A Novel Abundance-based Algorithm for Binning Metagenomic Sequences Using *l*-tuples

Yu-Wei Wu and Yuzhen Ye

School of Informatics and Computing, Indiana University,
901 E. 10th Street, Bloomington, IN 47408
{yuwwu, yye}@indiana.edu

Abstract. Metagenomics is the study of microbial communities sampled directly from their natural environment, without prior culturing. Among the computational tools recently developed for metagenomic sequence analysis, binning tools attempt to classify all (or most) of the sequences in a metagenomic dataset into different bins (i.e., species), based on various DNA composition patterns (e.g., the tetramer frequencies) of various genomes. Composition-based binning methods, however, cannot be used to classify very short fragments, because of the substantial variation of DNA composition patterns within a single genome. We developed a novel approach (AbundanceBin) for metagenomics binning by utilizing the different abundances of species living in the same environment. AbundanceBin is an application of the Lander-Waterman model to metagenomics, which is based on the l-tuple content of the reads. AbundanceBin achieved accurate, unsupervised, clustering of metagenomic sequences into different bins, such that the reads classified in a bin belong to species of identical or very similar abundances in the sample. In addition, AbundanceBin gave accurate estimations of species abundances, as well as their genome sizes—two important parameters for characterizing a microbial community. We also show that AbundanceBin performed well when the sequence lengths are very short (e.g. 75 bp) or have sequencing errors.

Keywords: Binning, metagenomics, EM algorithm, Poisson distribution

1 Introduction

Metagenomic studies have resulted in vast amounts of sequence, sampled from a variety of environments, leading to new discoveries and insights into the uncultured microbial world [1], such as the diversity of microbes in different environments [2, 3], microbial (and microbe-host) interactions [4, 5], and the environmental and evolutionary processes that shape these communities [6]. Current metagenomic projects are facilitated by the rapid advancement of DNA sequencing techniques. Recently developed next-generation sequencing (NGS) technologies [7] (such as Roche/454 [8] and Illumina/Solexa [9]) provide lower cost sequence, without the cloning step inherent in conventional capillary-based

methods. These NGS technologies have increased the amount of sequence data obtained in a single run by several orders of magnitude.

One of the primary goals of metagenomic projects is to characterize the organisms present in an environmental sample and identify the metabolic roles each organism plays. Many computational tools have been developed to infer species information from raw short reads directly, without the need for assembly—assembly of metagenomic sequences into genomes is extremely difficult, since the reads are often very short and are sampled from multiple genomes. We categorize the various computational tools for the estimation of taxonomic content into two basic classes, and will review them briefly below.

The first class of computational tools maps metagenomic sequences to taxa with or without using phylogeny (often referred to as the *phylotyping* of metagenomic sequences), utilizing similarity searches of the metagenomic sequences against a database of known genes/proteins. MEGAN [10] is a representative similarity-based phylotyping tool, which applies a simple lowest common ancestor algorithm to assign reads to taxa, based on BLAST results. Phylogenetic analysis of marker genes, including 16S rRNA genes [11], DNA polymerase genes [12], and the 31 marker genes defined by [13], are also applied to determining taxonomic distribution. MLTreeMap [14] and AMPHORA [15] are two phylogeny-based phylotyping tools that have been developed, using phylogenetic analysis of marker genes for taxonomic distribution estimation: MLTreeMap uses TreePuzzle [16], and AMPHORA uses PHYML [17]. CARMA [18] searches for conserved Pfam domains and protein families [19] in raw metagenomic sequences and classifies them into a higher-order taxonomy, based on the reconstruction of a phylogenetic tree for each matching Pfam family. These similarity-based and phylogeny-based phylotyping tools have a common limitation: they do not say much about the taxonomic distribution of the reads that do not match known genes/proteins, which may constitute the majority of the metagenomic sequences for some samples. A more recent approach PhymmBL [20] combines similarity search and DNA composition patterns to map metagenomic sequences to taxa, achieving an improved phylogenetic classification for short reads.

A second class of computational tools attempts to solve a related but distinct problem, the *binning* problem, which is to cluster metagenomic sequences into different bins (species). Most existing computational tools for binning utilize DNA composition. The basis of these approaches is that genome G+C content, dinucleotide frequencies, and synonymous codon usage vary among organisms, and are generally characteristic of evolutionary lineages [21]. Tools in this category include TETRA [22], MetaClust [23], CompostBin [24], TACOA [25], and a genomic barcode based method [26]. All the existing DNA composition based methods achieve a reasonable performance only for long reads—at least 800 bp. TACOA is able to classify genomic fragments of length 800 and 1000 bp into the phylogenetic rank of class with high accuracy (accurate classification at the order and genus level requires fragments of ≥ 3 kb) [25]; CompostBin was tested on simulated datasets of 1 kb reads [24]. This length limitation ($\sim$1 kb) will be difficult (if not impossible) to break, because of the local variation of DNA com-

position [21]. Foerstner et al. reported that the GC content of complex microbial communities seems to be globally and actively influenced by the environment, suggesting that it may be even harder to distinguish fragments from different species living in the same environment, based on DNA composition [27].

