Bioinformatics, 2016, 1–8
doi: 10.1093/bioinformatics/btw290
Advance Access Publication Date: 2 June 2016
Original Paper

OXFORD

Genome analysis

COCACOLA: binning metagenomic contigs using sequence COmposition, read CoverAge, CO-alignment and paired-end read LinkAge

Yang Young Lu[1], Ting Chen[1,2], Jed A. Fuhrman[3] and Fengzhu Sun[1,4,*]

[1]Molecular and Computational Biology Program, Department of Biological Sciences, University of Southern California, Los Angeles, CA 90089, USA, [2]Center for Synthetic and Systems Biology, TNLIST, Beijing 100084, China, [3]Department of Biological Sciences and Wrigley Institute for Environmental Studies, University of Southern California, Los Angeles, CA 90089, USA and [4]Center for Computational Systems Biology, Fudan University, Shanghai 200433, China

*To whom correspondence should be addressed.
Associate Editor: Cenk Sahinalp

Received on April 11, 2016; revised on April 25, 2016; accepted on April 29, 2016

Abstract

Motivation: The advent of next-generation sequencing technologies enables researchers to sequence complex microbial communities directly from the environment. Because assembly typically produces only genome fragments, also known as contigs, instead of an entire genome, it is crucial to group them into operational taxonomic units (OTUs) for further taxonomic profiling and down-streaming functional analysis. OTU clustering is also referred to as binning. We present COCACOLA, a general framework automatically bin contigs into OTUs based on sequence composition and coverage across multiple samples.

Results: The effectiveness of COCACOLA is demonstrated in both simulated and real datasets in comparison with state-of-art binning approaches such as CONCOCT, GroopM, MaxBin and MetaBAT. The superior performance of COCACOLA relies on two aspects. One is using L_1 distance instead of Euclidean distance for better taxonomic identification during initialization. More importantly, COCACOLA takes advantage of both hard clustering and soft clustering by sparsity regularization. In addition, the COCACOLA framework seamlessly embraces customized knowledge to facilitate binning accuracy. In our study, we have investigated two types of additional knowledge, the co-alignment to reference genomes and linkage of contigs provided by paired-end reads, as well as the ensemble of both. We find that both co-alignment and linkage information further improve binning in the majority of cases. COCACOLA is scalable and faster than CONCOCT, GroopM, MaxBin and MetaBAT.

Availability and implementation: The software is available at https://github.com/younglululu/COCACOLA.

Contact: fsun@usc.edu

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

1 Introduction

Metagenomic studies aim to understand microbial communities directly from environmental samples without cultivating member species (Riesenfeld *et al.*, 2004). The next-generation sequencing technologies allow biologists to extract genomic data with unprecedented high resolution and sufficient sequence depth, offering insights into complex microbial communities even including species with low abundance (Albertsen *et al.*, 2013). To further investigate the taxonomic structure of microbial samples, assembled sequence fragments, also known as contigs, need be grouped into operational taxonomic units (OTUs) that ultimately represent genomes or significant parts of genomes. OTU clustering is also called binning

(or genomic binning), serving as the key step toward taxonomic profiling and downstream functional analysis. Therefore, accurate binning of the contigs is an essential problem in metagenomic studies.

Despite extensive studies, accurate binning of contigs remains challenging for several major reasons, including chimeric assemblies owing to repetitive sequence regions within or across genomes, sequencing errors or artifacts, strain-level variation within the same species, etc. (Alneberg *et al.*, 2014; Mande *et al.*, 2012) The currently available binning methods can be broadly categorized into classification and clustering approaches. Classification approaches are 'taxonomy dependent', that is, reference databases are needed for the assignment from contigs or reads to meaningful taxons. The classification is either based on homology owing to sequence identity, or genomic signatures such as oligonucleotide composition patterns and taxonomic clades. Homology-based methods include MEGAN (Huson *et al.*, 2007), which assigns reads to the lowest common taxonomic ancestor. Examples of genomic signature-based methods include PhyloPythia (McHardy *et al.*, 2007) and Kraken (Wood and Salzberg, 2014), which are composition-based classifiers and naive Bayesian classifier (Rosen *et al.*, 2011), a clade-specific approach. In addition, hybrid methods are available to take both alignment and composition-based strategy into consideration, such as PhymmBL (Brady and Salzberg, 2009) and SPHINX (Mohammed *et al.*, 2011).

In comparison, clustering approaches are 'taxonomy independent', that is, no additional reference databases or taxonomic information is needed. These approaches require similarity measurements from GC content, tetra-mer composition (Albertsen *et al.*, 2013; Chatterji *et al.*, 2008; Yang *et al.*, 2010) or Interpolated Markov Models (Kelley and Salzberg, 2010), to contig coverage profile (Baran and Halperin, 2012; Wu and Ye, 2011).

Recently, several methods have been developed to bin contigs using the coverage profiles of the contigs across multiple metagenomic samples (Albertsen *et al.*, 2013; Alneberg *et al.*, 2014; Carr *et al.*, 2013; Imelfort *et al.*, 2014; Kang *et al.*, 2015; Nielsen *et al.*, 2014; Wu *et al.*, 2015). Here the coverage of a contig is defined as the fraction of reads mapped to the contig in a sample. The idea is that if two contigs are from the same genome, their coverage profiles across multiple samples should be highly correlated. These methods can be further improved by integrating coverage profiles with the sequence tetra-mer composition of the contigs (Alneberg *et al.*, 2014; Imelfort *et al.*, 2014; Kang *et al.*, 2015). Among these methods, GroopM (Imelfort *et al.*, 2014) is advantageous in its visualized and interactive pipeline. On one hand, it is flexible, allowing users to merge and split bins under expert intervention. On the other hand, in the absence of expert intervention, the automatic binning results of GroopM is not as satisfactory as CONCOCT (Alneberg *et al.*, 2014). CONCOCT (Alneberg *et al.*, 2014) makes use of the Gaussian mixture model (GMM) to cluster contigs into bins. Also, CONCOCT provides a mechanism to automatically determine the optimal OTU number by variational Bayesian model selection (Corduneanu and Bishop, 2001). MetaBAT (Kang *et al.*, 2015) calculates integrated distance for pairwise contigs and then clusters contigs iteratively by modified K-medoids algorithm. And MaxBin (Wu *et al.*, 2015) compares the distributions of distances between and within the same genomes.

