OXFORD

Briefings in Bioinformatics, 20(4), 2019, 1125–1139

doi: 10.1093/bib/bbx120
Advance Access Publication Date: 23 September 2017
Paper

A review of methods and databases for metagenomic classification and assembly

Florian P. Breitwieser, Jennifer Lu and Steven L. Salzberg

Corresponding author: Steven L. Salzberg, Center for Computational Biology, Johns Hopkins University, 1900 E. Monument St., Baltimore, MD, 21205, USA. E-mail: salzberg@jhu.edu

Abstract

Microbiome research has grown rapidly over the past decade, with a proliferation of new methods that seek to make sense of large, complex data sets. Here, we survey two of the primary types of methods for analyzing microbiome data: read classification and metagenomic assembly, and we review some of the challenges facing these methods. All of the methods rely on public genome databases, and we also discuss the content of these databases and how their quality has a direct impact on our ability to interpret a microbiome sample.

Key words: microbiome; microbial genomics; next-generation sequencing; bacteria; databases

Introduction

Microbiome research has been expanding rapidly as a consequence of dramatic improvements in the efficiency of genome sequencing. As the variety and complexity of experiments has grown, so have the methods and databases used to analyze these experiments. Ever-larger data sets present increasing challenges for computational methods, which must minimize processing and memory requirements to provide fast turnaround and to avoid overwhelming the computational resources available to most research laboratories. The rapid increase in the number and variety of genomes also present many challenges, rising in part from the effort required to fit traditional taxonomic naming schemes onto a microbial world that we now know is vastly richer and more complex than scientists realized when they first created taxonomic naming schemes in the distant past. Additional challenges arise from the rapid pace of 'draft' genome sequencing, which has produced tens of thousands of new genomes, many of which are highly fragmented and incomplete. As we discuss below, the variable quality of these genomes can lead to unexpected and erroneous results if the genomes are used without careful vetting.

This review discusses the computational challenges of analyzing metagenomics data, focusing on methods but also including a discussion of microbial taxonomy and genome resources, which are rarely discussed in benchmark studies and tool reviews despite their critical importance. We begin with a review of terminology and a comparison of marker gene sequencing, shotgun metagenome sequencing and meta-transcriptome sequencing, all of which are sometimes included in the term metagenomics.

Metataxonomics, metagenomics, metatranscriptomics

The most widely used sequencing-based approaches for microbiome research are metataxonomics and metagenomics (Table 1). Metataxonomics refers to the sequencing of marker genes, usually regions of the ribosomal RNA (rRNA) gene that is highly conserved across taxa. Note that there has been some ambiguity in the use of these terms; in the past, marker gene sequencing has also been referred to as metagenomics. In this review, we follow the proposal of Marchesi and Ravel [1] on terminology, and use the term 'metataxonomics' for marker gene

Florian P. Breitwieser is a postdoctoral fellow at the Center for Computational Biology at Johns Hopkins School of Medicine. His research interests include metagenomics classification and visualization methods and their application to infectious disease diagnosis.
Jennifer Lu is a Biomedical Engineering PhD student in Steven Salzberg's laboratory at the Center for Computational Biology at Johns Hopkins University. Her research focuses on computational genomics and the usage of sequencing for diagnosing microbial infections relating to human health and diseases.
Steven L. Salzberg is the Bloomberg Distinguished Professor of Biomedical Engineering, Computer Science and Biostatistics at Johns Hopkins University. His laboratory conducts research on DNA and RNA sequence analysis including genome assembly, transcriptome assembly, sequence alignment and metagenomics.
Submitted: 7 June 2017; **Received (in revised form):** 22 August 2017

Table 1. Metataxonomics, metagenomics and meta-transcriptomics strategies

Technique	Advantages and challenges	Main applications
Metataxonomics using amplicon sequencing of the 16S or 18S rRNA gene or ITS	+ Fast and cost-effective identification of a wide variety of bacteria and eukaryotes – Does not capture gene content other than the targeted genes – Amplification bias – Viruses cannot be captured	* Profiling of what is present * Microbial ecology * rRNA-based phylogeny
Metagenomics using random shotgun sequencing of DNA or RNA	+ No amplification bias + Detects bacteria, archaea, viruses and eukaryotes + Enables *de novo* assembly of genomes – Requires high read count – Many reads may be from host – Requires reference genomes for classification	* Profiling of what is present across all domains * Functional genome analyses * Phylogeny * Detection of pathogens
Meta-transcriptomics using sequencing of mRNA	+ Identifies active genes and pathways – mRNA is unstable – Multiple purification and amplification steps can lead to more noise	* Transcriptional profiling of what is active

sequencing. Because it only requires sequence from a single gene, this strategy provides a cost-effective means to identify a wide range of organisms. Metagenomics refers to the random 'shotgun' sequencing of microbial DNA, without selecting any particular gene [2]. Both metataxonomics and metagenomics can provide information on the species composition of a microbiome. Another strategy, metatranscriptomics, attempts to capture and sequence all of the RNA in a sample, which can help create a profile of all genes that are actively being transcribed, and may also provide a picture of the relative abundance of those genes [3].

Complementary approaches that are becoming increasingly popular in microbiome research, but are not further covered in this review, include metaproteomics and metametabolomics [4–6]. Metaproteomics uses mass spectrometry techniques, e.g. liquid chromatography-coupled tandem mass spectrometry, to generate profiles of protein expression and posttranslational modifications of proteins [5]. Typically, genome sequences are required for the mapping of generated mass spectra to proteins, and thus, this field also depends on metagenomics. Metametabolomics attempts to create profiles of metabolites, usually also created using mass spectrometry [6]. Mass spectrometry is more expensive and experimentally challenging than sequencing, although the field is making continual technical improvements [4]. Integrating the data of all these different 'meta-omics' approaches is challenging, but it can yield insights not found by looking at just one type of data [7].

Metataxonomics is an invaluable tool for microbial ecology. rRNA gene sequences are the most widely used marker sequences; these include the 16S rRNA gene for bacteria, the 18S rRNA gene for eukaryotes, and the internal transcribed spacer (ITS) regions of the fungal ribosome for fungi [8, 9]. These markers work well for phylogenetic profiling because they are ubiquitously present in the population, they have hypervariable regions that differentiate species and they are flanked by conserved regions that can be targeted by 'universal' primers [8]. A major advantage of rRNA analysis is that databases such as Greengenes [10], RDP [11] and SILVA [12] contain genes from millions of species, making them far more comprehensive than genome databases, which contain tens of thousands of species. The workflow for 16S analysis typically includes quality filtering, error correction (sometimes called de-noising), removal of chimeric sequences, clustering of reads into 'Operational Taxonomic Units' (OTUs) based on sequence similarity and classification of the OTUs [13–20]. An alternative approach before clustering of reads into OTUs is their direct classification using metagenomics classifiers (see section on 'metagenomics classification' and Table 3), as recently compared in [21]. The rest of this review will focus on metagenomics methods; for further discussion of metataxonomic methods, see [22–25].

Marker gene sequencing does have some drawbacks, which explains (in part) the rising popularity of metagenomics. First, marker gene-based methodologies do not capture viruses, which have no conserved genes analogous to 16S or 18S rRNA genes. The use of the 16S rRNA gene itself is imperfect as well: for the recently described Candidate Phyla Radiation, which comprises up to 15% of the bacterial domain [26], it was estimated that >50% of the organisms evaded detection with classical 16S amplicon sequencing [27]. The short reads produced by next-generation sequencers further limit analysis at the species level, although full-length 16S rRNA gene sequencing using long-read sequencers from Pacific Biosciences or Oxford Nanopore might help overcome this limitation [28]. The methodology of an experiment and laboratory-specific factors can also limit the effectiveness of marker gene sequencing approaches, although the same caveat applies to metagenomics [29–32].

Metagenomic analysis

Many strategies can be used for analysis of metagenomics shotgun data (Figure 1). A common first step is to run a variety of computational tools for quality control, which identify and remove low-quality sequences and contaminants. These include programs such as FastQC [33], Cutadapt [34], BBDuk [35] and Trimmomatic [36] (Table 2). FastQ Screen [37] matches reads against multiple reference genomes such as human, mouse, *Escherichia coli* and yeast, and can provide a quick overview of where the reads align. Diginorm [38], implemented in the khmer package [39], can be used to reduce redundancy of reads in high-depth areas by down-sampling reads, and thus normalize coverage and make subsequent analyses computationally cheaper. MultiQC [40] aggregates quality control reports from multiple samples into a single report that can be viewed more

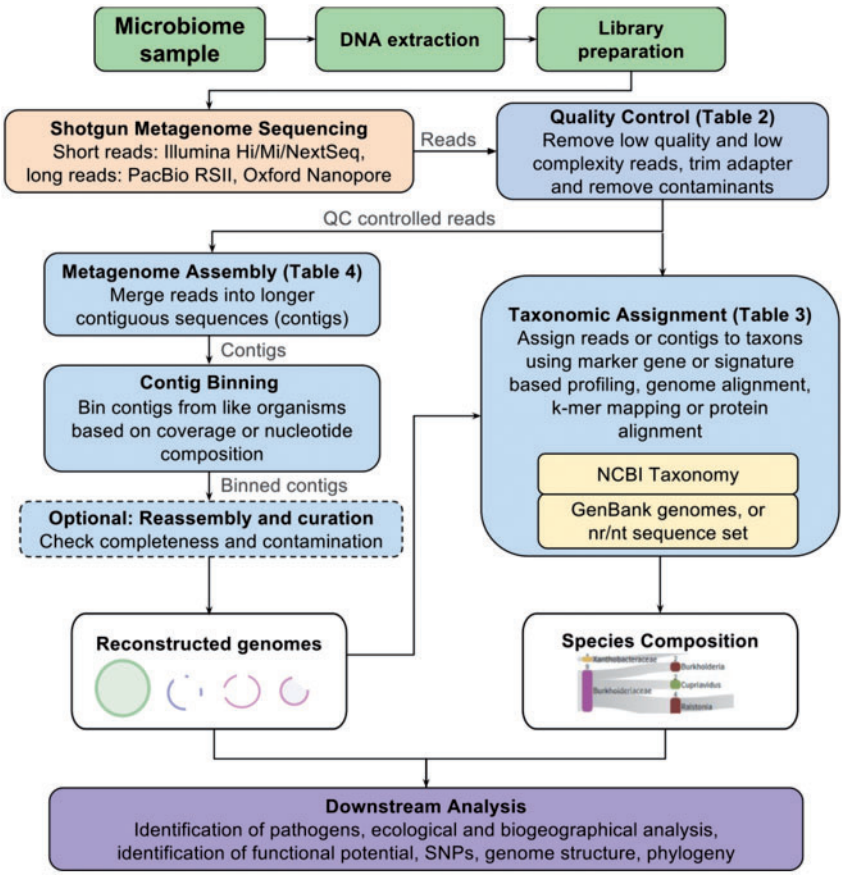

Figure 1. Common analysis procedures for metagenomics data. Note that the order of some of the analysis steps can be shuffled. For example, reads might be binned before assembly or before taxonomic assignment, so that the downstream algorithms can work only with a subset of the data.