In addition, metagenomic sequences may be sampled from species of very different abundances (for example, the Acid Mine Drainage project [28] found two dominant species, accompanied by several other rarer species in that environment), and the difference in abundances may affect the classification results for DNA-composition based approaches. For example, a weighted PCA was adopted instead of a standard PCA in CompostBin, to reduce the dimension of composition space, considering that the within-species variance in the more abundant species might be overwhelming, compared to between-species variance [24].

Here we report a novel *binning* tool, AbundanceBin, which can be used to classify very short sequences sampled from species with different abundance levels (Fig. 1a). The fundamental assumption of our method is that reads are sampled from genomes following a Poisson distribution [29]. In the context of metagenomics, we model the sequencing reads as a mixture of Poisson distributions. We propose an Expectation-Maximization (EM) algorithm to find parameters for the Poisson distributions, which reflect the relative abundance levels of the source species. We note that a similar method was first described by Li and Waterman, for the purpose of modeling the repeat content in a conventional genome sequencing project [30], and Sharon et al. proposed a statistical framework for protein family frequency estimation from metagenomic sequences based on the Lander-Waterman model [31], given that different protein families are of different lengths. AbundanceBin assigns reads to bins using the fitted Poisson distribution. In addition, AbundanceBin gives an estimation of the genome size (or the concatenated genome size of species of the same or very similar abundances), and the coverage (which reflects the abundances of species) of each bin, all in an unsupervised manner. Since AbundanceBin is based on l-tuple content (not the composition), in principle it can be applied to classify reads that are as short as l bp. We report below first the algorithm and then tests of AbundanceBin on several synthetic metagenomic datasets and a real metagenomic dataset.

2 Methods

Randomized shotgun sequencing procedures result in unequal sampling of different genomes, especially when the species abundance levels differ. We seek to discover the abundance values as well as the genome sizes automatically and then bin reads accordingly. We assume that the distribution of sequenced reads follows the Lander-Waterman model [29], which calculates the coverage of each nucleotide position using a Poisson distribution. We thus view the sequencing procedure in metagenomic projects as a mixture of m Poisson distributions, m being the number of species. The goal is to find the mean values λ_1 to λ_m, which are the abundance levels of the species, of these Poisson distributions.

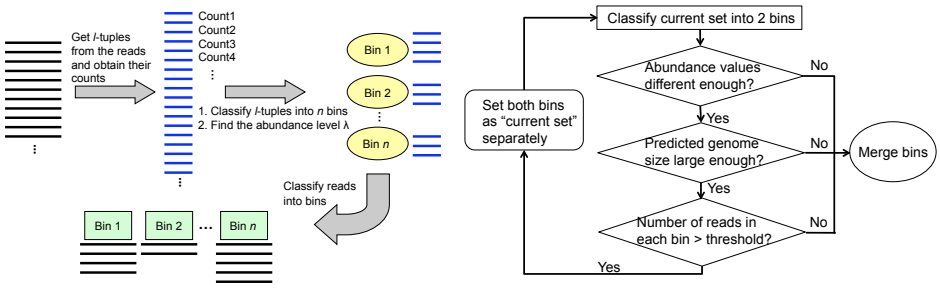


Fig. 1. (a) A schematic illustration of AbundanceBin pipeline, and (b) the recursive binning approach used to automatically determine the number of bins.

2.1 Mixed Poisson Distributions

In random shotgun sequencing of a genome, the probability that a read starting from a certain position is $N/(G-L+1)$, where N is the number of reads, G is the genome size, and L is the length of reads. $N/(G-L+1) \approx N/G$, given $G \gg L$. Assume x is a read and a l-tuple w belongs to x. The number of occurrences of w in the set of reads follows a Poisson distribution with parameter $\lambda = N(L-l+1)/(G-L+1) \approx NL/G$ in a random sampling process with read length L.

Similarly, for a metagnomic dataset, the number of occurrences of w in the set of reads also follows a Poisson distribution with parameter $\lambda = N(L-l+1)/(G-L+1) \approx NL/G$, but G in this case is the total length of the genomic sequences contained in the metagenomic dataset. In metagenomic datasets, the reads are from species with different abundances. If the abundance of a species i is n, the total number of occurrences of w in the whole set of reads coming from this species should follow a Poisson distribution with parameter $\lambda_i = n\lambda$, due to the additivity of Poisson distribution. Now the problem of finding the relative abundance levels of different species is transformed to the modeling of mixed Poisson distributions.

2.2 Binning Algorithm

Given a set of metagenomic sequences, the algorithm starts by counting l-tuples in all reads (Fig. 1a). Then we use an Expectation-Maximization (EM) algorithm to approximate the species abundance level and the genome size of each species, which consists of 4 steps, as follows.