In this article we present COCACOLA, a general framework for contig binning incorporating sequence COmposition, CoverAge, CO-alignment and paired-end reads LinkAge across multiple samples. By default, COCACOLA uses sequence composition and coverage across multiple samples for binning. Compared with recent approaches such as CONCOCT, GroopM, MaxBin and MetaBAT, COCACOLA performs better in three aspects. First, COCACOLA reveals superiority with respect to precision, recall and Adjusted Rand Index (ARI). Second, COCACOLA shows better robustness in the case of varying number of samples. COCACOLA is scalable and faster than CONCOCT, GroopM, MaxBin and MetaBAT.

In addition, the COCACOLA framework seamlessly embraces customized knowledge to facilitate binning accuracy. In our study, we have investigated two types of knowledge, in particular, the co-alignment to reference genomes and linkage between contigs provided by paired-end reads. We find that both co-alignment and linkage information facilitate better binning performance in the majority of the cases.

2 Materials and methods

2.1 Problem formulation

A microbial community is composed of a set of OTUs at different abundance levels, and our objective is to put contigs into the genomic OTU bins from which they were originally derived. OTUs are expected to be disentangled based on contigs comprising either the discriminative abundance or dissimilarity among sequences in terms of l-mer composition. The rationale of binning contigs into OTUs relies on the underlying assumption that contigs originating from the same OTU share similar relative abundance as well as sequence composition.

Formally, we encode the abundance and composition of the k-th OTU by a $(M+V)$ dimensional feature vector, $W_{\cdot k}$, $k = 1, 2, \cdots, K$, where M is the number of samples, V is the number of distinct l-mers and K is the total OTU number. Specifically, W_{mk} represents the abundance of the k-th OTU in the m-th sample, $m = 1, 2, \cdots, M$, respectively. And $W_{M+v,k}$ stands for the l-mer relative frequency composition of the k-th OTU, $v = 1, 2, \cdots, V$. Similarly, the feature vector of the n-th contig is denoted as $X_{\cdot n}$. Let $\mathbb{H}_{kn}$ be the indicator function describing whether the n-th contig belongs to the k-th OTU, i.e. $\mathbb{H}_{kn} = 1$ means the n-th contig originating from the k-th OTU and $\mathbb{H}_{kn} = 0$ otherwise. Therefore, $X_{\cdot n}$ can be represented as:

$$X_{\cdot n} = \mathbb{H}_{1n} W_{\cdot 1} + \mathbb{H}_{2n} W_{\cdot 2} + \cdots + \mathbb{H}_{kn} W_{\cdot K}, \quad n = 1, 2, \cdots, N \quad (1)$$

where N is the number of contigs. Equation (1) can be further written into the matrix form:

$$X \approx W\mathbb{H} \qquad s.t. \quad W \geq 0, \quad \mathbb{H} \in \{0,1\}^{K \times N}, \|\mathbb{H}_{\cdot n}\|_0 = 1 \quad (2)$$

where $W = (W_{\cdot 1}, W_{\cdot 2}, \cdots, W_{\cdot K})$ is a $(M+V) \times K$ non-negative matrix with each column encoding the feature vector of the corresponding OTU. And $\mathbb{H} = (\mathbb{H}_{\cdot 1}, \mathbb{H}_{\cdot 2}, \cdots, \mathbb{H}_{\cdot N})$ is a $K \times N$ binary matrix with each column encoding the indicator function of the corresponding contig. $\|\mathbb{H}_{\cdot n}\|_0 = \sum_{k=1}^{K} \mathbb{H}_{kn} = 1$ ensures the n-th contig belongs exclusively to only one particular OTU.

The matrices W and $\mathbb{H}$ are obtained by minimizing a certain objective function. In this article we use Frobenius norm, commonly known as the sum of squared error:

$$\arg \min_{W, \mathbb{H} \geq 0} \|X - W\mathbb{H}\|_F^2 \qquad s.t. \quad \mathbb{H} \in \{0,1\}^{K \times N}, \|\mathbb{H}_{\cdot n}\|_0 = 1 \quad (3)$$

Note that Equation (3) is NP-hard by formulation as an integer programming problem with an exponential number of feasible solutions (Jiang *et al.*, 2014). A common procedure to tackle Equation (3) relaxes binary constraint of $\mathbb{H}$ with numerical values. Hence

Equation (3) is reformulated as the following minimization problem:

$$\arg\min_{W,H} \|X - WH\|_F^2 \qquad s.t. \quad W, H \geq 0 \tag{4}$$

where H serves as a coefficient matrix instead of an indicator matrix. In the scenario of Equation (4), $W_{\cdot k}$, the feature vector of the k-th OTU represents the centroid of the k-th cluster. Meanwhile, each contig $X_{\cdot n}$ is approximated by a weighted mixture of clusters, where the weights are encoded in $H_{\cdot n}$. In other words, relaxation of binary constraint makes the interpretation from hard clustering to soft clustering, where hard clustering means that a contig can be assigned to one OTU only, while soft clustering allows a contig to be assigned to multiple OTUs. It has been observed that by imposing sparsity on each column of H, the hard clustering behavior can be facilitated (Kim and Park, 2008). Therefore, Equation (4) is further modified through the Sparse Non-negative Matrix Factorization form (Kim and Park, 2008):

$$\arg\min_{W,H\geq 0} \|X - WH\|_F^2 + \alpha \sum_{n=1}^{N} \|H_{\cdot n}\|_1^2 \tag{5}$$

where $\|\cdot\|_1$ indicates L_1-norm. Owing to non-negativity of H, $\|H_{\cdot n}\|_1$ stands for the column sum of the n-th column vector of H. The parameter $\alpha > 0$ controls the trade-off between approximation accuracy and the sparseness of H. Namely, larger α implies stronger sparsity while smaller value ensures better approximation accuracy.

2.2 Feature matrix representation of contigs

Similar to CONCOCT (Alneberg *et al.*, 2014), each contig longer than 1000 bp is represented by a $(M + V)$ dimensional column feature vector including M dimensional coverage and V dimensional tetra-mer composition. The coverage denotes the average number of mapped reads per base pair from each of M different samples. While the tetra-mer composition denotes the tetra-mer frequency for the contig itself plus its reverse complement. Owing to palindromic tetra-mers, $V = 136$.

