources.

We have attempted to further improve reads binning by using some other strategies. For examples, we tried to combine lower taxonomical level information with those from higher taxonomical levels. Besides assigning reads to the decision trees directly, we can use some strategies to better utilize the taxonomical information at different levels. And besides modeling the genome sequences with Markov Chains, we can use HMM model or other models to see

whether it can represent the genome sequences better. And since sufficiently long k-mers are usually unique in each genome, it may also helpful in binning metagenomic reads. Also, the reads assembly can aid the reads binning. With longer sequences, we may do reads binning better. How to combine these strategies and improve our current methods to bin reads better will be our future works.

LIST OF REFERENCES

1. Wooley, J.C., A. Godzik, and I. Friedberg, *A primer on metagenomics.* PLoS Comput Biol, 2010. **6**(2): p. e1000667.
2. DeLong, E.F. and N.R. Pace, *Environmental diversity of bacteria and archaea.* Systematic biology, 2001. **50**(4): p. 470-478.
3. Savage, D.C., *Microbial ecology of the gastrointestinal tract.* Annual Reviews in Microbiology, 1977. **31**(1): p. 107-133.
4. Handelsman, J., *Metagenomics: application of genomics to uncultured microorganisms.* Microbiology and molecular biology reviews, 2004. **68**(4): p. 669-685.
5. Kennedy, J., J.R. Marchesi, and A.D. Dobson, *Marine metagenomics: strategies for the discovery of novel enzymes with biotechnological applications from marine environments.* Microb Cell Fact, 2008. **7**(1): p. 27.
6. Rappé, M.S. and S.J. Giovannoni, *The uncultured microbial majority.* Annual Reviews in Microbiology, 2003. **57**(1): p. 369-394.
7. Simon, C. and R. Daniel, *Metagenomic analyses: past and future trends.* Applied and environmental microbiology, 2011. **77**(4): p. 1153-1161.
8. Kunin, V., et al., *A bioinformatician's guide to metagenomics.* Microbiology and Molecular Biology Reviews, 2008. **72**(4): p. 557-578.
9. Gilbert, J.A. and C.L. Dupont, *Microbial metagenomics: beyond the genome.* Annual Review of Marine Science, 2011. **3**: p. 347-371.
10. Mande, S.S., M.H. Mohammed, and T.S. Ghosh, *Classification of metagenomic sequences: methods and challenges.* Briefings in bioinformatics, 2012: p. bbs054.
11. Tyson, G.W., et al., *Community structure and metabolism through reconstruction of microbial genomes from the environment.* Nature, 2004. **428**(6978): p. 37-43.
12. Hutchison, C.A., *DNA sequencing: bench to bedside and beyond.* Nucleic acids research, 2007. **35**(18): p. 6227-6237.
13. Venter, J.C., et al., *Environmental genome shotgun sequencing of the Sargasso Sea.* science, 2004. **304**(5667): p. 66-74.
14. Eisen, J.A., *Environmental shotgun sequencing: its potential and challenges for studying the hidden world of microbes.* PLoS Biol, 2007. **5**(3): p. e82.
15. Altschul, S.F., et al., *Basic local alignment search tool.* Journal of molecular biology, 1990. **215**(3): p. 403-410.
16. Kent, W.J., *BLAT—the BLAST-like alignment tool.* Genome research, 2002. **12**(4): p. 656-664.
17. Huson, D.H., et al., *MEGAN analysis of metagenomic data.* Genome research, 2007. **17**(3): p. 377-386.
18. Haque, M.M., et al., *SOrt-ITEMS: Sequence orthology based approach for improved taxonomic estimation of metagenomic sequences.* Bioinformatics, 2009. **25**(14): p. 1722-1730.