Table 2. A selection of quality control software tools for metagenomics data

Tool	Synopsis	Reference	Web site
FastQC	Quality control tool showing statics such as quality values, sequence length distribution and GC content distribution	[33]	http://www.bioinformatics.babraham.ac.uk/projects/fastqc/
FastQ Screen	Screen a library against sequence databases to see if composition of library matches expectations	[37]	http://www.bioinformatics.babraham.ac.uk/projects/fastq_screen
BBtools	BBDuk trims and filters reads using *k*-mers and entropy information. BBNorm normalizes coverage by down-sampling reads (digital normalization)	[35]	http://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/
Trimmomatic	Flexible read trimming tool for Illumina data	[36]	http://www.usadellab.org/cms/?page=trimmomatic
Cutadapt	Find and remove adapter sequences, primers, poly-A tails and other types of unwanted sequence	[34]	https://cutadapt.readthedocs.io
khmer/diginorm	Tools for *k*-mer error trimming of reads and digital normalization of samples	[38, 39]	http://khmer.readthedocs.io
MultiQC	Summarize results from different analysis (such as FastQC) into one report	[40]	http://multiqc.info

Note: Most of these tools can also be used for other types of genome sequence data, e.g. whole-genome or RNA-seq data.

easily. If the microbiome comes from a host with a sequenced genome, such as human, it is useful to identify and filter out host reads before further analysis. Alternatively, some taxonomic classifiers can include the host genome in their databases.

After quality control, the reads can either be assembled into longer contiguous sequences called contigs or passed directly to taxonomic classifiers (Figure 1). Taxonomic classification of every read is a form of binning because it groups reads into bins corresponding to their taxon ID. Binning can also be done using other properties such as composition and co-abundance profiles, although those methods typically require assembly of reads into longer contigs, which provide better statistics for profiling [41]. (See [42] for a review of binning methods.) When the analysis only returns the estimated abundances of the different taxa (instead of a classification of each read), we call it

taxonomic profiling. The choice of assembly-based analyses versus direct taxonomic classification of reads depends on the research question.

Direct taxonomic classification is useful for quantitative community profiling and identification of organisms with close relatives in the database. Compared with marker gene-based community profiling, metagenomic shotgun sequencing alleviates biases from primer choice and enables the detection of organisms across all domains of life, assuming that DNA can be extracted from the target environment. Researchers can quantify the structure of microbial communities using ecological and biogeographic measures such as species diversity, richness and uniformity of the communities [22, 43]. In clinical microbiology, the focus is often on the presence or absence of infectious pathogens, which can be identified by matching reads against a reference database [44–47]. Even though human-associated microbes are comparatively well studied with many completed genomes in the reference database, some pathogens remain unsequenced, and others have only recently been discovered using metagenomics sequencing [48–52]. Insights into the functional potential of a microbiome can be gained by matching the reads against pathway or gene databases [53, 54]. Further discussion of functional analysis in metagenomics and metatranscriptomics can be found in [55].

When no close relative of a species is in the database, as often happens with samples from unexplored ecological niches, assembly and binning of the reads may be useful first steps in the analysis. Analysis of the binned draft genomes allows for a more qualitative understanding of the physiology of the uncultivated microbes. By identifying single-copy and conserved genes in the contig bins, taxonomy, genome completeness as well as contamination can be assessed [41, 56]. Some recent findings from metagenomic (draft) assemblies include the identification of the enzymes used for oil and paraffin degradation by *Smithella* spp. [57–59] and insights into metabolic pathways and interactions between microbes in methanogenic bioreactors [60].

Metagenomic classification

Metagenomic classification tools match sequences—typically reads or assembled contigs—against a database of microbial genomes to identify the taxon of each sequence. In the early days of metagenomics, the best strategy was to use BLAST [61] to compare each read with all sequences in GenBank. As the reference databases and the size of sequencing data sets have grown, alignment using BLAST has become computationally infeasible, leading to the development of metagenomics classifiers that provide much faster results, although usually with less sensitivity than BLAST. Some programs return an assignment of every read, while others only provide the overall composition of the sample. A variety of strategies have been used for the matching step: aligning reads, mapping k-mers, using complete genomes, aligning marker genes only or translating the DNA and aligning to protein sequences (Tables 3). Recent studies have attempted to benchmark the performance of metagenomics classifiers based on both accuracy and speed [62, 63], although these studies are limited by their (unavoidable) reliance on simulated data.

Taxonomic profiling with marker gene-based and other approaches

Marker gene approaches identify sets of clade-specific, single-copy genes, so that the identification of one of these genes can be used as evidence that a member of the associated clade is present. This allows faster assignment because the database, even with a million or more genes (as in MetaPhlAn [81]), is far smaller than a database containing the full genomes for all species. The assignment can then be made with fast, sensitive aligners, such as Bowtie2 [85] used by MetaPhlAn and HMMER [86] used by Phylosift [87] and mOTU [82]. GOTTCHA [76] generates a database with unique genome signatures based on unique 24 base-pair fragments, which it indexes with bwa-mem [88]. GOTTCHA can output either binary classification (presence/absence calls) or a taxonomic profile, which is based on coverage of the genomic signatures, The use of single-copy marker genes should in principle make abundance estimation more precise, although it is impossible to know the copy number of a gene for a species with an incomplete genome. Because marker gene methods identify only a few genes per genome, most of the reads in a sample do not receive a classification at all; instead, these algorithms provide the microbial composition, expressed in terms of relative abundance for all taxa that they recognize in the sample.

An alternative approach for metagenomics profiling is using the overlap of MinHash signatures [89] as implemented in Mash [83] and sourmash [84]. MinHashes allow one to estimate the similarity of data sets extremely efficiently, e.g. the overlap between all microbial genomes in GenBank and a metagenomics data set. The MinHash search databases are small and fast to build and search, allowing searches against the entire GenBank database on a laptop.

Nucleotide taxonomic classification and quantification

Kraken [64] was the first method to provide fast identification of all reads in a metagenomic sample. It accomplishes this using an algorithm that relies on exact k-mer matches, replacing alignment (which requires more computational work) with a simple table lookup. Kraken constructs a database that stores, with every k-mer in every genome, the species identifier (taxonomy ID) for that k-mer. When a k-mer is found in two or more taxa, Kraken stores the lowest-common ancestor (LCA) of those taxa with that k-mer. Database k-mers and their taxa are saved in a compressed lookup table that can be rapidly queried for exact matches to k-mers found in the reads (or contigs) of a metagenomics data set. CLARK [65] uses a similar approach, building databases of species- or genus-level specific k-mers, and discarding any k-mers mapping to higher levels. Both Kraken and CLARK set $k = 31$ by default, although the database can be built with any length k-mer. The selection of k reflects an important trade-off between sensitivity and specificity: excessively long k-mers may fail to match because of sequencing errors or genuine differences among species and strains, while overly short k-mers will yield nonspecific (and false) matches to many genomes. An alternative approach to using fixed k-mers is spaced or adaptive (variable-length) seeds, which encode patterns for which only a subset of the bases has to match perfectly [90–92]. An extension of Kraken using spaced seeds shows somewhat better accuracy for family and genus-level classification, but lower precision at the species level [93]. A similar extension was developed for CLARK [66]. Note that Kraken maps reads to the taxonomic tree, not to a specific level such as species or genus. Bracken [94] is an extension of Kraken that estimates species- or genus-level abundance based on a Bayesian probability algorithm. The Livermore Metagenomics Analysis Toolkit (LMAT) [77] is a k-mer-based classifier that uses a smaller default k-mer size ($k = 20$) than Kraken and CLARK, but