Adopting the notation of CONCOCT (Alneberg *et al.*, 2014), the coverage of all the N contigs is represented by an $N \times M$ matrix Y, where N is the number of contigs of interest and Y_{nm} indicates the coverage of the n-th contig from the m-th sample. Whereas the tetra-mer composition of the N contigs are represented by an $N \times V$ matrix Z where Z_{nv} indicates the count of v-th tetra-mer found in the n-th contig. Before normalization, a pseudo-count is added to each entry of the coverage matrix Y and composition matrix Z, respectively. As for the coverage, a small value is added, i.e. $Y'_{nm} = Y_{nm} + 100/L_n$, analogous to a single read aligned to each contig as prior, where L_n is the length of the n-th contig. As for the composition, a single count is simply added, i.e. $Z'_{nv} = Z_{nv} + 1$.

The coverage matrix Y is first column-wise normalized (i.e. normalization within each individual sample), followed by row-wise normalization (i.e. normalization across M samples) to obtain coverage profile p. The row-wise normalization aims to mitigate sequencing efficiency heterogeneity among contigs.

$$Y''_{nm} = \frac{Y'_{nm}}{\sum_{n=1}^{N} Y'_{nm}} \qquad p_{nm} = \frac{Y''_{nm}}{\sum_{m=1}^{M} Y''_{nm}} \tag{6}$$

The composition matrix Z is row-wise normalized for each contig (i.e. normalization across M tetra-mer count) to obtain composition profile q:

$$q_{nv} = \frac{Z'_{nv}}{\sum_{v=1}^{V} Z'_{nv}} \tag{7}$$

The feature matrix of contigs is denoted as $X = [p\ q]^T$, as the combination of coverage profile p and composition profile q. To be specific, X is a $(M + V) \times N$ non-negative matrix of which each column represents the feature vector of a particular contig.

2.3 Incorporating additional knowledge into binning

We consider two types of additional knowledge that may enhance the binning accuracy (Basu *et al.*, 2008). One option is paired-end reads linkage. Specifically, a high number of links connecting two contigs imply high possibility that they belong to the same OTU. Because the linkage may be erroneous owing to the existence of chimeric sequences, we keep linkages that are reported through multiple samples. The other option is co-alignment to reference genomes. That is, two contigs mapped to the same reference genome support the evidence that they belong to the same OTU.

We encode additional knowledge by an undirected network in the form of a non-negative weight matrix A, where $A_{nn'}$ quantifies the confidence level we believe the n-th contig and the n'-th contig to be clustered together. Based on the aforementioned matrix $\mathbf{A}$, a network regularization item is introduced to measure the coherence of binning (Cai *et al.*, 2011):

$$R_g = \frac{1}{2} \sum_{n,n'=1}^{N} \left\| H_{\cdot n} - H_{\cdot n'} \right\|^2 A_{nn'} = Tr(HLH^T) \tag{8}$$

where $Tr(\cdot)$ indicates the matrix trace, the sum of items along the diagonal. D denotes the diagonal matrix whose entries are column sums (or row sums owing to symmetry) of A, i.e. $D_{nn} = \sum_{n'=1}^{N} A_{nn'}$. The *Laplacian matrix* (Chung, 1997) is defined as $L = D - A$. With convention we use *normalized Laplacian matrix* instead, that is, $\mathcal{L} = D^{-1/2} L D^{-1/2} = I - D^{-1/2} A D^{-1/2} \triangleq I - \mathcal{A}$. By incorporating the network regularization in Equation (8), the objective function in Equation (5) changes to the following form:

$$\arg\min_{W,H\geq 0} \|X - WH\|_F^2 + \alpha \sum_{n=1}^{N} \left\| H_{\cdot n} \right\|_1^2 + \beta\, Tr(H\mathcal{L}H^T) \tag{9}$$

where the parameter $\beta > 0$ controls the trade-off of belief between unsupervised binning and additional knowledge. Namely, large β indicates strong confidence on the additional knowledge. Conversely, small β puts more weight on the data.

To use multiple additional knowledge sources together, a combined *Laplacian matrix* is constructed as a weighted average of individual *Laplacian matrices* $\bar{\mathcal{L}} = \left(\sum_d \alpha_d \mathcal{L}_d\right) / \left(\sum_d \alpha_d\right)$ where each positive weight α_d reflects the contribution of the corresponding information. For simplicity, weights are treated equally in the article.

2.4 Optimization by alternating non-negative least squares

Among comprehensive algorithms to solve Equation (9), the multiplicative updating approach (Lee and Seung, 1999) is most widely used. Despite its simplicity in implementation, slow convergence is of high concern. This article adopts a more efficient algorithm with provable convergence called alternating non-negative least squares (ANLS) (Kim and Park, 2008). ANLS iteratively handles two non-negative least square subproblems in Equation (10) until convergence. The ANLS algorithm is summarized in Algorithm 1.

$$H \leftarrow \arg\min_{H\geq 0} \|X - WH\|_F^2 + \alpha \sum_{n=1}^{N} \left\| H_{\cdot n} \right\|_1^2 + \beta\, Tr(H\mathcal{L}H^T) \tag{10a}$$

$$W \leftarrow \arg\min_{W\geq 0} \left\| X^T - H^T W^T \right\|_F^2 \tag{10b}$$

We solve Equation (10a) by block coordinate descent, that is, we divide Equation (10a) into N subproblems and minimize the objective function with respect to each subproblem at a time while keeping the rest fixed:

$$\begin{aligned}
&\arg\min_{H_{\cdot n}\geq 0} \|X_{\cdot n} - WH_{\cdot n}\|_2^2 + \alpha\|H_{\cdot n}\|_1^2 + \beta H_{\cdot n}^T \mathcal{L} H_{\cdot n}, \;\; n = 1, \cdots, N \\
&= \arg\min_{H_{\cdot n}\geq 0} \|X_{\cdot n} - WH_{\cdot n}\|_2^2 + \alpha\|H_{\cdot n}\|_1^2 + \beta H_{\cdot n}^T (H_{\cdot n} - 2\sum_{n'=1}^{N} \mathcal{A}_{nn'} H_{\cdot n'}^{old}) \\
&= \arg\min_{H_{\cdot n}\geq 0} \|X_{\cdot n} - WH_{\cdot n}\|_2^2 + \alpha\|H_{\cdot n}\|_1^2 + \beta \|H_{\cdot n} - \sum_{n'=1}^{N} \mathcal{A}_{nn'} H_{\cdot n'}^{old}\|_2^2
\end{aligned} \tag{11}$$

where the matrix H^{old} denotes the value of H obtained from the previous iteration. Following Jacobi updating rule, we combine N subproblems in Equation (11) into the matrix form:

$$\arg\min_{H\geq 0} \left\| \begin{pmatrix} X \\ 0_{1\times N} \\ \sqrt{\beta} H^{old} \mathcal{A} \end{pmatrix} - \begin{pmatrix} W \\ \sqrt{\alpha} e_{1\times K} \\ \sqrt{\beta} I_K \end{pmatrix} H \right\|_F^2 \tag{12}$$

where $0_{1\times N}$ is a N dimensional row vector of all 0, $e_{1\times K}$ is a K dimensional row vector of all 1.

