Bioinformatics, 32, 2016, i567–i575
doi: 10.1093/bioinformatics/btw466
ECCB 2016

OXFORD

MetaProb: accurate metagenomic reads binning based on probabilistic sequence signatures

Samuele Girotto, Cinzia Pizzi* and Matteo Comin*

Department of Information Engineering, University of Padova, Padova, Italy

*To whom correspondence should be addressed.

Abstract

Motivation: Sequencing technologies allow the sequencing of microbial communities directly from the environment without prior culturing. Taxonomic analysis of microbial communities, a process referred to as binning, is one of the most challenging tasks when analyzing metagenomic reads data. The major problems are the lack of taxonomically related genomes in existing reference databases, the uneven abundance ratio of species and the limitations due to short read lengths and sequencing errors.

Results: MetaProb is a novel assembly-assisted tool for unsupervised metagenomic binning. The novelty of MetaProb derives from solving a few important problems: how to divide reads into groups of independent reads, so that *k*-mer frequencies are not overestimated; how to convert *k*-mer counts into probabilistic sequence signatures, that will correct for variable distribution of *k*-mers, and for unbalanced groups of reads, in order to produce better estimates of the underlying genome statistic; how to estimate the number of species in a dataset. We show that MetaProb is more accurate and efficient than other state-of-the-art tools in binning both short reads datasets (*F*-measure 0.87) and long reads datasets (*F*-measure 0.97) for various abundance ratios. Also, the estimation of the number of species is more accurate than MetaCluster. On a real human stool dataset MetaProb identifies the most predominant species, in line with previous human gut studies.

Availability and Implementation: https://bitbucket.org/samu661/metaprob

Contacts: cinzia.pizzi@dei.unipd.it or comin@dei.unipd.it

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Metagenomics is the study of genomic sequences obtained directly from an environment. Characterizing the taxonomic diversity of microbial communities is one of the primary objectives in metagenomic studies, and it has become an increasingly popular field of study in the past decade (Mande *et al.*, 2012). For example, the diversity of microbes in humans is found to be associated with diseases such as inflammatory bowel disease (IBD) (Qin *et al.*, 2010) and colorectal cancer (Zeller *et al.*, 2014). In this field high-throughput next-generation sequencing (NGS) techniques enable researchers to directly sequence the genomes of multiple species without the need to isolate and culture individual microbes.

The taxonomic analysis of microbial communities is usually carried out by a process referred to as binning, in which reads from the same species are grouped together. By binning reads, researchers can identify the number and the abundance of species in the environment, and further understand what functional roles each species play and how these species work together.

Many computational methods have been developed to classify metagenomic reads. These methods can be broadly classified into two categories. One category is reference-based (supervised), in which one queries reads in reference databases and utilizes the origin of the hit sequences in reference databases to classify reads. Among the most important methods we can recall: Mega (Huson *et al.*, 2007), Kraken (Wood and Salzberg, 2014), Clark (Ounit *et al.*, 2015) and MetaPhlan (Segata *et al.*, 2012). The other category of methods is reference-free (unsupervised), BiMeta (Vinh *et al.*, 2015), MetaCluster (Wang *et al.*, 2012; Yang *et al.*, 2010), AbundanceBin (Wu and Ye, 2011), CompostBin (Chatterji *et al.*, 2008) in which reads are grouped together without the need of reference sequences. These methods are usually based on various definitions of similarity between reads.

Reference-based methods require to index a database of target genomes, e.g. the NCBI/RefSeq databases of bacterial genomes, that is used to classify query reads. These methods are usually very demanding, requiring computing capabilities with large amounts of

RAM and disk space. Yet, query sequences originating from the genomes of most microbes in an environmental sample lack taxonomically related sequences in existing reference databases. Most bacteria found in environmental samples are unknown and cannot be cultured and separated in the laboratory (Eisen, 2007). For these reasons, when using reference-based methods the number of unassigned reads can be very high (Lindgreen *et al.*, 2016). This might indicate that reference-based methods can be of help only when all genomes in the sample are known. Thus, the absence of a taxonomic context makes binning a very challenging task.

On the other hand, reference-free methods do not require to know all the genomes in the sample, but they try to divide the reads into groups so that reads from the same species are clustered together. Reference-free binning tools are based on the observation that the *k*-mer (length-*k* substrings of a fragment) distributions of the DNA fragments from the same genome are more similar than those from different genomes. Thus, without using any reference genome (i.e. unsupervised), one can determine if two fragments are from genomes of similar species based on their *k*-mer distributions. The major problem when processing metagenomic data is the fact that the proportion of species in a sample, a.k.a. abundance rate, can vary greatly. Most of the tools can only handle species with even abundance ratios, and their binning performances degrade significantly in real situations when the abundance ratios of the species are different. To handle uneven abundance ratios some algorithms have been recently developed (Vinh *et al.*, 2015; Wang *et al.*, 2012; Wu and Ye, 2011). For example AbundanceBin (Wu and Ye, 2011) works well for very different abundance ratios, but problems arise when some species have similar abundance ratios. Other tools like BiMeta (Vinh *et al.*, 2015) and MetaCluster (Wang *et al.*, 2012) try to group the reads into many small clusters so that reads from minority species (with low abundance ratios) could exist as isolated clusters. Both these methods use as means of comparison a simple Euclidean distance between the vectors of *k*-mers counts on the groups. However, it has been recently shown that the Euclidean distance of *k*-mers counts tends to be dominated by single-sequence noise and it is not suited for this task (Song *et al.*, 2014). The pairwise comparison of two sequences, or sets of sequences, can be performed with more sophisticated similarity measures, derived from research in alignment-free statistics (Comin *et al.*, 2015; Kantorovitz *et al.*, 2007; Pizzi, 2016; Sims *et al.*, 2009). Following the same paradigm, here we propose a new self-standardized statistic, called probabilistic sequence signature, that is not dominated by the noise in the individual sequences, and that can compare groups of reads with different abundance ratios.

In this paper, we describe a novel assembly-assisted method for metagenomic binning, called MetaProb, that is based on the definitions of independent reads set and of probabilistic sequence signatures. Our contributions can be summarized as follows: (i) the definition of a set of independent reads so that *k*-mers frequencies are not over-counted because of overlapping reads; (ii) the introduction of a novel way to process *k*-mers counts into probabilistic sequence signatures, that will correct for variable distribution of *k*-mers and for unbalanced groups of reads, in order to produce better estimates of the underlying genome statistic; (iii) the proposal of a probabilistic framework that can be easily adapted for different sequencing technologies, in fact Metaprob is suited for current shotgun reads (100 bp), as well as long reads (700 bp or above), as opposed to most methods; (iv) a novel and effective estimation of the number of species in a sample based on probabilistic sequence signatures.