Table 3. Metagenomic classifiers, aligners and profilers

Tool	Synopsis	Reference	Web site
Kraken	Fast taxonomic classifier using in-memory *k*-mer search of metagenomics reads against a database built from multiple genomes	[64]	https://ccb.jhu.edu/software/kraken/
Kraken-HLL	Extension of Kraken counting unique *k*-mers for taxa and allowing multiple databases		https://github.com/fbreitwieser/kraken-hll
CLARK(-S)	Fast taxonomic classifier using in-memory *k*-mer search of metagenomics reads against a database built from completed genomes. S extension uses spaced *k*-mer seeds for better classification	[65, 66]	http://clark.cs.ucr.edu
Kallisto	Taxonomic profiler using pseudo-alignment with *k*-mers using techniques based on transcript (RNA-seq) quantification	[67]	https://github.com/pachterlab/kallisto
k-SLAM	Taxonomic classifier using database of nonoverlapping *k*-mers in genomes. Reads are split into *k*-mers, and overlaps found by lexicographical ordering are pseudo-assembled	[68]	https://github.com/aindj/k-SLAM
Kaiju	Fast taxonomic classifier against protein sequences using FM-index with reduced amino acid alphabet	[69]	https://github.com/bioinformatics-centre/kaiju
DIAMOND	Protein homology search using spaced seeds with a reduced amino acid alphabet, 2000–20 000 times faster than BLASTX	[70]	https://github.com/bbuchfink/diamond
BLAST+	Highly sensitive nucleotide and translated-nucleotide protein alignment	[61, 71]	https://blast.ncbi.nlm.nih.gov
MEGAN6/CE	Desktop and Web metagenomics analysis suite. Uses BLAST or diamond to match sequences and assigns LCA of matches	[72, 73]	http://ab.inf.uni-tuebingen.de/software/megan6/
DUDes	Top-down assignment of metagenomics reads	[74]	https://sourceforge.net/projects/dudes/
Taxonomer	Web-based metagenomics classifier including binning and visualization	[75]	http://taxonomer.iobio.io/
GOTTCHA	Taxonomic profiler that maps reads against short unique subsequences ('signature') at multiple taxonomic ranks	[76]	http://lanl-bioinformatics.github.io/GOTTCHA/
LMAT(-ML)	K-mer-based taxonomic read classifier using extensive database including draft genomes and eukaryotes. ML (Marker Library) extension reduces RAM requirements by stringent pruning of non-informative and overlapping *k*-mers	[77, 78]	https://sourceforge.net/projects/lmat/
taxator-tk	Uses BLAST or LAST output for binning and taxonomic assignment via overlapping regions and pairwise distance measures	[79]	https://github.com/fungs/taxator-tk
Centrifuge	Fast taxonomic classifier using database compressed with FM-index, database and output format similar to Kraken	[80]	http://ccb.jhu.edu/software/centrifuge/
MetaPhlAn 2	Marker gene-based taxonomic profiler	[81]	https://bitbucket.org/biobakery/metaphlan2
mOTU	Taxonomic profiler based on a set of 40 prokaryotic marker genes	[82]	http://www.bork.embl.de/software/mOTU/
Mash	MinHash-based taxonomic profiler enabling superfast overlap estimations	[83]	http://mash.readthedocs.io
sourmash	Alternative implementation of MinHash algorithm using fast searches with sequence bloom trees for taxonomic profiling	[84]	https://github.com/dib-lab/sourmash
PanPhlAn	Pan-genome-based phylogenomic analysis	[2]	http://segatalab.cibio.unitn.it/tools/panphlan/

stores the list of source genomes with each *k*-mer instead of their lowest common taxonomic ancestor. LMAT includes microbial draft genomes as well as eukaryotic microbes in its 'Grand' database, which requires 500 GB RAM and classifies more reads than a database without draft genomes. LMAT-ML (for Marker Library) [78] implements more stringent *k*-mer pruning to retain only informative and nonoverlapping *k*-mers, which reduces the memory requirements to just 16 GB.

K-mers can also be represented in de Bruijn graphs. Kallisto [67, 95], which was originally developed for RNA-Seq analysis, uses a colored de Bruijn graph [96] in which each edge (i.e. *k*-mer) is assigned a set of 'colors', where a color encodes a genome in which the *k*-mer has been found. Given a sample read, Kallisto finds approximately matching paths in the colored de Bruijn graph, an approach the authors term 'pseudo-alignment'. After mapping, each read has a set of genomes associated with

it. Kallisto then infers strain abundances using an expectation–maximization (EM) algorithm [67]. k-SLAM [68] is a novel k-mer-based approach that uses local sequence alignments and pseudo-assembly, which generates contigs that can lead to more specific assignments.

Centrifuge [80] is a fast and accurate metagenomics classifier using the Burrows–Wheeler transform (BWT) and an FM-index to store and index the genome database. This strategy uses only about one-tenth the space of a Kraken index for the same database. Centrifuge also implements a feature that combines shared sequences from closely related genomes using MUMmer [97]. This greatly reduces redundancy for species where dozens of strains have been sequenced, further reducing the size of the index data structure.

MEGAN6/MEGAN-CE [73] and taxator-tk [79] both use the output of a local sequence aligner such as BLAST [61, 71], DIAMOND [70] or LAST [91]. MEGAN uses the LCA of the alignment results as its taxonomic assignment. A Web-based interface allows interactive exploration and functional analysis of its results. Taxator-tk first merges overlapping regions from the query (found by the local alignment) into larger subsequences. The pairwise distances of the subsequences to reference genomes are determined and used for binning and taxonomic assignment.

DUDes [74] computes taxonomic abundances from output of read aligners such as bwa-mem [88]. DUDes resolves ambiguities in mapping using an iterative approach that analyzes the read coverage of nodes in the taxonomic tree top-down, and uses permutation tests to select significant tree nodes. The algorithm can report multiple probable candidate strains or select the best candidate, instead of reporting just their LCA.

Taxonomer [75] provides a Web-based interface that enables fast classification of most reads. Taxonomer achieves fast classification by first binning reads into broad categories, and then classifying human, bacterial and fungal rRNA, labeling other reads as unknown. The visualization presents the results in interactive sunburst diagrams and enables the download of BIOM-formatted reports.

Fast amino acid database searches

Amino acid sequences are conserved at much greater evolutionary distances than DNA sequences, and this property can be exploited for more sensitive read classification, although the alignment step is slower. Both DIAMOND [70] and Kaiju [69] take this approach, comparing the six-frame translations of reads against protein databases. DIAMOND uses double-indexing of both a reference protein database and the translated sample reads. Each index contains seed-location pairs, where each seed is an amino acid fragment. After lexicographically ordering each index, DIAMOND traverses both lists in parallel to find matches between the database and the sample. For every match, DIAMOND attempts to align the sequencing read against the database protein and reports high-scoring matches. MEGAN [72] calculates taxonomic composition of samples based on BLAST or DIAMOND results using the LCA approach of multi-matching sequences.

Kaiju indexes the reference protein database using a BWT and saving each sequence in an FM-index table. This efficient database structure, similar to the one used in centrifuge (described above), allows metagenomic sequences to be searched against a large protein database. Given a metagenomic sample and the pre-built index, Kaiju first translates every read in all six reading frames, splitting the read at stop codons. Kaiju sorts all of the resulting protein fragments by length and compares each against the protein database, longest to shortest, finding and returning maximum exact matches.

Metagenomic assembly

Illumina sequencing technology, which is the most widely used sequencing method for metagenomics experiments today, generates read lengths in the range of 100–250 bp, with a typical sequencing run producing tens of millions of reads. Metagenomics experiments might generate hundreds of millions or even billions of reads from a single sample. Depending on number of reads and the complexity of the microbial species in the sample, some genomes might be sequenced deeply, allowing the experimenter to try to assemble the original genome sequence, or parts of it, from the short reads.

Genome assembly is a challenging problem, even for single genomes [98]; assembly of a mixed sample with many species in different abundances, as is necessary for a metagenomics sample, is even more complicated, requiring special-purpose assembly algorithms, reviewed and compared in [99, 100]. Perhaps, the biggest problem is the highly uneven sequencing depth of different organisms in a metagenomics sample. Standard assemblers assume that depth of coverage is approximately uniform across a genome; this assumption helps the algorithm in resolving repeats as well as removes erroneous reads. Relaxing this assumption means that any techniques within the assembler that rely on depth of coverage will no longer work.

A second issue that makes metagenomics assembly harder is the nonclonal nature of the organisms within a sample. For bacterial assembly (and for some eukaryotic assemblies), the source DNA can be grown up clonally, allowing the assembly algorithm to impose strict requirements for the percent identity between overlapping reads. In this context, lower sequence identity between two reads implies that they came from two slightly divergent copies of a repeat in the genome. In a metagenomics sample, between-strain differences can look exactly the same as variation between repeats.

Third, the depth of coverage of a particular species is rarely high, unless that species is present in high quantities in the sample. Even with tens of millions of reads, a metagenomics sample is not likely to contain deep coverage of more than one or two species, unless the sample itself is simple, i.e. containing only a few species. These and other issues mean that the results of metagenomics assembly will never be as good as those from assembly of a single, clonal organism.

Nonetheless, assembly and binning of a metagenomics sample often succeed in merging many of the reads, resulting in contigs that are easier to align to a genome database or analyze without alignment. Here, we list current assemblers and contig binners that have been designed for metagenomics, also summarized in Table 4. An overview of the techniques used in assembly is given in [41, 98, 99]. For more discussion on contig binning and curation and validation of reconstructed genome bins, see [41].