1. Initialize the total number of species S, their genome size l_i, and abundance level λ_i for $i = 1, 2, ..., S$.

2. Calculate the probability that the l-tuple w_j ($j = 1, 2, ..., W$; W is the total number of possible l-tuples) coming from ith species given its count $n(w_j)$ (see Appendix for details).

$$P(w_j \in s_i | n(w_j)) = \frac{l_i}{\sum_{m=1}^{S} l_m (\frac{\lambda_m}{\lambda_i})^{n(w_j)} e^{(\lambda_i - \lambda_m)}} \tag{1}$$

3. Calculate the new values for each l_i and λ_i.

$$l_i = \sum_{j=1}^{W} P(w_j \in s_i | n(w_j)) \tag{2}$$

$$\lambda_i = \frac{\sum_{j=1}^{W} n(w_j) P(w_j \in s_i | n(w_j))}{l_i} \tag{3}$$

4. Iterate step 2 and 3 until the parameters converge or the number of runs exceeds a maximum number of runs. The convergence of parameters is defined as

$$\forall \lambda_i \left\{ \left| \frac{\lambda_i^{t+1}}{\lambda_i^t} \right| < 10^{-5} \right\} \text{ and } \forall l_i \left\{ \left| \frac{l_i^{t+1}}{l_i^t} \right| < 10^{-5} \right\} \tag{4}$$

where $\lambda_i^{(t)}$ and $l_i^{(t)}$ represent the abundance level and genome length at iteration t, respectively. The maximum number of runs is set to 100 (which is sufficient for the convergence of the EM algorithm for all the cases we have tested).

Once the EM algorithm converges, we can estimate the probability of a read assigned to a bin, based on its l-tuples binning results as,

$$P(r_k \in s_i) = \frac{\prod_{w_j \in r_k} P(w_j \in s_i | n(w_j))}{\sum_{s_i \in S} \left(\prod_{w_j \in r_k} P(w_j \in s_i | n(w_j)) \right)} \tag{5}$$

where r_k is a given read, w_j is the l-tuples that belong to r_k, and s_i is any bin. A read will be assigned to the bin with the highest probability among all bins. A read remains unassigned if 90% of its l-tuples are excluded (counts too low or too high, see section 2.3), or if the highest probability is $< 50\%$.

2.3 Lower- and Upper-limit for *l*-tuple Counts

We set a lower- and upper-limit for l-tuple counts, as additional parameters, when we approximate λ_i and l_i. The lower-limit is introduced to deal with sequencing errors, and the upper-limit is introduced to handle l-tuples with extremely high counts, such as those from vector sequences or repeats of high copy numbers. Let the lower-limit be B_{lower} and the upper-limit be B_{upper}. Then the formula for calculating λ_i and l_i is modified to

$$l_i = \sum_{j=1}^{W} P(w_j \in s_i | n(w_j)), \forall n(w_j) > B_{lower} \wedge n(w_j) < B_{upper} \tag{6}$$

$$\lambda_i = \frac{\sum_{j=1}^{W} n(w_j) P(w_j \in s_i | n(w_j))}{l_i}, \forall n(w_j) > B_{lower} \wedge n(w_j) < B_{upper} \quad (7)$$

2.4 Automatic Determination of the Total Number of Bins by a Recursive Binning Approach

In the EM algorithm, we need to provide the number of bins as an input, in order to determine the parameters of the mixed Poisson distributions. However, this number is often unknown, as for most metagenomic projects. We implemented a recursive binning approach to determine the total number of bins automatically. The recursive binning approach works by separating the dataset into two bins and proceeds by further splitting bins (as shown in Fig. 1b)—it is a top-down approach. The recursive binning approach was motivated by the observation that reads from genomes with higher abundances are better classified than those with lower abundance. The recursive procedure continues if 1) the predicted abundance values of two bins differ significantly, i.e., $|\lambda_i - \lambda_j| / min(\lambda_i, \lambda_j) \geq 1/2$; 2) the predicted genome sizes are larger than a certain threshold (currently set to 400,000, considering that the smallest genomes of living organisms yet found are about 500,000 bp—*Nanoarchaeum equitans* has a genome of 490,885 bp, and *Mycoplasma genitalium* has a genome of 580,073 bp); and 3) the number of reads associated with each bin is larger than a certain threshold proportion (3%) of the total number of reads classified in the parent bin.

2.5 Performance Evaluation

We defined the classification error rate as the number of misclassified reads divided by the total number of reads. Chatterji et al [24] used a normalized error rate—the arithmetic average of the classification error rates for all the bins—to evaluate their binning approach CompostBin. We consider that the standard error rate, instead of normalized error rate, serves better for the performance evaluation of AbundanceBin, since AbundanceBin takes advantage of different species abundances. For comparison, we also provide the normalized classification error rates.