We performed experiments on synthetic and real datasets, and compared MetaProb with popular tools: AbundanceBin (Wu and Ye, 2011), BiMeta (Vinh *et al.*, 2015), MetaCluster (Wang *et al.*, 2012). MetaProb outperforms the other methods in its ability to correctly identify the species and their abundance levels.

2 Method: MetaProb

The composition of DNA, in terms of its constituent *k*-mers, is known to be a feature of the genome. A number of studies (Chor *et al.*, 2009; Huson *et al.*, 2007; Ounit *et al.*, 2015; Wang *et al.*, 2012) are based on the assumption that the *k*-mer frequency distributions of long fragments or whole genome sequences are unique to each genome. However, most sequencing technologies cannot produce long fragments and thus these compositional distances cannot be directly applied. In order to solve this issue, and to mimic the availability of long fragments, MetaProb addresses the problem of metagenomic binning in two phases. Figure 1 shows the processing pipeline of MetaProb. We will now describe the main steps of the processing, giving a brief explanation of the reasons why they were undertaken. In the following subsections each step will be described in details.

In Phase 1 reads are grouped together based on the extent of their overlap. This is measured in terms of shared *q*-mers, a technique widely used in de-novo assembly. As a result the reads in a group, because of their overlap, are likely to belong to the same species. However, reads from a same species might be distributed in different groups. As our final aim is to group together all the reads from a same species, further processing is needed to cluster the groups obtained in Phase 1 based on their similarity.

The similarity between groups can be defined in terms of *k*-mers frequency distribution within each group. However, by construction, the reads in a group must have a significant overlap. Because such overlaps might artificially inflate the count of some *k*-mers, we developed a strategy, based on independent sets of a graph, to select a subset of reads from a group in order to reduce the redundancy provided by large overlaps.

When entering Phase 2, each group is then represented by a set of independent reads on which the *k*-mers frequency distribution is computed. Because the straightforward application of the Euclidean distance to pairs of vectors representing *k*-mers distributions of different groups can be biased by the stochastic noise in each sequence (Lippert *et al.*, 2002; Song *et al.*, 2014), and by the possibly unbalanced size of the groups, we propose here a novel similarity measure based on self-standardized probabilistic sequence signatures that accounts for these issues.

The final step of MetaProb consists in the clustering of groups based on their signatures with the *k*-means algorithm. This algorithm requires in input the number of clusters, that in our case coincides with the number of species in the input dataset, a knowledge that in many real metagenomic samples is unknown. To address such cases MetaProb will use a novel estimator for the number of species that will provide the input parameter for the *k*-means algorithm.

2.1 Phase 1: merging reads into groups

In Phase 1, each read is considered as a group, and groups are progressively merged until a stopping criteria is met. Two groups are merged if they share at least *m* common *q*-mers. This is one of the most efficient way to measure the sequence overlap information between reads, and it has been used in a number of

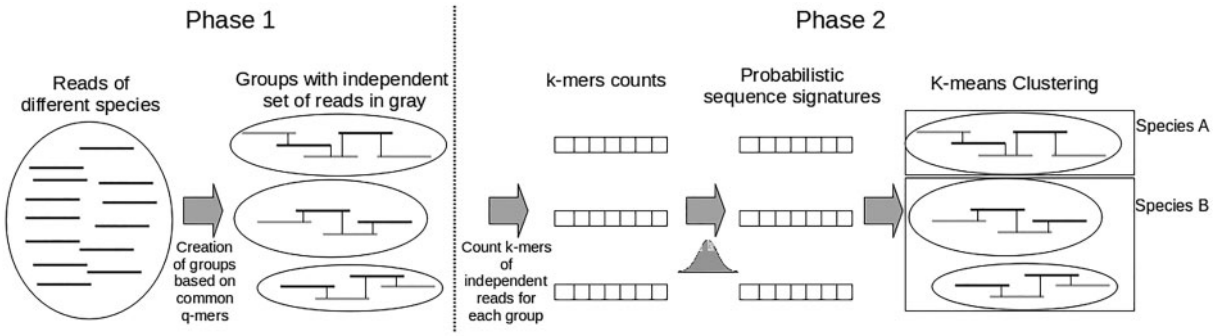

Fig. 1. Binning process of MetaProb. Phase 1 groups overlapping reads into groups. Phase 2 builds the probabilistic sequence signatures of independent reads and merges the groups into clusters

studies (Vinh *et al.*, 2015; Wang *et al.*, 2012; Yang *et al.*, 2010). These methods are based on the assumption that most q-mers are not shared by different genomes when q is sufficiently large. For example, as reported in (Vinh *et al.*, 2015), on 100 pairs of bacterial genomes the average ratio of common q-mers between the genomes is less than 1.02% when $q > 30$. Thus there is great probability that the reads having common q-mers, with q sufficiently large, are overlapping reads.

In this step, we indirectly build a graph of adjacent reads, based on the above criteria. However, since in the second phase we need to compute the distribution of k-mers, to avoid the over-counting introduced by overlaps, here we need to identify within each group a set of reads that minimizes this bias. Given the reads in a group, it is possible to consider the subset of reads that do not overlap with each others. If we consider a group as a graph of reads, we can use the well-known definition of independent set or stable set on graphs. An independent set defined on a graph is a set of vertices which does not contain adjacent vertices. Unfortunately the maximum independent set problem is known to be NP-hard, so we need to explore a tractable solution. The identification of the maximum independent set of reads, $I(G)$, can be performed on-line while computing group G. In the proposed algorithm groups are extended in a greedy fashion by considering first the reads x, $x \notin G$, with the largest number of common q-mers with some reads in G. If the read x is not adjacent to any read in $I(G)$, then this is a new independent read and we add x to $I(G)$.

The effect of sequencing errors and of erroneous q-mers may lead to include in a group reads from different organisms. In some studies, prior to the assembly step, a q-mers correction can be applied (Kelley *et al.*, 2010). Instead, to reduce the likelihood to insert in a group a read not belonging to that species we limit the size of groups by imposing a threshold T. Groups are extended until the size of $I(G)$, computed as the sum of lengths of the reads in $I(G)$, does not exceed the threshold T. Finally, when all groups are created, the probabilistic sequence signature of each group will be calculated based on the sets of independent reads $I(G)$.