Assembly of reads into longer contiguous sequences (contigs)

MetaVelvet [106] and Ray Meta [104] are single k-mer de Bruijn graph assemblers for metagenomics data. MetaVelvet is an extension of the Velvet assembler [124] that decomposes the single de Bruijn graph into multiple subgraphs (ideally

Table 4. Tools for whole-genome assembly and metagenomics assembly

Tool	Synopsis	Reference	Web site
Megahit	Co-assembly of metagenomic reads with variable *k*-mer lengths and low memory usage	[101]	https://github.com/voutcn/megahit
SPAdes	DBG assembler using multiple *k*-mers, works also for simple metagenomes	[102]	http://cab.spbu.ru/software/spades
MetaSPAdes	Extension of SPADES with better assemblies with different abundances, conserved regions and strain mixtures	[103]	http://cab.spbu.ru/software/spades/
Ray Meta	DBG assembler with fixed *k*-mer size	[104]	http://denovoassembler.sourceforge.net/
MetaVelvet(-SL)	DBG assembler using fixed *k*-mer size. SL extension identifies and splits chimeric nodes	[105, 106]	http://metavelvet.dna.bio.keio.ac.jp
IDBA-UD	DBG assembler using multiple *k*-mer sizes, analyzes coverages between paths to give better assemblies in complex metagenomes with uneven coverage	[107]	http://i.cs.hku.hk/~alse/hkubrg/projects/idba_ud/
MetAMOS	Framework for metagenomic assembly, analysis and validation	[108]	http://metamos.readthedocs.io
MOCAT2	Pipeline for read filtering, taxonomic profiling, assembly, gene prediction and functional analysis	[109]	http://mocat.embl.de/
Anvi'o	Analysis and visualization platform for metagenomics assembly and binning	[110]	http://merenlab.org/software/anvio/
Contig binning			
MaxBin	Efficient binning of metagenomic contigs based on EM algorithm using nucleotide composition	[111]	https://downloads.jbei.org/data/microbial_communities/MaxBin/MaxBin.html
CONCOCT	Bins contigs using nucleotide composition, coverage data in multiple samples and paired-end read information	[112]	https://github.com/BinPro/CONCOCT
COCACOLA	Binning contigs in using read coverage, correlation, sequence composition and paired-end read linkage	[113]	https://github.com/younglululu/COCACOLA
MetaBAT	Metagenome binning with abundance and tetra-nucleotide frequencies	[114]	https://bitbucket.org/berkeleylab/metabat
VizBin	Visualization of metagenomic data based on nonlinear dimension reduction	[115]	http://claczny.github.io/VizBin/
AbundanceBin	Binning method based on *k*-mer frequency in reads	[116]	http://omics.informatics.indiana.edu/AbundanceBin/
GroopM	Identifies population genomes using differential coverage of contigs	[117]	http://ecogenomics.github.io/GroopM/
MetaCluster	Read and contig binning in two rounds for low- and high-abundance organisms using various *k*-mer lengths	[118, 119]	http://i.cs.hku.hk/~alse/MetaCluster/
PhyloPythiaS(+)	Assigns contigs to taxonomic bin using support vector machine trained on reference sequences	[120, 121]	https://github.com/algbioi/ppsp/wiki
Assembly and binning quality assessment			
MetaQuast	Evaluate and compare metagenomics assemblies based on alignments with reference genomes	[122]	http://quast.sourceforge.net/metaquast
BUSCO	Assess genome assembly and gene set completeness based on single-copy orthologs, also for eukaryotes	[123]	http://busco.ezlab.org/
CheckM	Tools for assessing quality of (meta)genomic assemblies providing genome completion and contamination estimates, especially for bacteria and viruses	[56]	http://ecogenomics.github.io/CheckM/

Note: DBG, de Bruijn graph.

corresponding to different organisms) based on coverage information and graph connectivity. MetaVelvet-SL [105] improves the splitting of chimeric nodes—nodes that are shared between subgraphs of closely related species—and thus generates longer scaffolds than MetaVelvet. Ray Meta, conversely, constructs contigs by a heuristics-guided graph traversal.

The choice of *k* is important for single *k*-mer de Bruijn graph assemblers. Small *k*'s are more sensitive in making connections, but fail to resolve repeats. Large *k*'s may miss connections and are more sensitive to sequencing errors, but usually create longer contigs. Most current metagenomics assemblers thus generate contigs from iteratively constructed and refined de Bruijn graphs using multiple *k*-mer lengths. The IDBA assembler (Iterative De Bruijn Graph Assembler) [125] first implemented this approach going from small *k*'s to large *k*'s, replacing reads with preassembled contigs at each iteration. IDBA-UD [107] is a version of the IDBA assembler modified to tolerate uneven depth of coverage, as occurring in single-cell and metagenomics sequencing experiments. IDBA-UD first generates a de Bruijn graph from the reads using small *k*-mers (by default $k = 20$), and—after error correction—extracts contigs that are used as 'reads' in the graph construction with the next-higher *k*-mer size. IDBA-UD detects erroneous *k*-mers and *k*-mers from different genomes by looking at deviations from the average multiplicity of *k*-mers in a contig. This local thresholding allows IDBA-UD to more accurately decompose the de Bruijn graph.

MetaSPAdes [103] is an extension of the SPAdes assembler [102], which was originally developed for bacterial genome and single-cell sequencing assembly. SPAdes/MetaSPAdes use an approach similar to IDBA with iterative de Bruijn graph refinement, but keeping the complete read information together with preassembled contigs at each step. MetaSPAdes implements various heuristics for graph simplification, filtering and storage to allow the assembly of large metagenomics data sets. Importantly, MetaSPAdes uses 'strain-contigs' to inform the assembly of high-quality consensus backbone sequences, which are often longer than contigs from other assemblers [126].

Megahit [101] is a fast assembler that uses a range of *k*-mers for iteratively improving the assembly. Megahit (which works for both metagenomics and single-genome sequencing data) uses a memory-efficient succinct de Bruijn graph representation [127] and can optionally run on CUDA-enabled graphics processing units in the graph construction step. By default, Megahit only keeps highly reliable *k*-mers that appear more than once, but implements a strategy to recover low-depth edges by taking additional *k*-mers from high-quality reads, which increases the contiguity of low-depth regions ('mercy k-mers').

The aforementioned assemblers are for the short, accurate reads generated by Illumina sequencers. Long-read sequencing technologies by Pacific Biosciences and Oxford Nanopore, with read lengths sometimes exceeding 10 000 bp, have great promise for microbial whole-genome sequencing [128], and are now being applied for metagenomics assembly in low-diversity communities [129]. While their lower throughput may limit their usefulness for complex metagenomes in the near future, they are revolutionizing the assembly and structural variant analysis of single genomes. As their throughput improves, these technologies have tremendous potential for metagenomic analysis as well.

Binning of contigs from closely related organisms

Short read metagenome assemblies are often highly fragmented because of low coverage and interstrain variation, as explained above. Binning algorithms attempt to group contigs or scaffolds from the same or closely related organisms [41, 130], and subsequent analysis, such as taxonomic assignment and functional analysis, is then done on the bins instead of individual contigs [41]. Binning has been shown to cluster contigs even from rare species and can recover draft genomes from previously uncultivated bacteria [131]. The bins are sometimes referred to as 'population genomes', as the unsupervised binning usually cannot distinguish the genetic content of closely related organisms (strains) in complex microbial communities.

Binning algorithms can use taxonomic information from a reference database (taxonomy-dependent or supervised binning), or they can cluster sequences using statistical properties and/or contig coverage (unsupervised binning). Many current methods use a combination of these features. For supervised taxonomy-dependent binning, some of the methods described in the previous section on metagenomics classification can be used. When classifying contigs instead of reads, the search space is much smaller, and slower alignment or phylogenetic methods can be used. For example, taxator-tk [79] uses BLAST, PhyloSift [87] searches for similarities to marker genes using Hidden Markov model profiles with HMMER and PhyloPythiaS(+) [120, 121] assigns reads to bins using a support vector machine model trained on reference sequences.

Taxonomy-independent binning does not require prior knowledge about the genomes in a sample, but relies on features inherent to the sequence set. Composition-based binning is based on the observation that overall genome composition in terms of G/C content and di- and higher-order nucleotide frequencies vary between organisms and are often characteristic of taxonomic lineages [132]. Clustering then can be done on sequence composition 'fingerprints' of the contigs [133]. MetaCluster [118, 119] bins reads by first grouping them based on long unique *k*-mers ($k > 36$) and merging groups based on tetranucleotide or pentanucleotide frequency distribution. MetaCluster 5.0 further uses 16-mer frequencies in a second round to bin contigs from low-abundance species in complex samples. VizBin [115] uses a dimensionality reduction mechanism based on self-organizing maps to visualize as well as cluster contigs into bins.

Composition-based binning methods usually require fairly large contigs (> 1–2 kb) to generate robust statistics. It can be difficult to separate contigs from closely related microorganisms whose nucleotide frequencies may be similar [134]. Some binning methods use coverage profiles across multiple samples, e.g. MGS Canopy [135] generates abundance profiles of gene calls and clusters them by co-abundance across samples. GroopM [117] identifies population genomes using differential coverage profiles of assembled contigs. CONCOCT [112] combines both tetranucleotide frequencies and differential abundances across multiple samples for binning. COCACOLA [113] works similarly to CONCOCT but using different distance metrics and different clustering rules. MetaBAT [114] calculates composite probabilistic distances incorporating models of interspecies and intraspecies distances that were trained on sequenced genomes. MaxBin 2.0 [111] estimates the number of bins by counting single-copy marker genes and iteratively refines binning using an EM algorithm with probabilistic distances.

After binning, reads can be mapped back to the bins, and each bin can reassembled, which has the potential to produce longer contigs if the binning was successful. Because each bin should contain only one taxonomic group, the reassembly can be done using either a specialized metagenomics assembler, such as those described above, or a single-genome assembler. Validation of the assembly and binning is an important step in metagenomic genome reconstruction. MetaQUAST [128] computes genome statistics of metagenomics assemblies, and, by aligning against reference genomes, can report the number of misassemblies and mismatches. CheckM [60] and BUSCO [129] estimate both the completeness as well as the contamination of recovered genomes using lineage-specific single-copy marker genes and single-copy orthologs, respectively. When marker genes are missing, the genome is probably not complete, and if marker genes are present multiple times, it suggests contamination.

Assembly pipelines and analysis tool sets

Metagenomics assembly is a complicated process, involving quality control, assembly, contig binning, mapping of reads back to contigs, reassembly, gene annotation and visualization. Several analysis pipelines and visualization tools have been developed to facilitate this process. MetAMOS [108] is a comprehensive pipeline for assembly and annotation of metagenomics samples. It can run multiple assemblers to create contigs and scaffolds. It then runs bacterial gene finders on the resulting contigs, and finally searches the predicted genes against a

protein database to assign names and functions wherever possible. Anvi'o [110] is another pipeline that combines assembly, alignment, binning and classification results in an interactive interface that allows one to refine the binning and assembly. MOCAT2 [109] integrates read filtering, taxonomic profiling with mOTU [82], assembly, gene prediction and annotation to output taxonomic as well as functional profiles of metagenomics samples.