2.6 Metagenomic Datasets

We used MetaSim [32] to generate synthetic metagenomic datasets with reads sampled from species of various abundances. MetaSim takes as input a set of known genome sequences and an abundance profile, which determines the relative abundance of each genome sequence in a simulated dataset. The "Exact" profile defined by MetaSim is used to generate reads without sequencing errors, and "454" profile for reads generated with a 454 error model. The number of reads as well as the mean and variance of read lengths are adjusted accordingly: for average 400 bp, the mean value is set to 400 and the variance is set to 50; and for 75 bp, mean and variance are set to 75 and 5. All other settings are kept

as default. The genomes we used for generating synthetic metagenomic datasets, and the AMD metagenomic sequences and its scaffolds were downloaded from NCBI.

3 Results

We tested AbundanceBin on various synthetic metagenomic datasets with short and very short sequence lengths (75–400 bp), and the results show that AbundanceBin gives accurate classification of reads to different bins, and accurate estimation of the abundances—as well as the genome sizes—in each bin. We note that since these parameters are usually unknown in real metagenomic datasets, we focused on synthetic datasets for benchmarking. We also applied AbundanceBin to the actual AMD dataset and revealed a relatively clear picture of the complexity of the microbial community in that environment, consistent with the analysis reported in [28].

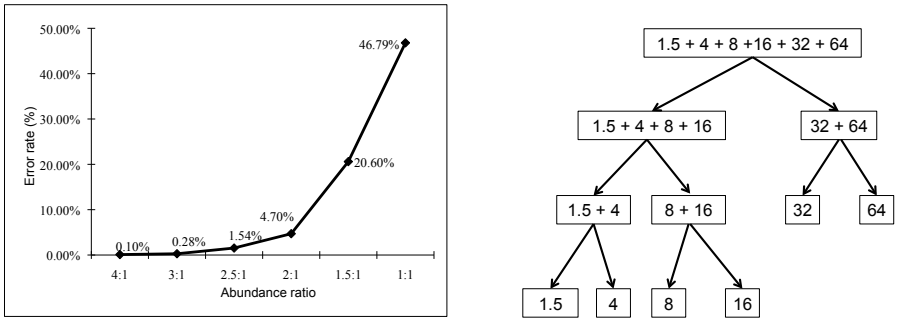
3.1 Tests of Abundance Differences and the Length of l-tuples

We did a series of experiments to test the abundance ranges of species required for accurate binning of reads. Fig. 2a shows the binning results for simulated short reads sampled from two genomes (*Mycoplasma genitalium* G37 and *Buchnera aphidicola* str. BP) at abundance ratios, 4:1, 3:1, 2.5:1, 2:1, 1.5:1, and 1:1 (with 50,000 simulated reads of ∼400 bases for each setting). The classification error rate is low if the abundance ratio is 2.0 (0.1% and 4.7% for ratio 4:1 and 2:1, respectively), but rises dramatically when the abundance ratio drops to 1.5:1 (the error rate is 20.6% for abundance ratio 1.5:1). We conclude that the abundance ratio needs to reach at least 2:1 for a good classification by AbundanceBin.

We also tested different lengths of l-tuples, and the results show that when l drops to 16, the binning performance dropped significantly for cases with two genomes. The performance improved slightly when l increases to 20 for cases with more than 3 genomes. Considering the performance on the tested cases, we chose to use $l = 20$ for the following experiments.

3.2 AbundanceBin Achieves Accurate Binning, Estimation of Species Abundance, and Genome Size

The binning results on several simulated datasets of short reads are summarized in Table 1. AbundanceBin achieved both accurate estimation of species abundances, and accurate assignment of reads to bins of different abundances. The classification error rates are 0.10% and 0.64% for the classification of reads of length 400 bp and 75 bp, respectively, sampled from two genomes (cases A and C in Table 1). The error rates for the classification of reads sampled from more genomes are slightly higher than for two-genome scenarios (e.g., the classification error rates for two synthetic metagenomic datasets with reads of length 400



(a) (b)

Fig. 2. (a) The classification error rates for classifying reads sampled from two genomes versus their abundance differences, and (b) the recursive binning of a read dataset into 6 bins of different abundances (each box represents a bin with the numbers indicating the abundance of the reads classified to that bin; e.g., the bin on the top has all the reads, which will be divided into two bins, one with reads of abundances 1.5, 4, 8 and 64, and the other bin with reads of abundances 32 and 64).

bp and 75 bp, sampled from 3 genomes, are 3.10% and 6.18%, respectively, as shown in Table 1). For the classification of reads sampled from more than two genomes, most of the errors occur in the least abundant bin. But AbundanceBin was still able to classify the reads from species of higher abundance correctly for all the tested synthetic metagenomic datasets, including one with reads sampled from 6 different genomes (see Table 2).

We emphasize here that AbundanceBin can bin reads as short as 75 bases with reasonable classification error rates, as shown in Table 1. As we discussed in the Introduction, binning of very short reads, such as 75 bases, is extremely difficult and cannot be achieved by any of the existing composition based binning approaches, due to the substantial variation in DNA composition within a single genome. AbundanceBin will also give an estimation of the genome size for each bin. As shown in Table 1, for most of the tested cases, the estimated genome sizes are very close to the real ones. We note that AbundanceBin will classify reads from different species of similar abundances into a single bin. In this case, the predicted genome size for that bin is actually the sum of the genome sizes of the species classified into that bin.