19. Stark, M., et al., *MLTreeMap-accurate Maximum Likelihood placement of environmental DNA sequences into taxonomic and functional reference phylogenies.* BMC genomics, 2010. **11**(1): p. 461.
20. Matsen, F.A., R.B. Kodner, and E.V. Armbrust, *pplacer: linear time maximum-likelihood and Bayesian phylogenetic placement of sequences onto a fixed reference tree.* BMC bioinformatics, 2010. **11**(1): p. 538.
21. Meyer, F., et al., *The metagenomics RAST server–a public resource for the automatic phylogenetic and functional analysis of metagenomes.* BMC bioinformatics, 2008. **9**(1): p. 386.
22. Wood, D.E. and S.L. Salzberg, *Kraken: ultrafast metagenomic sequence classification using exact alignments.* Genome Biol, 2014. **15**(3): p. R46.
23. Tanaseichuk, O., J. Borneman, and T. Jiang, *Separating metagenomic short reads into genomes via clustering.* Algorithms for Molecular Biology, 2012. **7**(1): p. 27.
24. Rosen, G.L., E.R. Reichenberger, and A.M. Rosenfeld, *NBC: the Naive Bayes Classification tool webserver for taxonomic classification of metagenomic reads.* Bioinformatics, 2011. **27**(1): p. 127-129.
25. Diaz, N.N., et al., *TACOA–Taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach.* BMC bioinformatics, 2009. **10**(1): p. 56.
26. Brady, A. and S.L. Salzberg, *Phymm and PhymmBL: metagenomic phylogenetic classification with interpolated Markov models.* Nature methods, 2009. **6**(9): p. 673-676.
27. Mohammed, M.H., et al., *SPHINX—an algorithm for taxonomic binning of metagenomic sequences.* Bioinformatics, 2011. **27**(1): p. 22-30.
28. Chatterji, S., et al. *CompostBin: A DNA composition-based algorithm for binning environmental shotgun reads*. in *Research in Computational Molecular Biology*. 2008. Springer.
29. Wu, Y.-W. and Y. Ye. *A novel abundance-based algorithm for binning metagenomic sequences using l-tuples*. in *Research in Computational Molecular Biology*. 2010. Springer.
30. Wang, Y., et al., *MetaCluster 5.0: a two-round binning approach for metagenomic data for low-abundance species in a noisy sample.* Bioinformatics, 2012. **28**(18): p. i356-i362.
31. Tanaseichuk, O., J. Borneman, and T. Jiang, *A probabilistic approach to accurate abundance-based binning of metagenomic reads*, in *Algorithms in Bioinformatics*. 2012, Springer. p. 404-416.
32. Teeling, H. and F.O. Glöckner, *Current opportunities and challenges in microbial metagenome analysis—a bioinformatic perspective.* Briefings in bioinformatics, 2012: p. bbs039.
33. Albertsen, M., et al., *Genome sequences of rare, uncultured bacteria obtained by differential coverage binning of multiple metagenomes.* Nature biotechnology, 2013. **31**(6): p. 533-538.
34. Dempster, A.P., N.M. Laird, and D.B. Rubin, *Maximum likelihood from incomplete data via the EM algorithm.* Journal of the royal statistical society. Series B (methodological), 1977: p. 1-38.