Microbial taxonomy and genome resources and their impact on classification

Almost all of the methods described here rely on a database of genomes and on taxonomy of species. The accuracy and reliability metagenomics analysis relies critically on these data resources. Here, we discuss several issues about both the data themselves—the genomes—and the taxonomy that we use to name and group all living species.

The NCBI Taxonomy database [136] provides the standard nomenclature and hierarchical taxon tree for GenBank, EMBL and DDBJ (which mirror one another, and which together comprise the International Nucleotide Sequence Database Collaboration, INSDC [137]), and thus for most metagenomic classifiers. Metataxonomic classifiers, on the other hand, often use the SILVA, RDP and Greengenes databases of ribosomal genes which, somewhat confusingly, have their own taxonomies [138]. Every sequence deposited within an INSDC database has a taxon identifier based on species information provided by the depositor.

The hierarchical concept of the taxonomy is convenient for benchmarking metagenomics classifiers, but several issues can make evaluation difficult and even misleading. The taxonomy concept was originally developed for multicellular eukaryotes, primarily plants and animals, and a common definition of 'species' is a group of organisms that can interbreed and produce fertile offspring [139]. This definition clearly does not work for prokaryotes, which reproduce asexually and have no distinction between somatic and germ line cells. Making things more complicated is the (relatively rare) process of horizontal gene transfer, which in bacteria and archaea allows for the direct exchange of DNA across species barriers.

Metagenomics classifiers may incorporate assumptions that are violated by the taxonomy or by the genome data itself, which will result in sequences being assigned to the wrong taxonomic ID. Here, we discuss some examples of how this can happen.

The same taxonomic level can contain different levels of sequence similarity. Although the set of species under a phylum represents a much wider range of diversity than the species within a genus, the level of similarity at a specific level of the tree is highly variable. A comparison of bacterial genomes present in GenBank (as of September 2014) showed that 6% of genomes with different species assignments have an average nucleotide identity (ANI) >93%, while 15% of genomes within the same species have an ANI <93% [139]. For example, *Yersinia pseudotuberculosis* and *Yersinia pestis*, which represent two distinct species, are over 98.5% identical, but *Yersinia enterocolitica* is <86% identical to either of them. *Mycobacterium tuberculosis* and *Mycobacterium bovis* have >99.6% identity, while the ANI of *Mycobacterium leprae* with either of them is <85%. Notably, the close *Y. pestis* and *Y. pseudotuberculosis* species are grouped together in the 'species group' *Y. pseudotuberculosis* complex, and *M. tuberculosis* and *M. bovis* are grouped in the species group *M. tuberculosis* complex. A well-known example of historic misplacement is *Shigella* [140], a genus that clearly falls within the *E. coli* species with ANIs above 97%—much higher than the ANIs of, for example *Escherichia fergusonii* to *E. coli* of about 93%.

The consequence of this variability for computational classifiers is that at the species or genus rank, different levels of sequence similarity in different parts of the taxonomic tree have a different meaning, making it impossible, for some taxa, to design consistent rules assigning reads or contigs (even long ones) to a species, and there is clearly no fixed percent-identity threshold that can be used to group sequences into the same species or genus.

The fungal taxonomy sometimes has two species and taxonomy IDs for the same organism. Fungi can have both teleomorphic (sexual reproductive stage) and anamorphic (asexual reproductive stage) phases. Historically, different names were given to the same fungi in the different stages. For example, *Fusarium solani* is a filamentous fungus whose spores are found in soil and plant debris, and which can cause keratitis [104]. This fungus is assigned to two different species in the NCBI taxonomy database: the anamorph is called *F. solani* and has taxonomy ID 169388, while the teleomorph is called *Nectria haematococca* with taxonomy ID 140110. The taxons are both listed as species in the genus *Fusarium*, and some sequences in GenBank are assigned to one taxonomy ID, and others to the other. (As of 28 May 2017, there were 6765 nucleotide sequences for *F. solani* and 16 643 for *N. haematococca* in GenBank.) The rules have been since updated to reflect a 'one fungus, one name' system [141], but it may take a long time to resolve the current multiplicity of names [142]. As a consequence, metagenomics classifiers might assign sequences to either taxon—and both would be correct, even though they appear to be different species.

Historically, no official species names were given to unculturable bacteria. Bacterial nomenclature is governed by the International Code of Nomenclature of Bacteria. In 2001, it was decided that the designation of a new microbial species would require the identification of a type strain representing that species, and that the type strain had to be deposited in at least two different culture collections as pure (axenic) culture [143]. Most bacteria and archaea, though, cannot be cultured with current methods. All of these bacteria are given *Candidatus* names (i.e. the name Candidatus is prepended to the putative genus and species name) or are named only informally [144, 145], but are not covered by the standard nomenclature [146]. The NCBI taxon 'unclassified Bacteria', which contains several candidate divisions, is placed directly under the 'Bacteria' taxon node (see next paragraph). As of 28 May 2017, the NCBI taxonomy has 16 400 formal bacterial species and >280 000 informal ones.

Unclassified organism sequences and metagenomes are close to the root of the taxonomy. The NCBI databases contain sequences of bacteria, eukaryotes and viruses that thus far are not placed into the taxonomic hierarchy. As of 21 August 2017, NCBI had 2756 genomes for 'unclassified bacteria' (taxonomy ID 2323), 168 genomes for 'unclassified viruses' (taxonomy ID 12429) and 4 genomes for 'unclassified viruses' (taxonomy ID 12429). All these taxa are at high levels in the taxonomic tree, just below their superkingdoms. Furthermore, GenBank and the BLAST nr/nt database (https://www.ncbi.nlm.nih.gov/books/NBK62345/) contain thousands of 'unclassified' sequences (taxonomy ID 12908), especially from metagenomes (e.g. 'human gut microbiome', taxonomy ID 408170). Shared sequences of such taxa and properly placed organisms can present a challenge for metagenomics methods that attempt to cluster

together sequences or compute the lowest common ancestor. Especially when using the BLAST nr/nt or nr databases, it may be useful to filter unclassified sequences, or include only microbial taxa, as is done by the kaiju classifier [69] when including eukaryotes from nr.

Taxonomy changes. One solution to some of the problems just listed is to rename or move the species in the microbial taxonomy. This does happen somewhat frequently, but the new names do not automatically percolate outward to every resource that has downloaded the genomes from GenBank. As a result, some benchmark genome sets used in metagenomics comparisons [148] have become outdated because some of the organisms have new names. This in turn can lead to mistaken conclusions when later studies download and reuse the data without going back to retrieve the original genomes from GenBank. NCBI taxonomy does keep track of all previous names of a taxon via synonyms; however, the taxonomy is not versioned, which makes it difficult to track or refer to a specific version.

Viruses and viral taxonomy

Most of the comments about bacterial genomes and taxonomy apply equally well to viruses, which thus far we have not discussed. Viruses do not have universally conserved genes such as the 16S and 18S rRNA genes, making it far more difficult to conduct systematic surveys of diversity. Nonetheless, it appears that the number of diversity of viral species may far exceed those of bacteria. A recent paper, for example, used metagenomic sequencing to discover >125 000 new DNA viruses [149], most of which encode proteins that have no sequence similarity to known isolates. Another study mined public databases to discover >12 000 new viral genomes linked to bacterial and archaeal hosts [150]. Faced with this rapid growth in the variety of viral species, a scientific consortium recently proposed a new framework for incorporating viruses discovered through metagenomic sequencing into the official taxonomy of the International Committee on Taxonomy of Viruses [151].

The relatively sparse sampling of the viral microbiome means that most viral species cannot yet be recognized by alignment of metagenomic samples to databases. Viruses also mutate much more rapidly than bacteria, so even when a known virus is present, alignment algorithms may need to permit more mismatches to identify. These and other issues mean that metagenomic methods for viruses sometimes require different methods from bacteria, which are beyond the scope of this discussion; a recent review of such methods can be found in [152].

Microbial genome resources

The most commonly used reference genome databases are the complete and draft genomes at GenBank [153], which for more than a quarter century has been the repository for genome sequence data from around the world. Sequence records in GenBank are owned by the submitter, and only the submitter can update that. In the vast majority of cases, DNA sequence records are never altered after their original submission.

GenBank relies on correct taxonomic identification and annotation provided by the submitter. Some genomes in GenBank have an incorrect species name, presumably because of labeling errors for bacterial samples. When such an error is discovered, NCBI (the home of GenBank) can request the submitter to update the record, but if the submitter does not respond, then NCBI can only suppress or flag the entry [142]. To avoid such errors, NCBI now performs a variety of quality checks when genomes are submitted to make sure that submitted genomes are not assigned to the wrong species [153].

Table 5. Number of entries in commonly used reference databases

Domain	Level	Draft genomes		Complete genomes[1]	
		GenBank	RefSeq	GenBank	RefSeq
Archaea	Entries	859	351	260 (20)	225 (12)
	Species	695	204	209 (14)	178 (7)
Bacteria	Entries	89 730	78 783	7314 (1346)	6973 (1066)
	Species	19 078	11 217	2677 (542)	2586 (406)
Fungi	Entries	1897	191	28 (414)	7 (38)
	Species	997	190	17 (68)	7 (36)
Protists	Entries	430	47	2 (49)	2 (27)
	Species	226	47	2 (38)	2 (26)
Viruses	Entries	3	3	0 (0)	7214 (22)
	Species	1	3	0 (0)	7073 (22)

[1]Numbers in parentheses represent incomplete genome assemblies for which at least one chromosome was assembled. Data as of 27 May 2017.