AbundanceBin also worked well on binning closely related species (closely related species often have similar genomes, and therefore it is often very difficult to separate reads sampled from closely related species). For the synthetic metagenomic datasets we tested, most reads from species that differ at only the species level can still be classified into correct bins with very low error rates. For examples, for two datasets, AbundanceBin resulted in binning with error rates

of 0.96% and 0.68% for the dataset simulated from the genomes of *Corynebacterium efficiens* YS-314 and *Corynebacterium glutamicum* ATCC 13032, and the dataset simulated from the genomes of *Helicobacter hepaticus* ATCC 51449 and *Helicobacter pylori* 26695 (both sets of genomes only differ at the species level), respectively. These results demonstrate the ability of our algorithm to separate short reads from closely related species, even if the species are of the same genus. (Note that AbundanceBin cannot separate reads from different strains of the same species into different bins.)

3.3 AbundanceBin can Handle Sequencing Errors

As mentioned in Methods, AbundanceBin can be configured to ignore l-tuples that only appear once to deal with sequencing errors, considering that those l-tuples are likely to be contributed by reads with sequencing errors and that the chance of having reads with sequencing errors at the same position will be extremely low. This may exclude some genuine l-tuples, but our tests reveal that AbundanceBin achieved even better classification if all l-tuples of count 1 are discarded (data not shown). AbundanceBin achieved slightly worse classification of reads when reads contain sequencing errors, as compared to the classification of simulated reads without sequencing errors (see cases E and F in Table 1). This is expected, given that many spurious l-tuples are generated with a 454 sequencing error model. For example, 12,901,691 20-tuples can be found in a dataset of simulated reads from two genomes with sequencing errors (case E in Table 1), 5 times more than the case without error models (2,370,720).

3.4 AbundanceBin Doesn't Require Prior Knowledge of the Total Number of Bins

Table 2 compares the performance of AbundanceBin using the recursive binning approach on several synthetic metagenomic datasets to that of AbundanceBin given the total number of bins. Overall the performances of the recursive binning approach are comparable to the cases with predefined bin numbers. Fig. 2b depicts the recursive binning results of the classification of one of the synthetic metagenomic datasets (which has reads sampled from 6 genomes) into 6 bins of different abundances (with classification error rate = 3.73%), starting with a bin that includes all the reads and ending with 6 bins each having reads correctly assigned to them. It is interesting that the recursive binning approach achieved even better performance for some cases. A simple explanation to this is that the recursive binning strategy may create bigger abundance differences, especially at the beginning of the binning process, and AbundanceBin works better at separating reads from species with greater abundance differences (see Fig. 2a). We note again that the high abundant bins are classified relatively well. The majority of errors occur in low abundant bins.

Table 1. Tests of AbundanceBin on synthetic metagenomic datasets (A-D without sequencing errors, and E-F with sequencing errors [a]).

ID	Spe [b]	Len [c]	Total reads	Bin	Abundance		Genome size		Error rate(%)
					Real	Predicted	Real	Predicted	
A	2	400 bp	50,000	1	27.23	26.27	580,076	570,859	0.10 (0.20 [d])
				2	6.83	6.49	615,980	614,605	
B	3	400 bp	50,000	1	24.64	23.78	580,076	568,549	3.10 (6.64)
				2	6.13	6.02	615,980	517,110	
				3	1.8	2.39	1,072,950	941,425	
C	2	75 bp	200,000	1	20.47	15.66	580,076	562,584	0.64 (1.07)
				2	5.08	3.92	615,980	608,401	
D	3	75 bp	200,000	1	27.6	20.93	580,076	565,859	6.18 (11.74)
				2	6.93	5.99	615,980	368,836	
				3	2.07	2.43	1,072,950	1,100,309	
E	2	297 bp	50,000	1	20.21	11.63	580,076	521,168	1.12 (0.99)
				2	5.07	3.01	615,980	945,435	
F	3	297 bp	150,000	1	55.48	30.58	580,076	559,395	8.20 (11.41)
				2	13.98	9.6	615,980	341,290	
				3	3.50	2.72	1,072,950	3,064,199	

[a]: The average sequencing error rate introduced is 3%, higher than the error rate of recent 454 machines (e.g., the accuracy rate reported in [33] is 99.5%). A 3% sequencing error can reduce the l-tuple counts by about half (i.e., about $1 - 0.97^{20} = 0.46$ of expected 20-mers without sequencing errors), which makes accurate estimation of abundance and genome size difficult. [b]: The number of species used in simulating each metagenomic dataset. The genomes used in these tests are *Mycoplasma genitalium* G37, *Buchnera aphidicola* str. BP, and *Chlamydia muridarum* Nigg. The first two genomes are used for the 2 species cases. [c]: The average length of the simulated reads. [d]: Normalized error rates (see Methods for details).