2.5 Initialization of W and H

Note that we need to initialize W and H as the input to Algorithm 1. A good initialization not only enhances the accuracy of the solution, but facilitates fast convergence to a better local minima as well (Langville *et al.*, 2006). We initialize W and H by K-means clustering, namely, W is set to be the K-means centroid of X with each column $W_{\cdot k}$ corresponding to the feature vector of the k-th centroid. Meanwhile, H is set to be the indicator matrix encoding the cluster assignment.

The distance measurement contributes crucially to the success of binning. Ideally, a proper distance measurement exhibits more distinguishable taxonomic difference. The traditional K-means approach takes Euclidean distance as default measurement to quantify closeness. However, as for the coverage profile, Su *et al.* (2012) shows L_1 distance produces more reasonable binning results than Euclidean and correlation-based distances. As for the composition profile, L_1 distance also reveals superiority over Euclidean and cosine distances (Liao *et al.*, 2014). Therefore, our method adopts K-means clustering with L_1 distance. Once preliminary K-means clustering is achieved, we eliminate suspicious clusters with few contigs using the bottom-up L Method (Salvador and Chan, 2004). Performance comparisons with respect to L_1 and Euclidean distance are given in the supplementary material.

2.6 Parameter tuning

We have two parameters (α, β) to be tuned in our algorithm. Traditional cross-validation-like strategy demands searching through a two dimensional grid of candidate values, which is computationally unaffordable in the case of large datasets. Instead, we first search a good marginal α value by fixing $\beta = 0$. After that, a one-dimensional search is performed on a range of candidate β values while keeping α fixed.

In our implementation, when $\beta = 0$, α is approximated by the regression of the corresponding Lagrange Multipliers from N constrained problems $\text{argmin}_{H_{\cdot n}\geq 0}\|X - WH_{\cdot n}\|_F^2$ with constraint $(\|H_{\cdot n}\|_1 - 1)^2 = 0$, where $n = 1, \cdots, N$. The resulting α is denoted by α^*. Then we run the algorithm with respect to each candidate β and fixed $\alpha = \alpha^*$, resulting in corresponding binning results with various cluster number. Notice that traditional internal cluster validity indices are only applicable on the basis of fixed cluster number scenario (Wiwie *et al.*, 2015), such as Sum of Square Error and Davies-Bouldin index (Davies and Bouldin, 1979). To be specific, the indices have the tendency toward monotonically increase or decrease as the cluster number increases (Liu *et al.*, 2013). We tackle the impact of monotonicity by adopting TSS (Tang-Sun-Sun) minimization index (Tang *et al.*, 2005), that is, we choose the candidate β with minimum TSS value, recorded as β^*. Then we can solve Equation (9) by using (α^*, β^*) as selected regularization parameters.

Algorithm 1. Optimization by ANLS

Input: feature matrix $X \in \mathbb{R}^{(M+V)\times N}$, initial basis matrix $W \in \mathbb{R}^{(M+V)\times K}$ and coefficient matrix $H \in \mathbb{R}^{K\times N}$, tolerance threshold ε, maximum iteration threshold T

1: **repeat**
2: Obtain optimal H of Equation (10a) by fixing W
3: Obtain optimal W of Equation (10b) by fixing H
4: **until** A particular stopping criterion involving ε is satisfied or iteration number exceeds T

Output: W,H

2.7 Post-processing

The resulting binning obtained from Algorithm 1 may contain clusters that are closely mixed to each other. Therefore, we define *separable conductance* as an effective measurement to diagnose the coupling closeness of pairwise clusters, so as to determine whether to merge them. Namely, we consider each cluster as having a spherical scope centered at its centroid. To be robust against outliers, the radius is chosen as the third quartile among the intra-cluster distances. The *separable conductance* between the c_1-th cluster and the c_2-th cluster, $sep(c_1, c_2)$, is defined as the number of contigs from the c_1-th cluster also included in the spherical scope of the c_2-th cluster, divided by the smaller cluster size of two. Intuitively, the *separable conductance* exploits the overlap between two clusters. The procedure of post-processing works as follows: we keep picking the pair of clusters with maximum *separable conductance* and merge them until it fails to exceed a certain threshold. The threshold is set to be 1 in this study.

2.8 Datasets

Alneberg *et al.* (2014) simulated a 'species' dataset and another 'strain' dataset. Both simulated datasets were constructed based on 16S rRNA samples originated from the Human Microbiome Project (HMP) (Consortium *et al.*, 2012). The relative abundance profiles of the different species/strains for the simulation were based on the HMP samples as well.

The simulated 'species' dataset consisted of 101 different species across 96 samples. It aimed to test the ability of CONCOCT to cluster contigs in complex populations (Alneberg *et al.*, 2014). The species were approximated by the OTUs from HMP with >3% sequence differences. Each species was guaranteed to appear in at least 20 samples. A total of 37 628 contigs remain for binning after co-assembly and filtering.

The simulated 'strain' dataset aimed to test the ability of CONCOCT to cluster contigs at different levels of taxonomic

resolution (Alneberg *et al.*, 2014). To be more specific, the simulated 'strain' dataset consisted of 20 different species or strains from the same species across 64 samples, including five different *Escherichia coli* strains, five different *Bacteroides* species, five different species from different *Clostridium* genera and five different *gut* bacteria. It was challenging for CONCOCT to separate the five different *E.coli* strains (Alneberg *et al.*, 2014). A total of 9417 contigs remain for binning after co-assembly and filtering.

In addition to two simulated datasets, we use a time-series study of 11 fecal microbiome samples from a premature infant (Sharon *et al.*, 2013), denoted as the 'Sharon' dataset. Because the true species that contigs belong to are not known, we assign the class labels by annotating contigs using the TAXAassign script (Ijaz and Quince, 2013). As a result, 2614 of 5579 contigs are unambiguously labeled on the species level for evaluation. Another real dataset embody 264 samples from the MetaHIT consortium (Qin *et al.*, 2010) (SRA:ERP000108), the same dataset used in MetaBAT (Kang *et al.*, 2015), denoted as the 'MetaHIT' dataset. In all, 17 136 of 192 673 co-assembled contigs are unambiguously labeled on the species level for evaluation.