An even bigger issue than incorrect species labels is contamination. The vast majority of genomes in GenBank today are 'draft' genomes (Table 5). These are genomes for which an assembly was generated from one or more sequencing data sets, but where most chromosomes are fragmented into many pieces. It is not uncommon for a draft genome to contain tens of thousands of such contigs. In any draft genome, some of the contigs might be contaminants, i.e. they might not belong to the species that was presumably sequenced, even though every contig is assigned to the same species. Common contaminants include sequencing vectors and adaptors, nucleic acids that are commonly present in laboratories such as from *E. coli* and PhiX174 (a phage used as Illumina sequencing control) and of course human DNA, which creeps into many sequencing projects by accident. If the laboratory that created the assembly did not screen out these contaminants, they are submitted to GenBank as part of the organism. GenBank itself runs a contaminant screen on all assemblies, and contigs that appear to be contaminants are reported back to the submitter, who is encouraged to remove them and resubmit. Despite the best efforts of GenBank curators, though, thousands of contaminants have already made their way into the draft genome data.

The result of these contaminants is that reads from a metagenomics project will match some draft genomes extremely well because the metagenomics project has some of the same contaminants (e.g. fragments of *E. coli* or human DNA). This in turn leads to incorrect taxonomic classification, even though the computational tools performed perfectly. For example, a strain of *Neisseria gonorrhoeae* was found to be contaminated with fragments of cow and sheep DNA [154], a problem that was discovered after a metagenomics study of the cow microbiome detected this particular *N. gonorrhoeae* strain and reported it to the authors of the Kraken program, who in turn discovered that the mistake was in the data, not the software.

RefSeq provides an alternative. The RefSeq project takes GenBank sequences and passes them through additional automated filters to produce a more curated genome resource [155]. RefSeq records are owned by NCBI and can be updated as needed to maintain annotation or to incorporate additional information. As shown in Table 5, ~79 000 of ~90 000 draft bacterial genomes are in RefSeq (data as of 27 May 2017). There are

various reasons why genomes may be excluded from RefSeq, e.g. the assemblies are too highly fragmented. For bacteria, currently, the most common reason that a GenBank genome is not included is that it is derived from a metagenome (about half of the excluded genomes). Note that this is a current policy and inclusion criteria may change in the future. The rate of inclusion into RefSeq has been much slower for eukaryotic microbes; currently, it contains only 191 of 1897 fungal genome assemblies. RefSeq also includes the viral domain, for which it validates and indexes one viral genome per species (and sometimes per serotype). As of May 2017, there are >7000 viral genomes in RefSeq. In addition, the NCBI Viral Genomes Resource (https://www.ncbi.nlm.nih.gov/genome/viruses/) [156] provides links to other validated viral genomes that are 'neighbors' (i.e. strains) of viral species in RefSeq.

Genomes are assigned to species or strains. Until 2014, every new microbial genome submitted to NCBI was assigned a new taxonomy ID, even if they were isolates of existing species. Owing to the dramatic increase in the number of genome sequences, this policy was changed in 2014, and since then only novel species and higher microbial orders get new taxonomy IDs [147]. Previously assigned strain taxonomy IDs remain in the database, which means that a single species may have genomes both at species and strain levels. For *E. coli*, for example, RefSeq contains 5596 genomes (as of 28 June 2017), of which 3292 have the taxonomy ID of *E. coli*, and the remainder have one of 2223 distinct strain-level taxonomy IDs. Overall, ~35% of the bacterial genomes in RefSeq and GenBank have strain-level IDs, and the remaining ~65% have species-level IDs. This can be challenging for algorithms that try to characterize metagenomic samples at the strain level.

Conclusions

Next-generation sequencing provides a powerful tool to study the microbes in, on, and around us. A great variety of computational tools have been developed to assist in the analysis of metagenomics data sets, which are large and constantly changing as the technology of sequencing improves. Here, we reviewed methods for classification and assembly of metagenomics data. Classification methods determine the mixture of species in a sample, either by using marker genes to estimate their abundance or by assigning a taxonomic identifier to every read. Assembly methods take the raw read data and assemble reads from the same species into larger contigs, which in turn can be assigned taxonomic labels. We also discussed some of the challenges presented by inconsistencies in microbial taxonomy itself, and by contamination in the draft genomes that almost all methods rely on. Many of these problems may be solved over time, but while the data are in a constant state of flux, users need to remain aware of these issues, so that they can avoid potential pitfalls when analyzing large, complex metagenomics data sets.

Key Points

- Classification methods for metagenomic reads rely on fast lookup algorithms to handle the enormous data sets generated by next-generation sequencing.
- Metagenomic assembly methods can reconstruct large sections of the genomes of some species in a microbial community, if the sequencing depth is sufficient.
- Genome databases are growing rapidly, but many draft genomes are contaminated with fragments of sequence from other species, which presents challenges for metagenomic analysis.
- Microbial taxonomy is rapidly changing in the genome era, with many species being renamed and grouped into different clades.

Funding

This work was supported in part by the US National Institutes of Health (NIH grants R01-HG006677 and R01-GM083873) and by the US Army Research Office (grant W911NF-1410490).

References

1. Marchesi JR, Ravel J. The vocabulary of microbiome research: a proposal. *Microbiome* 2015;**3**:31.
2. Scholz M, Ward DV, Pasolli E, *et al*. Strain-level microbial epidemiology and population genomics from shotgun metagenomics. *Nat Methods* 2016;**13**:435–8.
3. Moran MA, Satinsky B, Gifford SM, *et al*. Sizing up metatranscriptomics. *ISME J* 2013;**7**:237–43.
4. Baldrian P, López-Mondéjar R. Microbial genomics, transcriptomics and proteomics: new discoveries in decomposition research using complementary methods. *Appl Microbiol Biotechnol* 2014;**98**:1531–7.
5. Wilmes P, Heintz-Buschart A, Bond PL. A decade of metaproteomics: where we stand and what the future holds. *Proteomics* 2015;**15**:3409–17.
6. Beale DJ, Karpe AV, Ahmed W. Beyond metabolomics: a review of multi-omics-based approaches. In: DJ Beale, KA Kouremenos, EA Palombo (eds). *Microbial Metabolomics: Applications in Clinical, Environmental, and Industrial Microbiology*. Switzerland: Springer International Publishing, 2016, 289–312.
7. Franzosa EA, Hsu T, Sirota-Madi A, *et al*. Sequencing and beyond: integrating molecular 'omics' for microbial community profiling. *Nat Rev Microbiol* 2015;**13**:360–72.
8. Woese CR, Fox GE. Phylogenetic structure of the prokaryotic domain: the primary kingdoms. *Proc Natl Acad Sci USA* 1977;**74**:5088–90.
9. Schoch CL, Seifert KA, Huhndorf S, *et al*. Nuclear ribosomal internal transcribed spacer (ITS) region as a universal DNA barcode marker for Fungi. *Proc Natl Acad SciUSA* 2012;**109**:6241–6.
10. DeSantis TZ, Hugenholtz P, Larsen N, *et al*. Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Appl Environ Microbiol* 2006;**72**:5069–72.
11. Cole JR, Chai B, Farris RJ, *et al*. The Ribosomal Database Project (RDP-II): sequences and tools for high-throughput rRNA analysis. *Nucleic Acids Res* 2005;**33**:D294–6.
12. Carlton JM, Angiuoli SV, Suh BB, *et al*. Genome sequence and comparative analysis of the model rodent malaria parasite *Plasmodium yoelii yoelii*. *Nature* 2002;**419**:512–19.
13. Caporaso JG, Kuczynski J, Stombaugh J, *et al*. QIIME allows analysis of high-throughput community sequencing data. *Nat Methods* 2010;**7**:335–6.
14. Schloss PD, Westcott SL, Ryabin T, *et al*. Introducing mothur: open-source, platform-independent, community-supported software for describing and comparing microbial communities. *Appl Environ Microbiol* 2009;**75**:7537–41.
15. Edgar RC. UPARSE: highly accurate OTU sequences from microbial amplicon reads. *Nat Methods* 2013;**10**:996–8.