Table 2. Comparison of binning performance using the recursive binning approach ("Recursive") versus the binning when the total number of bins is given ("Predefined"))

Test cases	Error rate (normalized error rate)	
	Predefined	Recursive
3 genomes (no error model; 400 bp)	3.10% (6.64%)	3.24% (7.47%)
3 genomes (no error model; 75 bp)	6.18% (11.74%)	4.84% (9.31%)
3 genomes (454 error model; 297 bp)	8.21% (11.41%)	2.29% (4.21%)
4 genomes (no error model; 400bp)	1.12% (5.16%)	2.96% (6.96%)
6 genomes (no error model; 400bp)	2.50% (9.23%)	3.73% (13.07%)

3.5 Binning of Acid Mine Drainage (AMD) Datasets

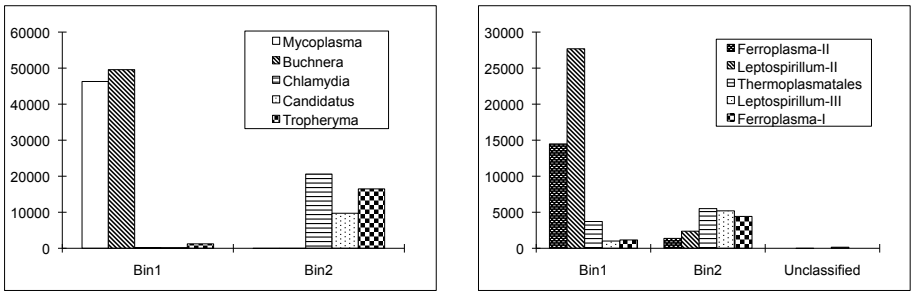
The AMD microbial community was reported to consist of two species of high abundance and three other less abundant species [28]. With the difference of two abundance levels in this environment, we expect that the algorithm could classify the AMD dataset into two bins. We applied AbundanceBin to a simulated AMD dataset (so that we have correct answers for comparison) and the real AMD dataset from [28].

The synthetic AMD dataset contains 150,000 reads from five genomes, with abundances 4:4:1:1:1. Our recursive binning approach automatically classified the reads into two bins with an error rate of 1.03% (see Fig. 3a). (Note here that each bin has reads sampled from multiple species. We consider that a read is classified correctly if it is classified into the bin of the correct abundance.) The binning accuracy dropped only slightly (with an error rate of 2.25%) for the synthetic AMD dataset when sequencing errors were introduced.

We also applied AbundanceBin to reads from the actual AMD dataset (downloaded from NCBI trace archive; 13696_environmental_sequence.007). AbundanceBin successfully classified these reads into exactly two bins (one of high abundance and one of low abundance) using the recursive binning approach (see Fig. 3b). Note the reads in this dataset have vector sequences, which resulted in a very small number of l-tuples of extremely high abundance (the highest count is 50,720)—this phenomena has been utilized for vector sequence removal, as described in [34]). Two approaches were employed to avoid the influences of the vector sequences: 1) we used the Figaro software package[34] to trim the vector sequences, and 2) we set an upper-limit for the count of all l-tuples, ignoring l-tuples with counts larger than the upper-limit (200 by default). We also downloaded the sequences of 5 scaffolds of the 5 partial genomes assembled from the AMD dataset, so that we can estimate the classification accuracy of AbundanceBin. The classification error rate of the AMD sequences is $\sim$14.38%. Note this error rate only gives us a rough estimation of the classification accuracy, since only 58% of the AMD reads can be mapped back to the assembled scaffolds based on similarity searches by BLAST—we mapped a read to a scaffold if the read matches the scaffold with BLAST E-value $\leq$ 1e-50, sequence similarity $\geq$ 95%, and a matched length of $\geq$ 70% of the read length. We emphasize that AbundanceBin achieved a much better classification (with an error rate of 1.03%) for the simulated AMD reads, for which we have correct answers to compare with.

4 Discussion

We have shown that our abundance based algorithm for binning has the ability to classify short reads from species with different abundances. Our approach has two unique features. First, our method is "unsupervised" (i.e., it doesn't require any prior knowledge for the binning). Second, our method is especially suitable for short reads, as long as the length of reads exceeds the length of the l-tuple (currently 20). AbundanceBin can in principle be applied to any metatagenomic sequences acquired by current NGS, without human interpretation.



(*a*) (*b*)

Fig. 3. The binning results for a simulated (a), and the actual (b) AMD datasets. The histogram shows the total number of reads from different genomes classified to each bin.