35. Venter, J.C., et al., *Environmental genome shotgun sequencing of the Sargasso Sea.* Science, 2004. **304**(5667): p. 66-74.
36. Leung, H.C., et al., *A robust and accurate binning algorithm for metagenomic sequences with arbitrary species abundance ratio.* Bioinformatics, 2011. **27**(11): p. 1489-95.
37. Schreiber, F., et al., *Treephyler: fast taxonomic profiling of metagenomes.* Bioinformatics, 2010. **26**(7): p. 960-1.
38. Brady, A. and S.L. Salzberg, *Phymm and PhymmBL: metagenomic phylogenetic classification with interpolated Markov models.* Nat Methods, 2009. **6**(9): p. 673-6.
39. Diaz, N.N., et al., *TACOA: taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach.* BMC Bioinformatics, 2009. **10**: p. 56.
40. Gerlach, W., et al., *WebCARMA: a web application for the functional and taxonomic classification of unassembled metagenomic reads.* BMC Bioinformatics, 2009. **10**: p. 430.
41. Huson, D.H., et al., *MEGAN analysis of metagenomic data.* Genome Res, 2007. **17**(3): p. 377-86.
42. Krause, L., et al., *Phylogenetic classification of short environmental DNA fragments.* Nucleic Acids Res, 2008. **36**(7): p. 2230-9.
43. McHardy, A.C., et al., *Accurate phylogenetic classification of variable-length DNA fragments.* Nature methods, 2007. **4**(1): p. 63-72.
44. Chatterji, S., et al. *CompostBin: A DNA Composition-Based Algorithm for Binning Environmental Shotgun Reads*. in *Proceedings of the 12th annual international conference on Research in computational molecular biology*. 2008. Springer-Verlag Berlin, Heidelberg.
45. Yang, B., et al., *Unsupervised binning of environmental genomic fragments based on an error robust selection of l-mers.* BMC Bioinformatics, 2010. **11 Suppl 2**: p. S5.
46. Wu, Y. and Y. Ye. *A Novel Abundance-Based Algorithm for Binning Metagenomic Sequences Using l-Tuples*. in *Research in Computational Molecular Biology, 14th Annual International Conference, RECOMB 2010*. 2010. Lisbon, Portugal: Springer
47. Wang, Y., et al., *MetaCluster 5.0: a two-round binning approach for metagenomic data for low-abundance species in a noisy sample.* Bioinformatics, 2012. **28**(18): p. i356-i362.
48. Ghosh, T.S., M. Monzoorul Haque, and S.S. Mande, *DiScRIBinATE: a rapid method for accurate taxonomic classification of metagenomic sequences.* BMC Bioinformatics, 2010. **11 Suppl 7**: p. S14.
49. Horton, M., N. Bodenhausen, and J. Bergelson, *MARTA: a suite of Java-based tools for assigning taxonomic status to DNA sequences.* Bioinformatics, 2010. **26**(4): p. 568-9.
50. Matsen, F.A., R.B. Kodner, and E.V. Armbrust, *pplacer: linear time maximum-likelihood and Bayesian phylogenetic placement of sequences onto a fixed reference tree.* BMC Bioinformatics, 2010. **11**: p. 538.
51. Monzoorul Haque, M., et al., *SOrt-ITEMS: Sequence orthology based approach for improved taxonomic estimation of metagenomic sequences.* Bioinformatics, 2009. **25**(14): p. 1722-30.