2.2 Phase 2: probabilistic sequence signatures

Once the groups are constructed we need to define a suitable distance measure to compare and cluster groups into candidate species. We recall that the simple Euclidean distance between k-mer frequency distributions, used by almost all methods, can be dominated by single-sequence noise (Lippert *et al.*, 2002; Song *et al.*, 2014). To address this issue a number of sequence signatures, based on k-mer counts statistics, have been proposed (Apostolico *et al.*, 2016; Comin *et al.*, 2015; Fernandes *et al.*, 2009; Kantorovitz *et al.*, 2007; Pizzi, 2016; Sims *et al.*, 2009).

Sequence signatures, a.k.a. alignment-free statistics, are receiving increasing attention because they are computationally efficient and can provide attractive alternatives when alignment-based approaches are unfeasible. For example, alignment-free techniques proved to be very efficient in the study of evolution of organisms based on whole genomes analysis (Apostolico and Denas, 2008; Apostolico *et al.*, 2014; Comin and Verzotto, 2012; Pizzi, 2016; Sims *et al.*, 2009; Ulitsky *et al.*, 2006). Some alignment-free measures use the patterns distribution to study the identification of cis-regulatory modules (CRM) (Apostolico *et al.*, 2011; Comin and Verzotto, 2014; Kantorovitz *et al.*, 2007; Parida *et al.*, 2014b) and also of entropic profiles (Comin and Antonello, 2013, 2014, 2016; Fernandes *et al.*, 2009; Parida *et al.*, 2014a).

Inspired by the recent developments in the field of alignment-free statistics we propose here a novel similarity measure based on probabilistic sequence signatures for the comparison of groups of reads. The idea is to account for the different distribution of k-mers counts and to remove the bias of unbalanced groups in a probabilistic framework with a self-standardized statistic.

Let us define a read X^i as a sequence of characters from the alphabet $\Sigma = \{A, C, G, T\}$, with $i = 1..|M|$, where M is our input set of metagenomic reads. We call X^i_w the frequency of the k-mer w in the read X^i. Given that reads are sequenced from both strands of a genome, X^i_w will include also the contribution of the reversed complement of w. We can consider the variables X^i_w as Bernoulli, if the length of k-mers is smaller w.r.t. the length of reads, $k << |X^i|$. Similarly to the other methods (Vinh *et al.*, 2015; Wang *et al.*, 2012), we will use $k = 4$, thus this approximation holds. Given a group of independent reads $I(G)$, computed from the previous step, we can define the variable X^G_w, that represents the number of times the k-mer w appears in the group $I(G)$: $X^G_w = \sum_{i=1}^{g} X^i_w$, where g is the number of independent reads in the group.

To account for the different probability of appearance of k-mers, the variables X^G_w need to be standardized. If we define the probability of a k-mer w to appear in the group $I(G)$ as P^G_w, and we recall that X^i_w is a Bernoulli, we can compute mean and variance of X^G_w as:

$$E[X^G_w] = \mu^G_w = P^G_w \sum_{i=1}^{g} (|X^i| - k + 1) = P^G_w |G| \quad (1)$$

$$Var(X^G_w) = (\sigma^G_w)^2 = P^G_w (1 - P^G_w) |G| \quad (2)$$

where $|G| = \sum_{i=1}^{g} (|X^i| - k + 1)$. Thus the variable X^G_w can be standardized as follows:

$$\tilde{X}_w^G = \frac{X_w^G - \mu_w^G}{\sigma_w^G} \quad (3)$$

As already observed the frequency of k-mers in different genomes can greatly vary. Similarly, it is difficult to estimate the probability P_w^G, as it does not follow the same model for different genomes. Thus we need to estimate P_w^G from the set of reads in input.

Long Read: For long reads datasets, the size of groups can be sufficiently large to be able to estimate the probability of P_w^G, for every group G. We define n_b^G, with $b \in \{A, C, G, T\}$, as the number of times the nucleotide b occurs within the group G, and the probability of the symbol b in the group G is:

$$p_b^G = \frac{n_b^G}{\sum_{i \in G} |X^i|} \quad (4)$$

Finally, if we consider the symbols independent and equally distributed within a group, the probability P_w^G can be computed as the product $P_w^G = p_{w_1}^G * p_{w_2}^G * p_{w_3}^G * p_{w_4}^G$, for a k-mer $w = w_1 w_2 w_3 w_4$.

Short Reads: For short reads datasets, the size of groups cannot be large enough to have good estimate. To this end we devise a different way to estimate these probabilities, independently from the groups, so that $P_w^G = P_w$, for all groups. We compute the distribution of all k-mers in the input dataset, by scanning all independent reads in the collection M. Thus, for short reads $P_w^G = P_w = \frac{X_w^M}{|M|}$.

Probabilistic sequence signature: The abundance ratios of species in a real metagenomic sample can be highly unbalanced. This may produce groups of different sizes. In order to be able to cluster them we need to remove this bias. Thus, we normalize the vector $\tilde{X}_w^G$ so that its module is unitary, and define the probabilistic sequence signature of a group G as:

$$f_w^G = \frac{\tilde{X}_w^G}{\sqrt[2]{\sum_{w \in \Sigma^k} (\tilde{X}_w^G)^2}} \quad (5)$$

The vectors of probabilistic sequence signatures, f_w^G, are used to compare two groups by means of their correlation. The standard k-means clustering is applied to these vectors until the groups are merged into C clusters, where C is either given or estimated, as will discuss in the next section.

2.3 Estimation of the species number

K-means is an algorithm that groups a set of data into C clusters. As many other similar algorithms, it requires that the number of clusters C must be known in advance. The automatic estimation of the number of clusters C is a very hard problem and it is made more difficult when the data has many dimensions, even when clusters are well-separated. Most methods that seek to estimate the number of species, like MetaCluster (Wang *et al.*, 2012), use often prior knowledge, other assumptions, or practical experience. One popular algorithm for this problem is G-means (Hamerly and Elkan, 2003), which is based on a statistical test for the hypothesis that a subset of data follows a Gaussian distribution. Unfortunately this test does not work well when applied to genomic data, because the assumption that the reads form clusters that are distributed as a Gaussian can be unrealistic. Instead, we develop an novel method to estimate the number of species, inspired by G-means, but with a different statistical test. G-means uses the Anderson-Darling statistic, that is meant to test the normality of the data, whereas we use the two-sample Kolmogorov–Smirnov test. We applied this test on two vectors: one vector is the cumulative distribution function (CDF) computed on the data, normalized with mean 0 and variance 1; the other vector is the expected cumulative distribution function (ECDF) of the same data. We called this estimator *SpeciesNumber*.