We implemented a simple strategy—excluding l-tuples that are counted only once from the abundance estimation—to handle sequencing errors, and tests have showed that AbundanceBin achieved better classification if all l-tuples of single count are discarded. One potential problem of discarding l-tuples of low counts is that some genuine l-tuples will be discarded as well, which results in a lower abundance estimation and a worse prediction of genome sizes, especially for the species with low abundance, as shown in Table 1. But we argue that AbundanceBin can still capture the relative abundances of different bins correctly, which is more important than the absolute values. Another potential problem is that reads from low abundant genomes may not be classified when sequencing errors are introduced in the reads. For example, the number of unclassified reads in a two-genome case (metagenomic dataset E in Table 1) is 12, and 389 in a three-genome case (metagenomic dataset F in Table 1). All unclassified reads in both cases belong to the least abundant species, indicating that the abundance values greatly affect the predicted results, especially when sequencing errors are present. We expect that both problems will become less problematic as sequencing coverage is increased, which is possible with massive throughput NGS techniques. As for the abundance ratio required for successful classification, we find that the ratio should be at least 2:1 to obtain an acceptable result. The required ratio, of course, is also affected by several other factors, such as the actual abundance level, the average length of reads, and the sequencing error rate. Our tests intentionally use well-classified datasets to allow us to follow changes in classification error resulting from abundance differences, but other factors besides the abundance ratio must also be considered.

AbundanceBin runs fast, and all the tests shown in the paper were completed within an hour (using single CPU on Intel(R) Xeon(R)@2.00GHz) with

very moderate memory usage. For example, binning of the synthetic metagenomic dataset A (see Table 1) requires 100MB memory, and dataset B 150MB. However, AbundanceBin may require large memory when working with very large datasets of short reads.

We expect that AbundanceBin will have three important applications. First it can be used for binning metagenomic sequences, as well as estimating species abundances and genome sizes. Second, it can be combined with other binning approaches. Note that AbundanceBin is not designed to separate reads from species of very similar abundances; we will develop an integrated method that combines AbundanceBin and other binning methods that use, for example, DNA composition. We expect that such an integrated method will achieve better classification performance by incorporating orthogonal information (abundance and composition, for example). Finally, we expect that applying AbundanceBin to separate reads into bins of different abundances (coverages), prior to the assembly of metagenomic sequences, will improve the quality of genome assembly.

5 Acknowledgements

This research was supported by NIH grant 1R01HG004908-02 and NSF CAREER award DBI-084568. We thank Dr. Haixu Tang for helpful discussions, and Dr. Thomas G. Doak for reading the manuscript.

References

1. Galperin M. Metagenomics: from acid mine to shining sea. *Environ Microbiol*, 2004; **6**:543545.
2. Tringe S, von Mering C, Kobayashi A, et al. Comparative metagenomics of microbial communities. *Science*, 2005; **308**(5721):554–557.
3. Dinsdale E, Pantos O, Smriga S, et al. Microbial ecology of four coral atolls in the northern line islands. *PLoS ONE*, 2008; **3**(2):e158.
4. Turnbaugh PJ, Ley RE, Mahowald MA, et al. An obesity-associated gut microbiome with increased capacity for energy harvest. *Nature*, 2006; **444**(7122):1027–1131.
5. Turnbaugh PJ, Hamady M, Yatsunenko T, et al. A core gut microbiome in obese and lean twins. *Nature*, 2009; **457**(7228):480–4.
6. Dinsdale EA, Edwards RA, Hall D, et al. Functional metagenomic profiling of nine biomes. *Nature*, 2008; **452**(7187):629–32.
7. Hutchison C. A. r. DNA sequencing: bench to bedside and beyond. *Nucleic Acids Res*, 2007; **35**(18):6227–37.
8. Margulies M, Egholm M, Altman WE, et al. Genome sequencing in microfabricated high-density picolitre reactors. *Nature*, 2005; **437**(7057):376–80.
9. Bentley DR. Whole-genome re-sequencing. *Curr Opin Genet Dev*, 2006; **16**(6):545–52.
10. Huson DH, Auch AF, Qi J, et al. MEGAN analysis of metagenomic data. *Genome Res*, 2007; **17**(3):377–386.