52. Stark, M., et al., *MLTreeMap--accurate Maximum Likelihood placement of environmental DNA sequences into taxonomic and functional reference phylogenies.* BMC Genomics, 2010. **11**: p. 461.
53. Wu, M. and J.A. Eisen, *A simple, fast, and accurate method of phylogenomic inference.* Genome Biol, 2008. **9**(10): p. R151.
54. Apweiler, R., et al., *UniProt: the Universal Protein knowledgebase.* Nucleic Acids Res, 2004. **32**(Database issue): p. D115-9.
55. Punta, M., et al., *The Pfam protein families database.* Nucleic Acids Res, 2012. **40**(Database issue): p. D290-301.
56. Bentley, S.D. and J. Parkhill, *Comparative genomic structure of prokaryotes.* Annu Rev Genet, 2004. **38**: p. 771-92.
57. Teeling, H., et al., *Application of tetranucleotide frequencies for the assignment of genomic fragments.* Environ Microbiol, 2004. **6**(9): p. 938-47.
58. Mande, S.S., M.H. Mohammed, and T.S. Ghosh, *Classification of metagenomic sequences: methods and challenges.* Brief Bioinform, 2012. **13**(6): p. 669-81.
59. Teeling, H. and F.O. Glockner, *Current opportunities and challenges in microbial metagenome analysis--a bioinformatic perspective.* Brief Bioinform, 2012. **13**(6): p. 728-42.
60. Dempster, A., Laird, NM, Rubin, DB., *Maximum Likelihood from Incomplete Data via the EM Algorithm.* Journal of the Royal Statistical Society, 1977. **39**(1): p. 1–38.
61. White, J.R., et al., *Figaro: a novel statistical method for vector sequence removal.* Bioinformatics, 2008. **24**(4): p. 462-7.
62. Delcher, A.L., S.L. Salzberg, and A.M. Phillippy, *Using MUMmer to identify similar regions in large sequence sets.* Curr Protoc Bioinformatics, 2003. **Chapter 10**: p. Unit 10 3.
63. Li, R., et al., *SOAP2: an improved ultrafast tool for short read alignment.* Bioinformatics, 2009. **25**(15): p. 1966-7.
64. Richter, D.C., et al., *MetaSim: a sequencing simulator for genomics and metagenomics.* PLoS One, 2008. **3**(10): p. e3373.
65. Li, X. and M.S. Waterman, *Estimating the repeat structure and length of DNA sequences using L-tuples.* Genome Res, 2003. **13**(8): p. 1916-22.
66. Li, L., et al., *A mixture model-based discriminate analysis for identifying ordered transcription factor binding site pairs in gene promoters directly regulated by estrogen receptor-alpha.* Bioinformatics, 2006. **22**(18): p. 2210-6.
67. Pride, D.T., et al., *Evolutionary implications of microbial genome tetranucleotide frequency biases.* Genome Res, 2003. **13**(2): p. 145-58.
68. Audic, S. and J.M. Claverie, *Self-identification of protein-coding regions in microbial genomes.* Proc Natl Acad Sci U S A, 1998. **95**(17): p. 10026-31.
69. Rosen, G., et al., *Metagenome fragment classification using N-mer frequency profiles.* Adv Bioinformatics, 2008. **2008**: p. 205969.
70. Salzberg, S.L., et al., *Microbial gene identification using interpolated Markov models.* Nucleic Acids Res, 1998. **26**(2): p. 544-8.

71. Wang, Y., H. Hu, and X. Li, *MBBC: an efficient approach for metagenomic binning based on clustering.* BMC bioinformatics, 2015. **16**(1): p. 36.
72. Reddy, T., et al., *The Genomes OnLine Database (GOLD) v. 5: a metadata management system based on a four level (meta) genome project classification.* Nucleic acids research, 2014: p. gku950.
73. Scholz, M.B., C.-C. Lo, and P.S. Chain, *Next generation sequencing and bioinformatic bottlenecks: the current state of metagenomic data analysis.* Current opinion in biotechnology, 2012. **23**(1): p. 9-15.
74. Lazarevic, V., et al., *Metagenomic study of the oral microbiota by Illumina high-throughput sequencing.* Journal of microbiological methods, 2009. **79**(3): p. 266-271.
75. Krause, L., et al., *Phylogenetic classification of short environmental DNA fragments.* Nucleic acids research, 2008. **36**(7): p. 2230-2239.
76. Wu, Y.-W. and Y. Ye, *A novel abundance-based algorithm for binning metagenomic sequences using l-tuples.* Journal of Computational Biology, 2011. **18**(3): p. 523-534.
77. Luo, R., et al., *SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler.* Gigascience, 2012. **1**(1): p. 18.
78. Qin, J., et al., *A human gut microbial gene catalogue established by metagenomic sequencing.* nature, 2010. **464**(7285): p. 59-65.
79. Kultima, J.R., et al., *MOCAT: a metagenomics assembly and gene prediction toolkit.* 2012.
80. Namiki, T., et al., *MetaVelvet: an extension of Velvet assembler to de novo metagenome assembly from short sequence reads.* Nucleic acids research, 2012. **40**(20): p. e155-e155.
81. Alneberg, J., et al., *Binning metagenomic contigs by coverage and composition.* Nature methods, 2014. **11**(11): p. 1144-1146.
82. Zerbino, D.R. and E. Birney, *Velvet: algorithms for de novo short read assembly using de Bruijn graphs.* Genome research, 2008. **18**(5): p. 821-829.