In general G-means runs k-means increasing the number of clusters C in a hierarchical fashion until the test accepts the hypothesis that the data assigned to each k-means center are Gaussian, however for every iteration k-means is run on the entire dataset. Instead, our estimator SpeciesNumber at each iteration saves the clusters that passes the Kolmogorov–Smirnov test, and removes the corresponding reads, so that future iterations will no longer consider those clusters. If a cluster does not pass the test, then it is a good candidate to be splitted. The process is repeated until all data pass the test and consequently there are no more clusters to be created.

The SpeciesNumber algorithm starts with a small number of k-means centers, and grows the number of centers. At each iteration the algorithm can do only one of the followings two steps: 1) splits one center into two, only for those centers whose data does not pass the Kolmogorov–Smirnov test; or 2) saves those centers whose data pass the test and eliminate the reads that belong to these clusters from the input. After each round of splitting, we run k-means on the remaining reads and all the remaining centers to refine the current solution. The initial number of species can be initialized to $C = 1$, or we can choose a larger value of C if we have some prior knowledge about its range. The input parameters of the SpeciesNumber estimator are the set of reads and a confidence level a for which we used the standard value of 0.95.

3 Results and discussion

In this section we discuss the results of the comparison between MetaProb and several other state-of-the-art reference-free binning algorithms: MetaCluster 5.0.1 (Yang *et al.*, 2010), AbundanceBin (Wu and Ye, 2011) and BiMeta (Vinh *et al.*, 2015).

3.1 Datasets description

We considered 28 different datasets: 25 of these datasets were already used in (Vinh *et al.*, 2015), and include several simulated bacterial metagenomes built with MetaSim (Richter *et al.*, 2008); 2

Algorithm 1: Pseudocode of the SpeciesNumber estimator.

Input: Set of reads X, confidence level a
Output: C clusters SpeciesNumber(X, a)
1. Let C_{save} be the clusters saved at each iteration;
2. Let C be the initial set of centers;
3. C = kmeans(C, X);
4. Let $\{x_i | class(x_i) = j\}$ be the set of data-points assigned to the center c_j;
5. Compute the CDF(c_j) vector for each cluster c_j;
6. Use the Kolmogorov–Smirnov statistical test, on CDF(c_j), to detect if each $\{x_i | class(x_i) = j\}$ follows the expected distribution (ECDF) (at confidence level a);
7. If the cluster c_j pass the test, save c_j in C_{save} and update $X = X/\{x_i | class(x_i) = j\}$ and $C = C/\{c_j\}$. Otherwise replace c_j with two clusters;
8. Repeat from step 3 until no more centers are added;
9. Return C_{save};

datasets contain synthetic metagenomes based on real reads; and one real metagenomic sample from the Human Microbiome Project.

The set of 25 datasets used in (Vinh *et al.*, 2015) can be partitioned in three groups: *S*, *L* and *R*, depending on their main characteristics. Each dataset in *S* or *L* comprises paired-end short reads (length of approximately 80 bp) generated according to the Illumina error profile with an error rate of 1%. The datasets in *L* are built over the genomes of two species, *Eubacterium eligens* and *Lactobacillus amylovorus*. Such datasets are used to evaluate binning algorithms on set of reads with different abundance ratio between the two species. The datasets in *S* are much more varied in terms of number of species (up to 30), abundance ratio (balanced/unbalanced), and phylogenetic distance. The datasets in *R* contain Roche 454 single-end long reads of length approximately 700 bp, and sequencing error rate of 1%.

We also include in our tests two metagenomes that are constructed from real sequencing data. We use the dataset of short-reads Illumina MiSeq from Kraken (Wood and Salzberg, 2014), that is composed of 10 genomes with two abundance profiles. The MiSeq metagenomes were built using 10 sets of bacterial whole-genome shotgun reads. These reads were found either as part of the GAGE-B project (Magoc *et al.*, 2013) or in the NCBI Sequence Read Archive. A summary of the metagenomes can be found in Supplementary Material, Table 1 for short reads, and in Table 2 for long-reads.

Finally, MetaProb was tested also on a real metagenomic sample of Human feces from the Human Microbiome Project (SRR1804065).

3.2 Performance evaluation metrics

Precision, Recall and F-measure metrics are used to compare the performances of the binning algorithms under examination. Precision measures the ability of the approach to build clusters composed by reads coming from a same species. On the other hand, recall measures the ability to cluster together all the reads of a given species. Therefore, when evaluating the performances of a binning algorithm one should take into account both these aspects. A common way of doing so is through the F-measure, i.e. the harmonic mean of precision and recall.

Let n be the number of species in a metagenomic dataset, and C be the number of clusters returned by the algorithm. Let A_{ij} be the number of reads from species j assigned to cluster i. Following the definitions in (Vinh *et al.*, 2015) we have:

$$\text{Precision} = \frac{\sum_{i=1}^{C} \max_j A_{ij}}{\sum_{i=1}^{C}\sum_{j=1}^{n} A_{ij}} \tag{6}$$

$$\text{Recall} = \frac{\sum_{j=1}^{n} \max_i A_{ij}}{\sum_{i=1}^{C}\sum_{j=1}^{n} A_{ij} + \#\text{unassigned\_reads}} \tag{7}$$

$$F\text{-measure} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}} \tag{8}$$

The input parameters of MetaProb are: the length k used for the probabilistic sequence signatures, the values of q, m and T used in the phase 1, and the number of clusters C (optional). For fairness we used the same parameters values for all the approaches under comparison. In particular, the length of the k-mers is set to 4, and the number of species C is given explicitly in input. The choice of the best k to compare genomic sequences has been extensively studied in the context of metagenomic binning (Chatterji *et al.*, 2008; Vinh *et al.*, 2015; Wu and Ye, 2011; Yang *et al.*, 2010), as well as in a number of different applications (Comin *et al.*, 2015; Kantorovitz *et al.*, 2007; Sims *et al.*, 2009). In all these different contexts the best performances are obtained with $k=4$ or $k=5$. After a series of test we choose $k=4$ as default value. While MetaCluster 5 and AbundanceBin do not need further parameters, both BiMeta and our MetaProb require in input the length q of the q-mers, and the minimum threshold m of shared q-mers needed to detect reads overlaps in phase 1. These parameters were set to $q=30$, and $m=5$ (short reads) or $m=45$ (long reads), similarly to (Vinh *et al.*, 2015). The parameter T that limits the size of groups in phase 1 was set to 9000. Due to its characteristics MetaCluster 5 could be tested only on the short read datasets.