2.9 Evaluation criteria

We use the standard measures including precision, recall and ARI to evaluate the clustering results. Their definitions are given in the supplementary material.

3 Results

Given the same input, i.e. sequence composition and coverage across multiple samples, we show the effectiveness of COCACOLA on simulated 'species' and 'strain' datasets, in comparison with three state-of-art, methodologically distinct methods for contigs binning: CONCOCT (Alneberg *et al.*, 2014), GroopM (Imelfort *et al.*, 2014), MaxBin (Wu *et al.*, 2015) and MetaBAT (Kang *et al.*, 2015). The comparison excludes Canopy (Nielsen *et al.*, 2014) that is based on binning co-abundant gene groups instead of binning contigs. Furthermore, we investigate the performance improvement of COCACOLA after incorporating two additional knowledge, co-alignment to reference genomes and linkage between contigs provided by paired-end reads, as well as the ensemble of both. Results reveal both information facilitating better performance in the majority of cases. Finally, we report the performance of COCACOLA on two real datasets.

3.1 Performance on the simulated datasets

Even though both COCACOLA and CONCOCT are able to determine the OTU number automatically, an initial estimation of OTU number K is needed to start from. Because the OTU number is usually unknown, we study the binning performance with respect to the value of K chosen empirically. Comprehensive studies on binning performance with respect to varying K are given in the supplementary material.

We observed that K-means clustering tends to generate empty clusters given large K. Our strategy is to increase K until there are more than $K/2$ empty clusters, and we choose the corresponding K as the input. At this stage, we emphasize more on the redundancy of OTU number rather than the accuracy. Thus, we obtain $K = 192$ and $K = 48$ as input to the simulated 'species' and 'strain' dataset, respectively.

For the simulated 'species' dataset, Figure 1(a) compares COCACOLA against CONCOCT, GroopM, MaxBin and MetaBAT in terms of precision, recall and ARI. The precision of COCACOLA is 0.9978, suggesting that almost all contigs within each cluster originate from the same species. In comparison, the precision of CONCOCT, GroopM, MaxBin and MetaBAT is 0.9343, 0.9324, 0.9973 and 0.9958, respectively. The recall obtained by COCACOLA is 0.9993, implying that nearly all contigs derived from the same species are grouped into the same clusters. In contrast, the

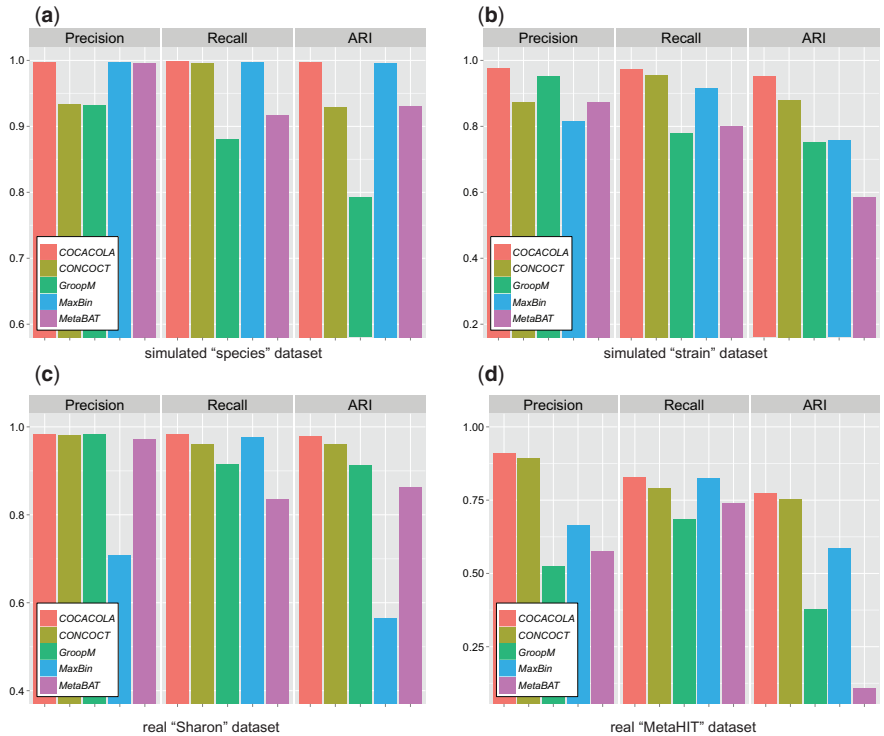

Fig. 1. The performance of COCACOLA, CONCOCT, GroopM, MaxBin and MetaBAT on both simulated datasets (**a** and **b**) and real datasets (**c** and **d**)

recall of CONCOCT, GroopM, MaxBin and MetaBAT is 0.996, 0.881, 0.9973 and 0.9174, respectively. As for ARI, COCACOLA achieves 0.997 while CONCOCT, GroopM, MaxBin and MetaBAT get 0.9296, 0.7922, 0.9961 and 0.9308, respectively.

For the simulated 'strain' dataset, the results are shown by Figure 1(b). The precision, recall and ARI of COCACOLA reach 0.9766, 0.9747 and 0.9512, respectively. In comparison, CONCOCT, GroopM, MaxBin and MetaBAT achieve 0.8733, 0.9525, 0.8151 and 0.8730 in terms of precision, 0.9552, 0.7805, 0.9167 and 0.8009 in terms of recall, 0.8809, 0.7529, 0.757 and 0.5858 in terms of ARI, respectively.

We conclude that COCACOLA performs well in constructing species from highly complicated environmental samples. Besides, COCACOLA performs well in handling strain-level variations, which cannot be fully resolved owing to assembly limitation (Alneberg *et al.*, 2014).

3.2 The effect of incorporating additional knowledge on binning

We investigate the performance improvement of COCACOLA after incorporating two additional knowledge as proposed in the 'Methods' section, in particular, co-alignment to reference genomes and linkage between contigs provided by paired-end reads. Moreover, we study the ensemble of both. The comparison is between the binning result by COCACOLA incorporating additional knowledge against the result without. The comparison is based on sub-samples of the simulated 'species' dataset. We choose sub-samples of size ranging from 10 to 90, with 10 as increment. To avoid duplicate contribution from a particular sample, we choose sub-samples without overlapping. Therefore, the numbers of sub-samples are 9, 4, 3, 2, 1, 1, 1, 1, 1, respectively. Because the contributions from additional knowledge nearly diminish when the sample size exceeds $K = 30$, therefore we focus on the 16 cases from $K = 10$ to $K = 30$.