16. Li W, Godzik A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics* 2006;**22**:1658–9.
17. Mahe F, Rognes T, Quince C, *et al*. Swarm: robust and fast clustering method for amplicon-based studies. *PeerJ* 2014;**2**: e593.
18. Callahan BJ, McMurdie PJ, Rosen MJ, *et al*. DADA2: high-resolution sample inference from Illumina amplicon data. *Nat Methods* 2016;**13**:581–3.
19. Callahan BJ, Sankaran K, Fukuyama JA, *et al*. Bioconductor workflow for microbiome data analysis: from raw reads to community analyses. *F1000Res* 2016;**5**:1492.
20. McMurdie PJ, Holmes S. phyloseq: an R package for reproducible interactive analysis and graphics of microbiome census data. *PLoS One* 2013;**8**:e61217.
21. Siegwald L, Touzet H, Lemoine Y, *et al*. Assessment of common and emerging bioinformatics pipelines for targeted metagenomics. *PLoS One* 2017;**12**:e0169563.
22. Oulas A, Pavloudi C, Polymenakou P, *et al*. Metagenomics: tools and insights for analyzing next-generation sequencing data derived from biodiversity studies. *Bioinform Biol Insights* 2015;**9**:75–88.
23. D'Amore R, Ijaz UZ, Schirmer M, *et al*. A comprehensive benchmarking study of protocols and sequencing platforms for 16S rRNA community profiling. *BMC Genomics* 2016;**17**:55.
24. Kopylova E, Navas-Molina JA, Mercier C, *et al*. Open-source sequence clustering methods improve the state of the art. *mSystems* 2016;**1**:e00003-15.
25. Nguyen NP, Warnow T, Pop M, *et al*. A perspective on 16S rRNA operational taxonomic unit clustering using sequence similarity. *NPJ Biofilms Microbiomes* 2016;**2**:16004.
26. Brown CT, Hug LA, Thomas BC, *et al*. Unusual biology across a group comprising more than 15% of domain bacteria. *Nature* 2015;**523**:208–11.
27. Eloe-Fadrosh EA, Ivanova NN, Woyke T, *et al*. Metagenomics uncovers gaps in amplicon-based detection of microbial diversity. *Nat Microbiol* 2016;**1**:15032.
28. Shin J, Lee S, Go MJ, *et al*. Analysis of the mouse gut microbiome using full-length 16S rRNA amplicon sequencing. *Sci Rep* 2016;**6**:29681.
29. Salter SJ, Cox MJ, Turek EM, *et al*. Reagent and laboratory contamination can critically impact sequence-based microbiome analyses. *BMC Biol* 2014;**12**:87.
30. Brooks JP, Edwards DJ, Harwich MD, Jr, *et al*. The truth about metagenomics: quantifying and counteracting bias in 16S rRNA studies. *BMC Microbiol* 2015;**15**:66.
31. Tremblay J, Singh K, Fern A, *et al*. Primer and platform effects on 16S rRNA tag sequencing, Front. *Microbiol* 2015;**6**:771.
32. Clooney AG, Fouhy F, Sleator RD, *et al*. Comparing apples and oranges? Next generation sequencing and its impact on microbiome analysis. *PLoS One* 2016;**11**:e0148028.
33. Babraham Bioinformatics. FastQC a quality control tool for high throughput sequence data. http://www.bioinformatics.babraham.ac.uk/projects/fastqc/
34. Martin M. Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet J* 2011;**17**:10–12.
35. DOE Joint Genome Institute. BBDuk guide. http://jgi.doe.gov/data-and-tools/bbtools/bb-tools-user-guide/bbduk-guide/
36. Bolger AM, Lohse M, Usadel B. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics* 2014;**30**: 2114–20.
37. Babraham Bioinformatics. FastQ Screen. http://www.bioinformatics.babraham.ac.uk/projects/fastq_screen/
38. Titus Brown C, Howe A, Zhang Q, *et al*. A reference-free algorithm for computational normalization of shotgun sequencing data. *arXiv e-prints* 2012.
39. Crusoe MR, Alameldin HF, Awad S, *et al*. The khmer software package: enabling efficient nucleotide sequence analysis, *F1000Rese* 2015;**4**:900.
40. Ewels P, Magnusson M, Lundin S, *et al*. MultiQC: summarize analysis results for multiple tools and samples in a single report. *Bioinformatics* 2016;**32**:3047–8.
41. Sangwan N, Xia F, Gilbert JA. Recovering complete and draft population genomes from metagenome datasets. *Microbiome* 2016;**4**:8.
42. Mande SS, Mohammed MH, Ghosh TS. Classification of metagenomic sequences: methods and challenges. *Brief Bioinform* 2012;**13**:669–81.
43. Chiarucci A, Bacaro G, Scheiner SM. Old and new challenges in using species diversity for assessing biodiversity. *Philos Trans R Soc Lond B Biol Sci* 2011;**366**:2426–37.
44. Langelier C, Zinter MS, Kalantar K, *et al*. Metagenomic sequencing detects respiratory pathogens in hematopoietic cellular transplant patients. *Am J Respir Crit Care Med* 2017, [Epub ahead of print].
45. Salzberg SL, Breitwieser FP, Kumar A, *et al*. Next-generation sequencing in neuropathologic diagnosis of infections of the nervous system. *Neurol Neuroimmunol Neuroinflamm* 2016;**3**:e251.
46. Breitwieser FP, Pardo CA, Salzberg SL. Re-analysis of metagenomic sequences from acute flaccid myelitis patients reveals alternatives to enterovirus D68 infection. *F1000Res* 2015;**4**:180.
47. Schlaberg R, Chiu CY, Miller S, *et al*. Validation of metagenomic next-generation sequencing tests for universal pathogen detection. *Arch Pathol Lab Med* 2017;**141**:776–86.
48. Greninger AL, Messacar K, Dunnebacke T, *et al*. Clinical metagenomic identification of *Balamuthia mandrillaris* encephalitis and assembly of the draft genome: the continuing case for reference genome sequencing. *Genome Med* 2015;**7**: 113.
49. Mongkolrattanothai K, Naccache SN, Bender JM, *et al*. Neurobrucellosis: unexpected answer from metagenomic next-generation sequencing. *J Pediatric Infect Dis Soc* 2017: piw066.
50. Kandathil AJ, Breitwieser FP, Sachithanandham J, *et al*. Presence of Human Hepegivirus-1 in a cohort of people who inject drugs. *Ann Intern Med* 2017;**167**:1–7.
51. Cuestas ML. New virus discovered in blood supply: Human Hepegivirus-1 (HHpgV-1). *Rev Argent Microbiol* 2016;**48**:180–1.
52. Berg MG, Lee D, Coller K, *et al*. Discovery of a novel human pegivirus in blood associated with hepatitis C virus co-infection. *PLoS Pathog* 2015;**11**:e1005325.
53. Truong DT, Tett A, Pasolli E, *et al*. Microbial strain-level population structure and genetic diversity from metagenomes. *Genome Res* 2017;**27**:626–38.
54. Hahn AS, Altman T, Konwar KM, *et al*. A geographically-diverse collection of 418 human gut microbiome pathway genome databases. *Sci Data* 2017;**4**:170035.
55. Niu SY, Yang J, McDermaid A, *et al*. Bioinformatics tools for quantitative and functional metagenome and metatranscriptome data analysis in microbes. *Brief Bioinform* 2017: bbx051.
56. Parks DH, Imelfort M, Skennerton CT, *et al*. CheckM: assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. *Genome Res* 2015;**25**: 1043–55.

57. Tan B, de Araujo E Silva R, Rozycki T, *et al*. Draft genome sequences of three Smithella spp. obtained from a methanogenic alkane-degrading culture and oil field produced water. *Genome Announc* 2014;**2**:e01085-14.
58. Tan B, Nesbo C, Foght J. Re-analysis of omics data indicates Smithella may degrade alkanes by addition to fumarate under methanogenic conditions. *ISME J* 2014;**8**:2353–6.
59. Wawrik B, Marks CR, Davidova IA, *et al*. Methanogenic paraffin degradation proceeds via alkane addition to fumarate by 'Smithella' spp. mediated by a syntrophic coupling with hydrogenotrophic methanogens. *Environ Microbiol* 2016;**18**:2604–19.
60. Nobu MK, Narihiro T, Rinke C, *et al*. Microbial dark matter ecogenomics reveals complex synergistic networks in a methanogenic bioreactor. *ISME J* 2015;**9**:1710–22.
61. Altschul SF, Gish W, Miller W, *et al*. Basic local alignment search tool. *J Mol Biol* 1990;**215**:403–10.
62. Lindgreen S, Adair KL, Gardner PP. An evaluation of the accuracy and speed of metagenome analysis tools. *Sci Rep* 2016;**6**:19233.
63. Kelley DR, Salzberg SL. Clustering metagenomic sequences with interpolated Markov models. *BMC Bioinformatics* 2010;**11**:544.
64. Wood DE, Salzberg SL. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biol* 2014;**15**:R46.
65. Ounit R, Wanamaker S, Close TJ, *et al*. CLARK: fast and accurate classification of metagenomic and genomic sequences using discriminative k-mers. *BMC Genomics* 2015;**16**:236.
66. Ounit R, Lonardi S. Higher classification sensitivity of short metagenomic reads with CLARK-S. *Bioinformatics* 2016;**32**:3823–5.
67. Bray NL, Pimentel H, Melsted P, *et al*. Near-optimal probabilistic RNA-seq quantification. *Nat Biotechnol* 2016;**34**:525–7.
68. Ainsworth D, Sternberg MJE, Raczy C, *et al*. k-SLAM: accurate and ultra-fast taxonomic classification and gene identification for large metagenomic data sets. *Nucleic Acids Res* 2017;**45**:1649–56.
69. Menzel P, Ng KL, Krogh A. Fast and sensitive taxonomic classification for metagenomics with Kaiju. *Nat Commun* 2016;**7**:11257.
70. Buchfink B, Xie C, Huson DH. Fast and sensitive protein alignment using DIAMOND. *Nat Methods* 2015;**12**:59–60.
71. Camacho C, Coulouris G, Avagyan V, *et al*. BLAST+: architecture and applications. *BMC Bioinformatics* 2009;**10**:421.
72. Huson DH, Auch AF, Qi J, *et al*. MEGAN analysis of metagenomic data. *Genome Res* 2007;**17**:377–86.
73. Huson DH, Beier S, Flade I, *et al*. MEGAN community edition—interactive exploration and analysis of large-scale microbiome sequencing data. *PLoS Comput Biol* 2016;**12**:e1004957.
74. Piro VC, Lindner MS, Renard BY. DUDes: a top-down taxonomic profiler for metagenomics. *Bioinformatics* 2016;**32**:2272–80.
75. Flygare S, Simmon K, Miller C, *et al*. Taxonomer: an interactive metagenomics analysis portal for universal pathogen detection and host mRNA expression profiling. *Genome Biol* 2016;**17**:111.
76. Freitas TA, Li PE, Scholz MB, *et al*. Accurate read-based metagenome characterization using a hierarchical suite of unique signatures. *Nucleic Acids Res* 2015;**43**:e69.
77. Ames SK, Hysom DA, Gardner SN, *et al*. Scalable metagenomic taxonomy classification using a reference genome database. *Bioinformatics* 2013;**29**:2253–60.
78. Gardner SN, Ames SK, Gokhale MB, *et al*. Searching more genomic sequence with less memory for fast and accurate metagenomic profiling. *bioRxiv* 2016.
79. Droge J, Gregor I, McHardy AC. Taxator-tk: precise taxonomic assignment of metagenomes by fast approximation of evolutionary neighborhoods. *Bioinformatics* 2015;**31**:817–24.
80. Kim D, Song L, Breitwieser FP, *et al*. Centrifuge: rapid and sensitive classification of metagenomic sequences. *Genome Res* 2016;**26**:1721–9.
81. Truong DT, Franzosa EA, Tickle TL, *et al*. MetaPhlAn2 for enhanced metagenomic taxonomic profiling. *Nat Methods* 2015;**12**:902–3.
82. Sunagawa S, Mende DR, Zeller G, *et al*. Metagenomic species profiling using universal phylogenetic marker genes. *Nat Methods* 2013;**10**:1196–9.
83. Ondov BD, Treangen TJ, Melsted P, *et al*. Mash: fast genome and metagenome distance estimation using MinHash. *Genome Biol* 2016;**17**:132.
84. Titus Brown C, Irber L. Sourmash: a library for MinHash sketching of DNA. *J Open Source Softw* 2016;**1**.
85. Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. *Nat Methods* 2012;**9**:357–9.
86. Eddy SR. Accelerated profile HMM searches. *PLoS Comput Biol* 2011;**7**:e1002195.
87. Darling AE, Jospin G, Lowe E, *et al*. PhyloSift: phylogenetic analysis of genomes and metagenomes. *PeerJ* 2014;**2**:e243.
88. Li H, Durbin R. Fast and accurate long-read alignment with Burrows-Wheeler transform. *Bioinformatics* 2010;**26**:589–95.
89. Broder AZ. *On the Resemblance and Containment of Documents*. Palo Alto, CA: Digital Systems Research Center, 1998, 21–29.
90. Ma B, Tromp J, Li M. PatternHunter: faster and more sensitive homology search. *Bioinformatics* 2002;**18**:440–5.
91. Kielbasa SM, Wan R, Sato K, *et al*. Adaptive seeds tame genomic sequence comparison. *Genome Res* 2011;**21**:487–93.
92. Noé L, Martin DEK. A coverage criterion for spaced seeds and its applications to support vector machine string kernels and k-mer distances. *J Comput Biol* 2014;**21**:947–63.
93. Břinda K, Sykulski M, Kucherov G. Spaced seeds improve k-mer-based metagenomic classification. *Bioinformatics* 2015;**31**:3584–92.
94. Lu J, Breitwieser FP, Thielen P, *et al*. Bracken: estimating species abundance in metagenomics data. *PeerJ Comput Sci* 2017;**3**:e104.
95. Schaeffer L, Pimentel H, Bray N, *et al*. Pseudoalignment for metagenomic read assignment. *Bioinformatics* 2017;**33**:2082–8.
96. Iqbal Z, Caccamo M, Turner I, *et al*. *De novo* assembly and genotyping of variants using colored de Bruijn graphs. *Nat Genet* 2012;**44**:226–32.
97. Delcher AL, Phillippy A, Carlton J, *et al*. Fast algorithms for large-scale genome alignment and comparison. *Nucleic Acids Res* 2002;**30**:2478–83.
98. Nagarajan N, Pop M. Sequence assembly demystified. *Nat Rev Genet* 2013;**14**:157–67.
99. Ghurye JS, Cepeda-Espinoza V, Pop M. Metagenomic assembly: overview, challenges and applications. *Yale J Biol Med* 2016;**89**:353–62.
100. Vollmers J, Wiegand S, Kaster AK. Comparing and evaluating metagenome assembly tools from a microbiologist's perspective—not only size matters! *PLoS One* 2017;**12**:e0169662.
101. Li D, Liu CM, Luo R, *et al*. MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics* 2015;**31**:1674–6.
102. Bankevich A, Nurk S, Antipov D, *et al*. SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *J Comput Biol* 2012;**19**:455–77.