To better evaluate the performance of MetaProb, the discussion of the experimental results is done separately for of short paired-end read datasets and for long single-end read datasets.

3.3 Results on short paired-end reads

These sets of experiments considered all four binning approaches (our MetaProb, BiMeta, AbundanceBin and MetaCluster) and all the dataset in *S* and *L*, and the two MiSeq datasets.

Table 1 shows the results of the comparison in terms of F-measure. MetaProb has higher F-measure for 11 out of 18 datasets. In the other 7 cases, in which the F-measure of BiMeta is higher than the F-measure of MetaProb, the difference between the two is relatively low (0.04 on average). Moreover, MetaProb has the best performance on average when we consider all the datasets.

Table 1 also shows that the datasets from *S*7 to *S*10_S are among the most difficult to analyze by all the tested tools. These datasets are characterized by the presence of several species (up to 30), and by an unbalanced abundance ratio. Despite this, MetaProb is still competitive in the classification.

In general, the high values of MetaProb in terms of F-measure derive from both balanced and high values of recall and precision in all datasets, although they are not necessarily always the highest values for each individual dataset. Figures 2 and 3 show the details of

Table 1. F-Measure on short paired-end read datasets

F-Measure	Abundance Bin	MetaCluster	BiMeta	MetaProb
S1	0.683	0.672	0.978	**0.991**
S2	0.713	0.631	0.581	**0.901**
S3	0.824	0.415	**0.978**	0.928
S4	0.883	0.460	**0.994**	0.908
S5	0.552	0.643	0.690	**0.832**
S6	0.692	0.492	0.858	**0.970**
S7	0.606	0.652	**0.843**	0.782
S8	0.528	0.529	0.743	**0.769**
S9	Error	0.639	**0.791**	0.719
S10_S	0.137	0.052	0.429	**0.495**
L1	0.625	0.549	0.980	**0.984**
L2	0.793	0.675	0.980	**0.992**
L3	0.900	0.667	0.986	**0.993**
L4	0.959	0.703	**0.987**	0.986
L5	0.977	0.612	**0.991**	0.983
L6	0.984	0.649	**0.990**	0.984
MiSeq_a1	0.534	0.555	0.645	**0.737**
MiSeq_a2	0.496	0.638	0.667	**0.670**
Average	0.699	0.568	0.840	**0.868**

Best results are in bold.

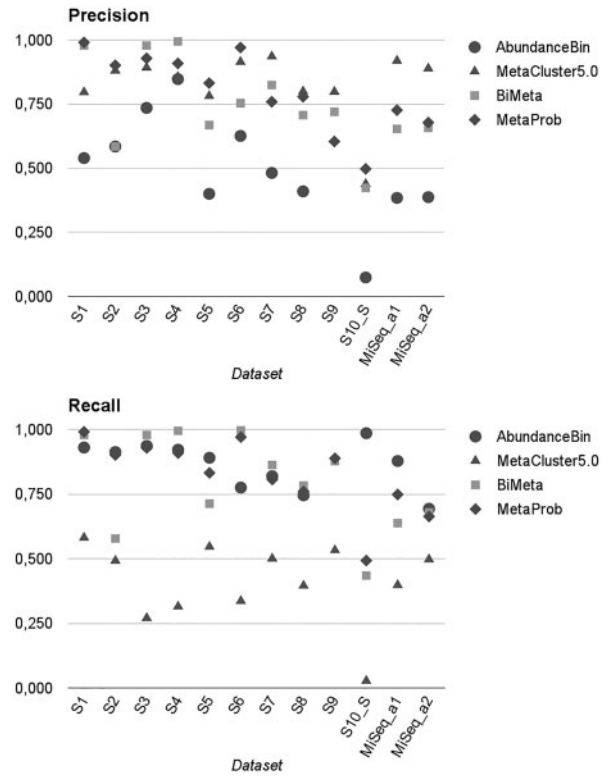

Fig. 2. Precision and recall of various binning algorithms on short paired-end reads datasets (*S* and MiSeq). These datasets are varied in terms of number of species and abundance ratios

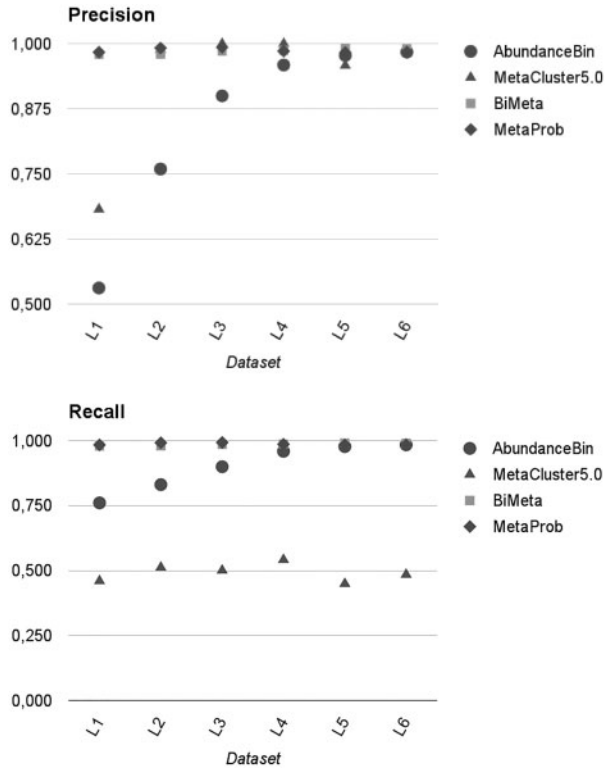

Fig. 3. Precision and recall of various binning algorithms on short paired-end reads datasets. The datasets *L* are highly unbalanced

precision and recall on short-reads datasets for all the approaches under comparison.