In terms of co-alignment, we design the symmetric weight matrix $\mathbf{A}_{nn'} = 1$ if contig n and contig n' are aligned to the same species using the TAXAassign script (Ijaz and Quince, 2013). As shown in Figure 2(a–c), the precision is improved noticeably in 7 cases and decreased in 3 cases, the recall is improved noticeably in 11 cases and decreased slightly in 1 case, the ARI is improved noticeably in 10 cases and decreased slightly in 2 cases.

In terms of linkage, we design the symmetric weighted matrix $\mathbf{A}_{nn'}$ as the number of samples supporting linkage connecting contig n and contig n'. As depicted in Figure 2(d–f), the precision is improved noticeably in seven cases and decreased in two cases, the recall is improved noticeably in seven cases and decreased slightly in

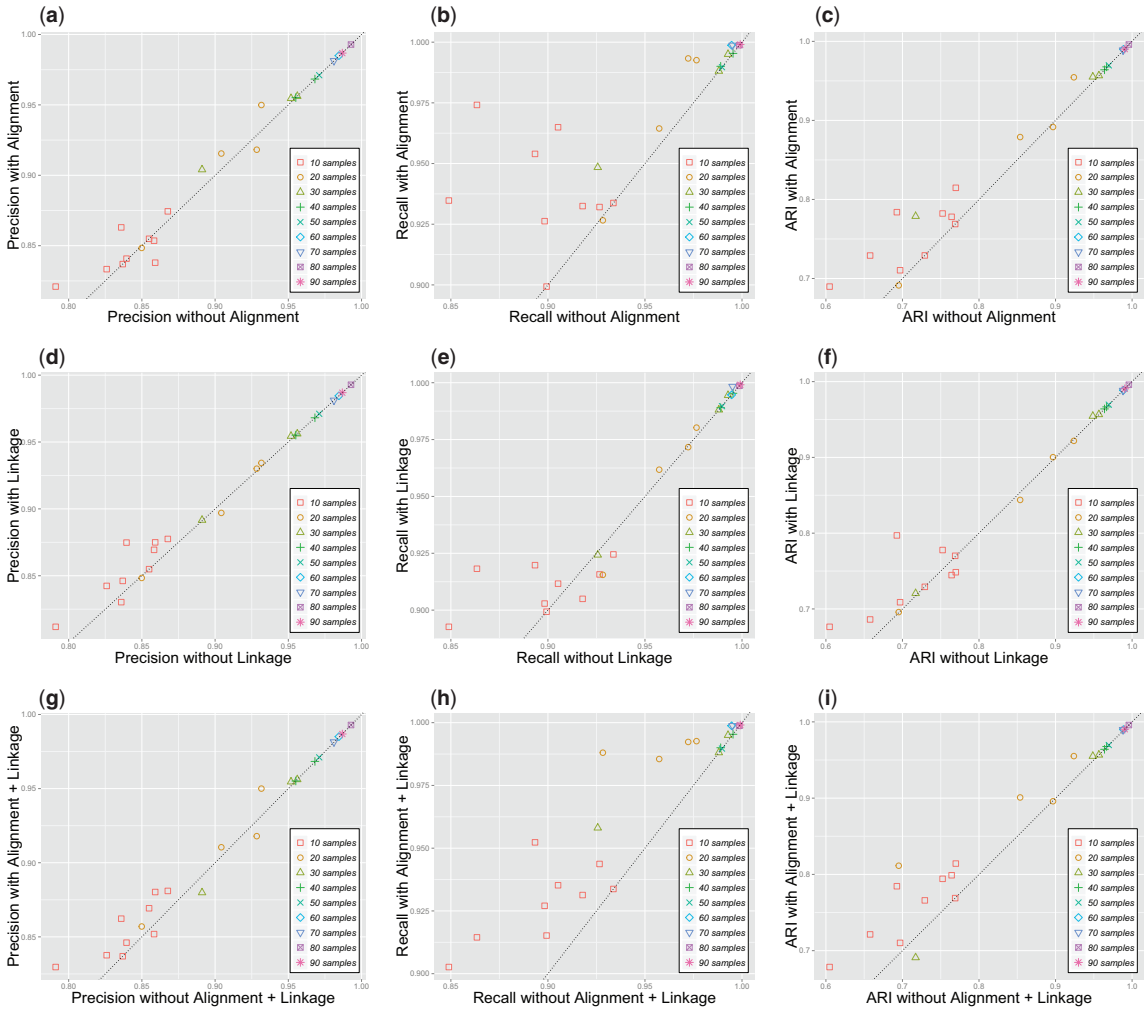

Fig. 2. Evaluation of the impact of incorporating two additional knowledge (Section 2.3) on sub-samples of simulated 'species' dataset. The first option is co-alignment information to reference genomes, depicted by (**a–c**). The second option is paired-end reads linkage, depicted by (**d–f**). The ensemble of both is depicted by (**g–i**)

four cases, the ARI is improved noticeably in five cases and decreased in three cases.

In terms of the ensemble of co-alignment and linkage, as depicted in Figure 2(g–i), the precision is improved noticeably in 10 cases and decreased in 3 cases, the recall is improved noticeably in 13 cases and no case suffers decreasing, the ARI is improved noticeably in 11 cases and decreased in 1 cases.

We have the following conclusions: (i) When there are sufficient number of samples, the contributions from additional knowledge diminish. (ii) Additional knowledge such as co-alignment and linkage information facilitate better overall performance in the majority of cases. (iii) Ensemble of both information performs more stable than individual information.

3.3 Performance on real datasets

Applying COCACOLA to the 'Sharon' dataset (Figure 1(c)), given initial choice of $K = 30$, the precision, recall and ARI reach 0.9889, 0.9759 and 0.9670, respectively. In comparison, CONCOCT, GroopM, MaxBin and MetaBAT achieve 0.9801, 0.9820, 0.7077 and 0.9705 in terms of precision, 0.9606, 0.9147, 0.9767 and 0.8344 in terms of recall, 0.9600, 0.9126, 0.5639 and 0.8634 in terms of ARI, respectively. COCACOLA identifies six OTUs corresponding to six reported genomes. In comparison, CONCOCT, GroopM, MaxBin and MetaBAT identify 14, 24, 5 and 11 OTUs, respectively.