11. Chakravorty S, Helb D, Burday M, et al. A detailed analysis of 16s ribosomal RNA gene segments for the diagnosis of pathogenic bacteria. *J Microbiol Methods*, 2007; **69**(2):330–9.
12. Monier A, Claverie JM, Ogata H. Taxonomic distribution of large DNA viruses in the sea. *Genome Biol*, 2008; **9**(7):R106.
13. Ciccarelli FD, Doerks T, von Mering C, et al. Toward automatic reconstruction of a highly resolved tree of life. *Science*, 2006; **311**(5765):1283–7.
14. von Mering C, Hugenholtz P, Raes J, et al. Quantitative phylogenetic assessment of microbial communities in diverse environments. *Science*, 2007; **315**(5815):1126–1130.
15. Wu M, Eisen JA. A simple, fast, and accurate method of phylogenomic inference. *Genome Biol*, 2008; **9**(10):R151.
16. Schmidt HA, Strimmer K, Vingron M, et al. TREE-PUZZLE: maximum likelihood phylogenetic analysis using quartets and parallel computing. *Bioinformatics*, 2002; **18**(3):502–4.
17. Guindon S, Gascuel O. A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood. *Syst Biol*, 2003; **52**(5):696–704.
18. Krause L, Diaz NN, Goesmann A, et al. Phylogenetic classification of short environmental DNA fragments. *Nucleic Acids Res*, 2008; **36**(7):2230–9.
19. Finn RD, Mistry J, Schuster-Bockler B, et al. Pfam: clans, web tools and services. *Nucleic Acids Res*, 2006; **34**(Database issue):D247–51.
20. Brady A, Salzberg SL. Phymm and PhymmBL: metagenomic phylogenetic classification with interpolated Markov models. *Nat Methods*, 2009; **6**(9):673–6.
21. Bentley SD, Parkhill J. Comparative genomic structure of prokaryotes. *Annu Rev Genet*, 2004; **38**:771–92.
22. Teeling H, Waldmann J, Lombardot T, et al. TETRA: a web-service and a standalone program for the analysis and comparison of tetranucleotide usage patterns in DNA sequences. *BMC Bioinformatics*, 2004; **5**:163.
23. Woyke T, Teeling H, Ivanova NN, et al. Symbiosis insights through metagenomic analysis of a microbial consortium. *Nature*, 2006; **443**(7114):950–5.
24. Chatterji S, Yamazaki I, Bai Z, et al. CompostBin: A DNA composition-based algorithm for binning environmental shotgun reads. In *The 12th Annual International Conference on Research in Computational Molecular Biology, RECOMB 2008*. Springer, Singapore, 2008; 17–28.
25. Diaz NN, Krause L, Goesmann A, et al. TACOA: taxonomic classification of environmental genomic fragments using a kernelized nearest neighbor approach. *BMC Bioinformatics*, 2009; **10**:56.
26. Zhou F, Olman V, Xu Y. Barcodes for genomes and applications. *BMC Bioinformatics*, 2008; **9**:546.
27. Foerstner KU, von Mering C, Hooper SD, et al. Environments shape the nucleotide composition of genomes. *EMBO Rep*, 2005; **6**(12):1208–13.
28. Tyson GW, Chapman J, Hugenholtz P, et al. Community structure and metabolism through reconstruction of microbial genomes from the environment. *Nature*, 2004; **428**(6978):37–43.
29. Lander ES, Waterman MS. Genomic mapping by fingerprinting random clones: a mathematical analysis. *Genomics*, 1988; **2**(3):231–9.
30. Li X, Waterman MS. Estimating the repeat structure and length of DNA sequences using *l*-tuples. *Genome Res*, 2003; **13**(8):1916–22.
31. Sharon I, Pati A, Markowitz VM, et al. A statistical framework for the functional analysis of metagenomes. In *RECOMB 2009*. Springer Berlin / Heidelberg, Tucson, AZ, 2009; 496–511.

32. Richter DC, Ott F, Auch AF, et al. MetaSim: a sequencing simulator for genomics and metagenomics. *PLoS ONE*, 2008; **3**(10):e3373.
33. Huse SM, Huber JA, Morrison HG, et al. Accuracy and quality of massively parallel DNA pyrosequencing. *Genome Biol*, 2007; **8**(7):R143.
34. White JR, Roberts M, Yorke JA, et al. Figaro: a novel statistical method for vector sequence removal. *Bioinformatics*, 2008; **24**(4):462–7.

6 Appendix

Equation (1) is used to calculate the probability that l-tuple w_j (j=1,2,...,W; W is the total number of possible l-tuples) coming from the ith species given its count $n(w_j)$. It is computed by applying Bayes' rule as follows.

$$\Pr\left(w_j \in s_i \mid n(w_j)\right) = \frac{\Pr\left(n(w_j) \mid w_j \in s_i\right)\Pr\left(w_j \in s_i\right)}{\Pr\left(n(w_j)\right)} \tag{8}$$

$$= \frac{\Pr\left(n(w_j) \mid w_j \in s_i\right)\Pr\left(w_j \in s_i\right)}{\sum_{m=1}^{S}\Pr\left(n(w_j) \in s_m \mid w_j \in s_m\right)\Pr\left(w_j \in s_m\right)} \tag{9}$$

$$= \frac{\Pr\left(n(w_j) \mid w_j \in s_i\right)\cdot\frac{l_i}{G}}{\sum_{m=1}^{S}\Pr\left(n(w_j) \in s_m \mid w_j \in s_m\right)\cdot\frac{l_m}{G}} \tag{10}$$

$$= \frac{\frac{\lambda_i^{n(w_j)}e^{-\lambda_i}}{n(w_j)!}\cdot l_i}{\sum_{m=1}^{S}\left(\frac{\lambda_m^{n(w_j)}e^{-\lambda_m}\cdot}{n(w_j)!}l_m\right)} \tag{11}$$

$$= \frac{l_i}{\sum_{m=1}^{S}\left(\left(\frac{\lambda_m}{\lambda_i}\right)^{n(w_j)}\cdot e^{\lambda_i-\lambda_m}\cdot l_m\right)} \tag{12}$$

where $\Pr\left(w_j \in s_i\right) = \frac{l_i}{G}$ is the prior probability that word j is from species i, and G is the total length of genomic sequences contained in the metagenomic dataset. Equation (12) is the result of applying the probability mass function of Poisson distribution into the probability function of equation (10).