MetaCluster is characterized by a high precision in many datasets, but also by a low recall that negatively influences the F-measure. The performances of AbundanceBin are somewhat complementary to those of MetaCluster, as they are characterized by high values of recall, coupled with very low precision in many cases. In summary, MetaCluster and AbundanceBin showed unbalanced performances in terms of precision and recall, having the former approach its strength in precision, and the latter approach its strength in recall. Nevertheless, MetaProb showed a precision higher than MetaCluster in 11 out of 18 datasets, and a recall higher than AbundanceBin in 9 out of 18 datasets.

The behavior of BiMeta is closer to the behavior of MetaProb, showing high values in both precision and recall. However, there are significant differences in terms of precision in the analysis of the datasets *S2*, *S5*, *S6*. This difference can be explained by the use made by MetaProb of statistical standardization, which is capable to best describe the model and properly separate the reads.

While the performances on the datasets *S* are quite varied, Figure 3 shows MetaProb and BiMeta as clear winners for the datasets *L* in terms of both precision and recall, even when considered separately. The results show that MetaProb and BiMeta not only have the highest performances, but are also stable for different ratios of species abundances.

The most interesting short-reads datasets are the MiSeq datasets (see Table 1 and Fig. 2). These are the more realistic metagenomes, composed by mixing real reads from individual genomes. On this difficult test the results of MetaProb, in terms of F-measure, improve over all other tools. Thus, even for these realistic datasets, the performance are consistent and similar with the most difficult simulated datasets.

In summary for short-read data the presence of several species, and the variety in the phylogenetic distance affect the performances of all the tools under analysis. Nevertheless, MetaProb achieved the best performances in terms of average F-measure.

3.4 Result on long single-end reads

Table 2 reports the results of the experimental comparison among MetaProb, BiMeta and AbundanceBin on long reads datasets. MetaProb is the best algorithm in 8 out of 9 cases, and very close to AbundanceBin in the remaining case. Its average F-measure is very high (0.968), 10% higher than BiMeta and 25% higher than AbundanceBin. The only case (*R9*) in which the value of F-measure is under 90% is due to a low precision caused by a unbalanced dataset with several species and different kind of phylogenetic distance, as we can see in short read datasets.

Table 2. F-measure on long single-end read datasets *R*

F-Measure	Abundance Bin	BiMeta	MetaProb
R1	0.674	0.609	**0.971**
R2	0.667	0.773	**0.968**
R3	0.672	0.780	**0.928**
R4	0.686	0.992	**0.993**
R5	0.709	0.988	**0.998**
R6	0.761	0.953	**0.994**
R7	0.950	0.890	**0.986**
R8	0.926	0.980	**0.994**
R9	**0.891**	0.860	0.881
Average	*0.771*	*0.870*	***0.968***

Best results are in bold.

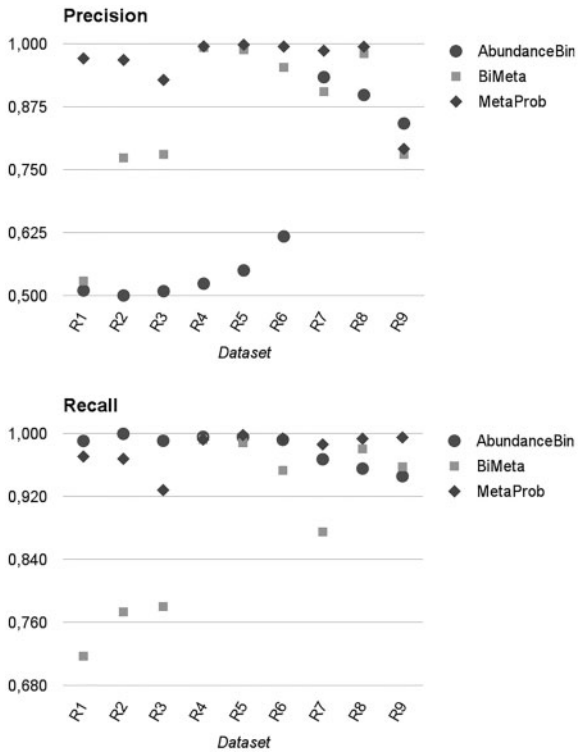

Fig. 4. Precision and recall of various binning algorithms on long single-end reads datasets

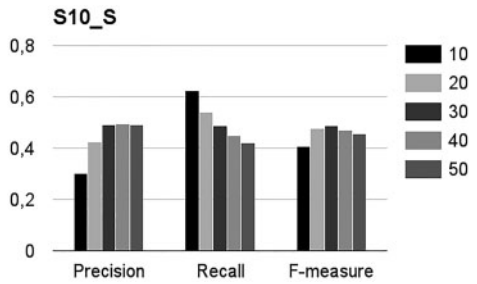

Fig. 5. Precision, recall and F-measure on S10_S dataset as a function of the number of species *C* of *k*-means

Table 3. Estimating the number of species *C*: comparison between MetaProb and MetaCluster. The best results are in bold

MetaCluster	Precision	Recall	F-measure	C est.	C real
S7	0.9256	0.6711	0.7781	9	5
S8	0.7363	0.3533	0.4775	35	5
S9	0.8255	0.4519	0.5841	133	15
S10_S	0.4064	0.0278	0.0521	16	30
MiSeq_a1	0.8828	0.4909	0.6309	16	10
MiSeq_a2	0.8385	0.3926	0.5348	**14**	10
MetaProb	**Precision**	**Recall**	**F-measure**	C est.	C real
S7	0.8179	0.7453	0.7799	**6**	5
S8	0.7393	0.7536	0.7464	**6**	5
S9	0.8723	0.8168	0.8437	**31**	15
S10_S	0.4321	0.5142	0.4696	**31**	30
MiSeq_a1	0.6598	0.6580	0.6589	**10**	10
MiSeq_a2	0.7678	0.6206	0.6864	15	10

Figure 4 shows the detailed precision and recall for all the datasets R. Clearly MetaProb outperforms the competitors in terms of precision, and in most cases also in terms of recall. Moreover, the performances are quite stable for both these metrics. These results show how the statistical standardization performed by MetaProb has an even higher impact in the performances when long reads are analyzed.

3.5 Estimating the number of species

The estimation of the number of species C is in general a very difficult task. To have a complete overview of the performance of MetaProb, when the number of species is unknown, we devised two tests. First of all we evaluated the variations of precision, recall and F-measure as a function of the number of species *C* of *k*-means. In a second test we compared the SpeciesNumber estimator, described in Section 2.3, and the estimator of MetaCluster 5, that tries to solve the same problem.

The first analysis is necessary to evaluate how the performances of MetaProb are affected when an incorrect number of species is used. The data reported refer to the dataset with the largest number of species *S*10_S, that contains 30 species.