Next, we investigate the performance improvement of COCACOLA after incorporating additional knowledge. We use linkage information only because it is circular to use TAXAassign script (Ijaz and Quince, 2013) on both alignment and labeling. COCACOLA still identifies six OTUs, with the precision, recall and ARI reaching 0.9923, 0.9797 and 0.9743, slightly outperforms the case without additional knowledge.

Applying COCACOLA to the 'MetaHIT' dataset (Figure 1(d)), given initial choice of $K = 100$, the precision, recall and ARI reach 0.9082, 0.8272 and 0.7717, respectively. In comparison, CONCOCT, GroopM, MaxBin and MetaBAT achieve 0.8933, 0.5247, 0.6655 and 0.5738 in terms of precision, 0.7901, 0.6843, 0.8228 and 0.7397 in terms of recall, 0.7518, 0.3757, 0.5866 and 0.1088 in terms of ARI, respectively.

Next we investigate the performance improvement of COCACOLA after incorporating linkage information. The performance is further slightly improved from 0.9082 to 0.9084 in terms of precision, from 0.8272 to 0.8350 in terms of recall and from 0.7717 to 0.7844 in terms of ARI, respectively.

3.4 Running time of COCACOLA, CONCOCT, GroopM, MaxBin and MetaBAT

COCACOLA shares the same data parsing pipeline as CONCOCT and differs only in the binning step, whereas GroopM uses its own workflow. It is reasonable to compare running time of binning directly between COCACOLA and CONCOCT. To bring GroopM into context, we take into account the stages related to binning and therefore exclude the data parse stage. As for MaxBin and MetaBAT we simply pre-calculate the abundance and depth information. MaxBin involves multi-threaded parameter, which is set as the number of cores. All of five methods run on the 12-cores and 60GB-RAM computing platform provided by the USC High Performance Computing Cluster. The comparison is conducted on both the simulated datasets and real datasets (Table 1). We conclude that COCACOLA runs faster than CONCOCT, GroopM, MaxBin and MetaBAT.

4 Discussion

In this article, we develop a general framework to bin metagenomic contigs using sequence composition and coverage across multiple samples. Our approach, COCACOLA, outperforms state-of-art binning approaches CONCOCT (Alneberg *et al.*, 2014), GroopM (Imelfort *et al.*, 2014), MaxBin (Wu *et al.*, 2015) and MetaBAT (Kang *et al.*, 2015) on both simulated and real datasets.

The superior performance of COCACOLA relies on several aspects. First, initialization plays an important role in binning accuracy. Second, COCACOLA uses L_1 distance instead of Euclidean distance for better taxonomic identification. Third, COCACOLA takes advantage of both hard clustering and soft clustering. Specifically, soft clustering (such as the GMM used by CONCOCT) allows a contig to be assigned probabilistically to multiple OTUs, hence gains more robust results in general in comparison with hard clustering (such as the Hough partitioning used by GroopM). However, in complex environmental samples with strain-level variations, the corresponding OTUs are closely intertwined. Whereas soft clustering in turn further mixes the OTUs up and thus deteriorates clustering performance. COCACOLA obtains better trade-off between hard clustering and soft clustering by exploiting sparsity.

However, we notice that binning metagenomic contigs remains challenging when the number of samples is small, regardless of using COCACOLA, CONCOCT, GroopM, MaxBin or MetaBAT. With small number of metagenomic samples, the relationship between the contigs cannot be accurately inferred based on the relationship between the abundance profiles. Therefore, future research needs to study how to re-weight the contributions of abundance profiles and composition profiles in unsupervised (Cai *et al.*, 2010) or semi-supervised (Zhao and Liu, 2007) scenario. Moreover, recent studies suggest that Euclidean or L_1 distance between l-mer frequencies do not perform as well as alternative dissimilarity measurements such as d_2^* and $d_2^{(shepp)}$ (Wan *et al.*, 2010) in comparing genome sequence. However, the use of such measurements is computationally challenging, which needs further exploration.

The COCACOLA framework seamlessly embraces customized knowledge to facilitate binning accuracy. In our study, we have investigated two types of knowledge, in particular, the co-alignment to reference genomes and linkage of contigs provided by paired-end

Table 1. Running Time of COCACOLA, CONCOCT, GroopM, MaxBin and MetaBAT

Dataset	COCACOLA		CONCOCT		GroopM		MaxBin		MetaBAT	
	Time	Speedup	Time	Speedup	Time	Speedup	Time	Speedup	Time	Speedup
'species'	1m41.50s	1×	17m14.71s	10.2×	1h57m28s	69.4×	49m48.52s	29.4×	4m16.14s	2.5×
'strain'	10.94s	1×	1m10.99s	6.5×	17m00.46s	93.3×	9m54.80s	54.4×	2m31.52s	13.9×
'Sharon'	13.22s	1×	25.11s	1.9×	4m45.85s	21.6×	1m36.09s	7.3×	24.66s	1.9×
'MetaHIT'	2m39.12s	1×	20m20.90s	7.7×	12m47.68s	4.8×	2h20m52s	53.1×	7m25.07s	2.8×

reads. Even though the contributions from additional knowledge diminish when there are sufficient number of samples, they play an important role in binning results when the number of samples is small. In future studies, we intend to explore better customized prior knowledge. one option is exploiting phylogenetic information in taxonomic annotation (Purdom, 2011). Another option relies on identifying functional annotation of contigs, including open reading frames that are likely to encode proteins (Ye and Tang, 2009), or co-abundance gene groups (Nielsen *et al.*, 2014), etc. We have also investigated the ensemble of both co-alignment and linkage knowledge, and it shows more stable performance than individual information. In future studies, we aim to find optimal weights (Tsuda *et al.*, 2005) instead of equal weights.

Acknowledgements

The authors thank anonymous referees for helpful comments on this work. The research is partially supported by NSF DMS-1518001 and OCE 1136818.

Conflict of Interest: none declared.

References

Albertsen, M. *et al.* (2013) Genome sequences of rare, uncultured bacteria obtained by differential coverage binning of multiple metagenomes. *Nat. Biotechnol.*, **31**, 533–538.

Alneberg, J. *et al.* (2014) Binning metagenomic contigs by coverage and composition. *Nat. Methods*, **11**, 1144–1146.