In Figure 5 we report all the performance metrics while varying the number of species from 10 to 50. We observe that if the number of species increases, the precision improves while the recall decreases. On the other hand if the number of species is underestimated the recall improves and the precision worsens. The F-measure follows the major variation between precision and recall, however its peak is at the correct value of $C = 30$, see Figure 5. From this study we can see that, if we use a number of species C in the broad range [20,50], without knowing its correct value, still we have an F-measure that is comparable with the best F-measure obtained for the correct value of *C*.

In the second test we compare the estimation of number of species of MetaProb and MetaCluster on the datasets with the largest number of reads and largest number of species that best represents the real metagenomes. As we can see in Table 3 both MetaProb and MetaCluster tend to overestimate the correct number of species *C*, except for *S*10_*S* dataset in which MetaCluster underestimates the value of *C*. However the prediction of MetaProb overestimates the number of species *C* just by a small amount compared to MetaCluster. For example MetaCluster predicts twice the number of clusters for *S*7, 7 times more species for *S*8 and of about 9 times more clusters for *S*9. Instead MetaProb, on the same datasets, overestimates only by 1 cluster in *S*7 and *S*8 and 2 times more clusters in *S*9. In *S*10_*S* MetaCluster underestimates the number of clusters from 30 to 16, instead the prediction of MetaProb was 31, very close to the actual value. In the most realistic metagenome MiSeq_a1, MetaProb estimates the correct number of clusters, whereas MetaCluster overestimates this value. Both methods tend to overestimate the number of species for the dataset MiSeq_a2, however the predictions are not too far from the correct value. If we consider the F-measures we can see that MetaProb has the highest scores for all datasets, and that these values are comparable with the best F-measures obtained when the number of species is known.

3.6 Results on real metagenome

We also ran our tool using a real metagenome dataset. We chose one DNA stool samples of a female from the Human Microbiome Project (SRR1804065), generated using Illumina, also used in (Sobih *et al.*, 2016). The average read length is 100 bp and the total number of reads from the sample was 43 747 562. This time,

however, the 'ground truth' was not available. Solely with the purpose to evaluate our method on real data, as many other studies, we use BLAST to map the reads against all bacterial genomes, and filter out the reads that do not map to any genome. If two paired-end reads do not map on the same genome we discard them. After this filter 2 531 376 reads can be mapped to one or more species, with a sequence identity of 95%. We run MetaProb on this dataset with default parameters, along with the estimation of the number of clusters. MetaProb reports 9 clusters of various sizes. In Table 4 we show the resulting clusters in order of size. For each cluster we report the majority species, the precision of the cluster, the abundance rate of the cluster, and the abundance rate of the majority species. The cluster abundance rate is computed as the size of clusters divided by the total number of reads. The abundance rate of the majority species is the number of reads, from a given cluster, assigned to the majority species divided by the total number of reads. The most abundant species is *Bacteroides vulgatus*, with a total abundance of 51%, which was also reported as the most abundant species in human feces (Qin *et al.*, 2010). Other abundant species like *Parabacteroides distasonis*, *Faecalibacterium prausnitzii* and *Bacteroides salanitroni* can also be detected with a relative high precision (greater than 65%). These bacteria are also among the most abundant species in stool samples (Qin *et al.*, 2010). Other bacteria associated with feces are also discovered, like *Bacteroides thetaiotaomicron* and *Odoribacter splanchnicus*, but with a low precision. If we further analyze the clusters without a strong majority species, like clusters 5 and 9, we found that the majority of reads belong to the same family. However, this study does not cover the low abundance species. A possible setup to expand the taxonomy annotation of all species, is to filter out the most abundant species and rerun MetaProb on the remaining reads.

3.7 Time performances

Besides the quality of clustering, we assessed also the time performances of all the tools. All the experiments were performed on a laptop equipped with Intel core i7-4510U CPU @ 2.00G Hz and 16 GB of RAM.

Table 4. Experiment on real fecal metagenome

Cluster	Majority species	Precision	Cluster Abund.	Species Abund.
1	Bacteroides vulgatus	80%	49.1%	39.1%
2	Bacteroides vulgatus	56%	22.1%	12.4%
3	Parabacteroides distasonis	66%	7.2%	4.8%
4	Bacteroides salanitronis	65%	6.9%	4.5%
5	Bacteroides thetaiotaomicron	28%	4.9%	1.4%
6	Parabacteroides distasonis	40%	3.4%	1.3%
7	Faecalibacterium prausnitzii	77%	2.8%	2.2%
8	Odoribacter splanchnicus	53%	2.4%	1.3%
9	Parabacteroides distasonis	33%	1.2%	0.4%

Best results are in bold.

Table 5. Average running time on short and long read datasets

Average Time	AbundanceBin	MetaCluster	BiMeta	MetaProb
Short Read	2161.772	**144.715**	1047.715	164.585
Long Read	1969.560	–	1652.692	**286.042**

Best results are in bold.

Metaprob is implemented in C++, and exploits multithreading (as MetaCluster). Table 5 shows the average time for the analysis of short and long reads datasets. MetaProb is an order of magnitude faster than AbundanceBin and BiMeta both on short and long reads. On short reads MetaCluster is the fastest algorithms, but the performances of MetaProb are comparable. In details, MetaProb is actually faster than MetaCluster on 12 out of 18 short read datasets. (data shown in the Supplementary Material, Tables 3 and 4).

4 Conclusion and future work

Binning metagenomics reads remains a crucial step in metagenomics analysis. In this work we presented MetaProb, an assembly-assisted approach for reference-free metagenomic binning. Our approach can deal with short and long reads in a novel probabilistic framework, by using probabilistic sequence signatures. We compared the binning performances over several short and long reads datasets against other state-of-art binning algorithms, showing that MetaProb achieves in most cases the best performances in terms of F-measure. The estimation of the number of species in a metagenomic sample can be performed with MetaProb, adding a degree of freedom in the analysis. On a real fecal metagenomic data MetaProb was able to detect the most abundant species with high precision. MetaProb is also much faster than AbundanceBin and BiMeta, and it has time performances comparable to those of MetaCluster 5. In the future we plan to extend the features of MetaProb with the ability to annotate each read with the taxonomy.

Funding

This work was partially supported by the Italian MIUR project PRIN20122F87B2.

Conflict of Interest: none declared.