Baran, Y. and Halperin, E. (2012) Joint analysis of multiple metagenomic samples. *PLoS Comput. Biol.*, **8**, e1002373.

Basu, S. *et al.* (2008). *Constrained Clustering: Advances in Algorithms, Theory, and Applications*. Data Mining and Knowledge Discovery Series. Chapman & Hall/CRC Press, Boca Raton, Florida, USA.

Brady, A. and Salzberg, S.L. (2009) Phymm and PhymmBL: metagenomic phylogenetic classification with interpolated Markov models. *Nat. Methods*, **6**, 673–676.

Cai, D. *et al.* (2010). Unsupervised feature selection for multi-cluster data. In: *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington, DC, USA, pp. 333–342.

Cai, D. *et al.* (2011) Graph regularized nonnegative matrix factorization for data representation. *IEEE Trans. Pattern Anal. Mach. Intell.*, **33**, 1548–1560.

Carr, R. *et al.* (2013) Reconstructing the genomic content of microbiome taxa through shotgun metagenomic deconvolution. *PLoS Comput. Biol.*, **9**, e1003292.

Chatterji, S. *et al.* (2008) Compostbin: a DNA composition-based algorithm for binning environmental shotgun reads. *Res. Comput. Mol. Biol.*, 17–28.

Chung, F.R. (1997). *Spectral Graph Theory*, Vol. **92**. American Mathematical Society.

Consortium, H.M.P. *et al.* (2012) Structure, function and diversity of the healthy human microbiome. *Nature*, **486**, 207–214.

Corduneanu, A. and Bishop, C.M. (2001) Variational Bayesian model selection for mixture distributions. In: *Artificial intelligence and Statistics 2001*, Key West, Florida, USA, pp. 27–34.

Davies, D.L. and Bouldin, D.W. (1979) A cluster separation measure. *IEEE Trans. Pattern Anal. Mach. Intell.*, **1**, 224–227.

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

Ijaz, U. and Quince, C. (2013). TAXAassign v0.4. https://github.com/umerijaz/taxaassign.

Imelfort, M. *et al.* (2014) GroopM: an automated tool for the recovery of population genomes from related metagenomes. *PeerJ*, **2**, e603.

Jiang, P. *et al.* (2014) A clustering approach to constrained binary matrix factorization. In: *Data Mining and Knowledge Discovery for Big Data*, Springer-Verlag, Berlin Heidelberg, pp. 281–303.

Kang, D.D. *et al.* (2015) MetaBAT, an efficient tool for accurately reconstructing single genomes from complex microbial communities. *PeerJ*, **3**, e1165.

Kelley, D.R. and Salzberg, S.L. (2010) Clustering metagenomic sequences with interpolated Markov models. *BMC Bioinformatics*, **11**, 544.

Kim, J. and Park, H. (2008) Sparse nonnegative matrix factorization for clustering. Technical Report GT-CSE-08-01, Georgia Institute of Technology, Atlanta, Georgia, USA.

Langville, A.N. *et al.* (2006) Initializations for the nonnegative matrix factorization. In *Proceedings of the twelfth ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, Philadelphia, Pennsylvania, USA, pp. 23–26.

Lee, D.D. and Seung, H.S. (1999) Learning the parts of objects by non-negative matrix factorization. *Nature*, **401**, 788–791.

Liao, R. *et al.* (2014) A new unsupervised binning approach for metagenomic sequences based on n-grams and automatic feature weighting. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **11**, 42–54.

Liu, Y. *et al.* (2013) Understanding and enhancement of internal clustering validation measures. *IEEE Trans. Cybern.*, **43**, 982–994.

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

McHardy, A.C. *et al.* (2007) Accurate phylogenetic classification of variable-length DNA fragments. *Nat. Methods*, **4**, 63–72.

Mohammed, M.H. *et al.* (2011) SPHINXan algorithm for taxonomic binning of metagenomic sequences. *Bioinformatics*, **27**, 22–30.

Nielsen, H.B. *et al.* (2014) Identification and assembly of genomes and genetic elements in complex metagenomic samples without using reference genomes. *Nat. Biotechnol.*, **32**, 822–828.

Purdom, E. (2011) Analysis of a data matrix and a graph: metagenomic data and the phylogenetic tree. *Ann. Appl. Stat.*, 2326–2358.

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

Riesenfeld, C.S. *et al.* (2004) Metagenomics: genomic analysis of microbial communities. *Annu. Rev. Genet.*, **38**, 525–552.

Rosen, G.L. *et al.* (2011) NBC: the naive bayes classification tool webserver for taxonomic classification of metagenomic reads. *Bioinformatics*, **27**, 127–129.

Salvador, S. and Chan, P. (2004). Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In: *Proceedings of the 16th IEEEE International Conference on Tools with AI (ICTAI)*, Boca Raton, Florida, USA, pp. 576–584.

Sharon, I. *et al.* (2013) Time series community genomics analysis reveals rapid shifts in bacterial species, strains, and phage during infant gut colonization. *Genome Res.*, **23**, 111–120.

Su, C.H. *et al.* (2012) The impact of normalization and phylogenetic information on estimating the distance for metagenomes. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **9**, 619–628.

Tang, Y. *et al.* (2005). Improved validation index for fuzzy clustering. In: *American Control Conference*, pp. 1120–1125.

Tsuda, K. *et al.* (2005) Fast protein classification with multiple networks. *Bioinformatics*, **21**, ii59–ii65.

Wan, L. *et al.* (2010) Alignment-free sequence comparison (ii): theoretical power of comparison statistics. *J. Comput. Biol.*, **17**, 1467–1490.

Wiwie, C. *et al.* (2015) Comparing the performance of biomedical clustering methods. *Nat. Methods*, page (epub ahead of print).

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

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.

Wu, Y.W. *et al.* (2016) MaxBin 2.0: an automated binning algorithm to recover genomes from multiple metagenomic datasets. *Bioinformatics*, **32**, 605–607.

Yang, B. *et al.* (2010) Unsupervised binning of environmental genomic fragments based on an error robust selection of l-mers. *BMC Bioinformatics*, **11**, S5.

Ye, Y. and Tang, H. (2009) An ORFome assembly approach to metagenomics sequences analysis. *J. Bioinform. Comput. Biol.*, **7**, 455–471.

Zhao, Z. and Liu, H. (2007) Semi-supervised feature selection via spectral analysis. In: *Proceedings of SIAM International Conference on Data Mining*, pp. 641–646.