References

Apostolico,A. and Denas,O. (2008) Fast algorithms for computing sequence distances by exhaustive substring composition. *Algorithms Mol. Biol.*, **3**, 13.

Apostolico,A. *et al.* (2011) Efficient algorithms for the discovery of gapped factors. *Algorithms Mol. Biol.*, **6**, 1–10.

Apostolico,A. *et al.* (2014). Alignment free sequence similarity with bounded hamming distance. In: *Proceedings of Data Compression Conference*, DCC'14. IEEE, pp. 183–192.

Apostolico,A. *et al.* (2016) Sequence similarity measures based on bounded hamming distance. *Theor. Comput. Sci.*, **638**, 76–90.

Chatterji,S. *et al.* (2008). Research in Computational Molecular Biology: 12th Annual International Conference, RECOMB 2008, Singapore, March 30–April 2, 2008. In: *Proceedings, chapter CompostBin: A DNA Composition-Based Algorithm for Binning Environmental Shotgun Reads*. Springer, Berlin, Heidelberg, pp. 17–28.

Chor,B. *et al.* (2009) Genomic DNA k-mer spectra: models and modalities. *Genome Biol.*, **10**, R108.

Comin,M. and Antonello,M. (2013) *Fast Computation of Entropic Profiles for the Detection of Conservation in Genomes*. Springer, Berlin, Heidelberg, pp. 277–288.

Comin,M. and Antonello,M. (2014) Fast entropic profiler: an information theoretic approach for the discovery of patterns in genomes. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **11**, 500–509.

Comin,M. and Antonello,M. (2016) On the comparison of regulatory sequences with multiple resolution entropic profiles. *BMC Bioinformatics*, **17**, 1–12.

Comin,M. and Verzotto,D. (2012) Whole-genome phylogeny by virtue of unic subwords. In: *2012 23rd International Workshop on Database and Expert Systems Applications (DEXA)*, pp. 190–194.

Comin,M. and Verzotto,D. (2014) Beyond fixed-resolution alignment-free measures for mammalian enhancers sequence comparison. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, **11**, 628–637.

Comin,M. *et al.* (2015) Clustering of reads with alignment-free measures and quality values. *Algorithms Mol. Biol.*, **10**, 4.

Eisen,J.A. (2007) Environmental shotgun sequencing: its potential and challenges for studying the hidden world of microbes. *PLoS Biol*, **5**, e82.

Fernandes,F. *et al.* (2009) Entropic profiler – detection of conservation in genomes using information theory. *BMC Res. Notes*, **2**, 1–8.

Hamerly,G. and Elkan,C. (2003) Learning the K in K-Means. In: *Advances in Neural Information Processing Systems 16 (NIPS)*, pp. 281–288.

Huson,D.H. *et al.* (2007) Megan analysis of metagenomic data. *Genome Res.*, **17**, 377–386.

Kantorovitz,M.R. *et al.* (2007) A statistical method for alignment-free comparison of regulatory sequences. *Bioinformatics*, **23**,

Kelley,D.R. *et al.* (2010) Quake: quality-aware detection and correction of sequencing errors. *Genome Biol.*, **11**, 1–13.

Lindgreen,S. *et al.* (2016) An Evaluation of the Accuracy and Speed of Metagenome Analysis Tools. *Scientific Reports* **6**, 19233.

Lippert,R.A. *et al.* (2002) Distributional regimes for the number of k-word matches between two random sequences. *PNAS*, **99**, 13980–13989.

Magoc,T. *et al.* (2013) Gage-b: an evaluation of genome assemblers for bacterial organisms. *Bioinformatics*, **29**, 1718–1725.

Mande,S.S. *et al.* (2012) Classification of metagenomic sequences: methods and challenges. *Brief. Bioinf.*, **13**, 669–681.

Ounit,R. *et al.* (2015) Clark: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers. *BMC Genomics*, **16**, 1–13.

Parida,L. *et al.* (2014a) Entropic profiles, maximal motifs and the discovery of significant repetitions in genomic sequences. In: Brown D. and Morgenstern (B.eds.) *Algorithms in Bioinformatics, Volume 8701 of Lecture Notes in Computer Science*. Springer, Berlin, Heidelberg, pp. 148–160.

Parida,L. *et al.* (2014b) Irredundant tandem motifs. *Theor. Comput. Sci.*, **525**, 89–102. (Advances in Stringology).

Pizzi,C. (2016) Missmax: alignment-free sequence comparison with mismatches through filtering and heuristics. *Algorithms Mol. Biol.*, **11**, 1–10.

Qin,J. *et al.* (2010) A human gut microbial gene catalogue established by metagenomic sequencing. *Nature*, **464**, 59–65.

Richter,D.C. *et al.* (2008) MetaSim: a sequencing simulator for genomics and metagenomics. *PloS One*, **3**, e3373.

Segata,N. *et al.* (2012) Metagenomic microbial community profiling using unique clade-specific marker genes. *Nat. Methods*, **9**,

Sims,G.E. *et al.* (2009) Alignment-free genome comparison with feature frequency profiles (ffp) and optimal resolutions. *Proc. Natl. Acad. Sci.*, **106**,

Sobih,A. *et al.* (2016) Metaflow: Metagenomic profiling based on whole-genome coverage analysis with min-cost flows. *bioRxiv*.

Song,K. *et al.* (2014) New developments of alignment-free sequence comparison: measures, statistics and next-generation sequencing. *Brief. Bioinf.*, **15**, 343–353.

Ulitsky,I. *et al.* (2006) The average common substring approach to phylogenomic reconstruction. *J. Comput. Biol.*, **13**, 336–350.

Vinh,L.V. *et al.* (2015) A two-phase binning algorithm using l-mer frequency on groups of non-overlapping reads. *Algorithms Mol. Biol.*, **10**, 1–12.

Wang,Y. *et al.* (2012) Metacluster 5.0: a two-round binning approach for metagenomic data for low-abundance species in a noisy sample. *Bioinformatics*, **28**,

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

Wu,Y.W. and Ye,Y. (2011) A novel abundance-based algorithm for binning metagenomic sequences using l-tuples. *J. Comput. Biol.*, **18**, 523–534.

Yang,B. *et al.* (2010). *ACM BCB'10*, chapter Metacluster: unsupervised binning of environmental genomic fragments and taxonomic annotation. ACM, New York, USA.

Zeller,G. *et al.* (2014) Potential of fecal microbiota for early-stage detection of colorectal cancer. *Mol. Syst. Biol.*, **10**, 766.