103. Nurk S, Meleshko D, Korobeynikov A, *et al*. metaSPAdes: a new versatile metagenomic assembler. *Genome Res* 2017;**27**: 824–34.
104. Boisvert S, Raymond F, Godzaridis E, *et al*. Ray Meta: scalable de novo metagenome assembly and profiling. *Genome Biol* 2012;**13**:R122.
105. Afiahayati Sato K, Sakakibara Y. MetaVelvet-SL: an extension of the Velvet assembler to a de novo metagenomic assembler utilizing supervised learning. *DNA Res* 2015;**22**: 69–77.
106. Namiki T, Hachiya T, Tanaka H, *et al*. MetaVelvet: an extension of Velvet assembler to de novo metagenome assembly from short sequence reads. *Nucleic Acids Res* 2012;**40**:e155.
107. Peng Y, Leung HC, Yiu SM, *et al*. IDBA-UD: a *de novo* assembler for single-cell and metagenomic sequencing data with highly uneven depth. *Bioinformatics* 2012;**28**:1420–8.
108. Treangen TJ, Koren S, Sommer DD, *et al*. MetAMOS: a modular and open source metagenomic assembly and analysis pipeline. *Genome Biol* 2013;**14**:R2.
109. Kultima JR, Coelho LP, Forslund K, *et al*. MOCAT2: a metagenomic assembly, annotation and profiling framework. *Bioinformatics* 2016;**32**:2520–3.
110. Eren AM, Esen OC, Quince C, *et al*. Anvi'o: an advanced analysis and visualization platform for 'omics data. *PeerJ* 2015;**3**: e1319.
111. Wu YW, Simmons BA, Singer SW. MaxBin 2.0: an automated binning algorithm to recover genomes from multiple metagenomic datasets. *Bioinformatics* 2016;**32**:605–7.
112. Alneberg J, Bjarnason BS, de Bruijn I, *et al*. Binning metagenomic contigs by coverage and composition. *Nat Methods* 2014;**11**:1144–6.
113. Lu YY, Chen T, Fuhrman JA, *et al*. COCACOLA: binning metagenomic contigs using sequence COmposition, read CoverAge, CO-alignment and paired-end read LinkAge. *Bioinformatics* 2017;**33**:791–8.
114. Kang DD, Froula J, Egan R, *et al*. MetaBAT, an efficient tool for accurately reconstructing single genomes from complex microbial communities. *PeerJ* 2015;**3**:e1165.
115. Laczny CC, Sternal T, Plugaru V, *et al*. VizBin - an application for reference-independent visualization and human-augmented binning of metagenomic data. *Microbiome* 2015; **3**:1.
116. Wu YW, Ye Y. A novel abundance-based algorithm for binning metagenomic sequences using l-tuples. *J Comput Biol* 2011;**18**:523–34.
117. Imelfort M, Parks D, Woodcroft BJ, *et al*. GroopM: an automated tool for the recovery of population genomes from related metagenomes. *PeerJ* 2014;**2**:e603.
118. Wang Y, Leung HC, Yiu SM, *et al*. MetaCluster 5.0: a two-round binning approach for metagenomic data for low-abundance species in a noisy sample. *Bioinformatics* 2012;**28**: i356–62.
119. Wang Y, Leung HC, Yiu SM, *et al*. MetaCluster 4.0: a novel binning algorithm for NGS reads and huge number of species. *J Comput Biol* 2012;**19**:241–9.
120. Patil KR, Roune L, McHardy AC. The PhyloPythiaS web server for taxonomic assignment of metagenome sequences. *PLoS One* 2012;**7**:e38581.
121. Gregor I, Droge J, Schirmer M, *et al*. PhyloPythiaS+: a self-training method for the rapid reconstruction of low-ranking taxonomic bins from metagenomes. *PeerJ* 2016;**4**:e1603.
122. Mikheenko A, Saveliev V, Gurevich A. MetaQUAST: evaluation of metagenome assemblies. *Bioinformatics* 2016;**32**: 1088–90.
123. Simao FA, Waterhouse RM, Ioannidis P, *et al*. BUSCO: assessing genome assembly and annotation completeness with single-copy orthologs. *Bioinformatics* 2015;**31**:3210–2.
124. Zerbino DR, Birney E. Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res* 2008;**18**: 821–9.
125. Peng Y, Leung HC, Yiu SM, *et al*. IDBA–a practical iterative de Bruijn graph de novo assembler. In: *14th Annual International Conference, RECOMB 2010, Lisbon, Portugal, 25-28 April 2010*. In Research in Computational Molecular Biology. Springer-Verlag, Berlin Heidelberg, 2010; vol. 6044, 426–40.
126. Sczyrba A, Hofmann P, Belmann P, *et al*. Critical assessment of metagenome interpretation—a benchmark of computational metagenomics software. *bioRxiv* 2017.
127. Bowe A, Onodera T, Sadakane K, *et al*. Succinct de Bruijn Graphs. In: *Algorithms in Bioinformatics*. Berlin, Heidelberg: Springer, 2012, 225–35.
128. Koren S, Phillippy AM. One chromosome, one contig: complete microbial genomes from long-read sequencing and assembly. *Curr Opin Microbiol* 2015;**23**:110–20.
129. Driscoll CB, Otten TG, Brown NM, *et al*. Towards long-read metagenomics: complete assembly of three novel genomes from bacteria dependent on a diazotrophic Cyanobacterium in a freshwater lake co-culture. *Stand Genomic Sci* 2017;**12**:9.
130. Sedlar K, Kupkova K, Provaznik I. Bioinformatics strategies for taxonomy independent binning and visualization of sequences in shotgun metagenomics. *Comput Struct Biotechnol J* 2017;**15**:48–55.
131. Albertsen M, Hugenholtz P, Skarshewski A, *et al*. Genome sequences of rare, uncultured bacteria obtained by differential coverage binning of multiple metagenomes. *Nat Biotechnol* 2013;**31**:533–8.
132. Land M, Hauser L, Jun SR, *et al*. Insights from 20 years of bacterial genome sequencing. *Funct Integr Genomics* 2015;**15**: 141–61.
133. Dick GJ, Andersson AF, Baker BJ, *et al*. Community-wide analysis of microbial genome sequence signatures. *Genome Biol* 2009;**10**:R85.
134. Vernikos G, Medini D, Riley DR, *et al*. Ten years of pan-genome analyses. *Curr Opin Microbiol* 2015;**23**:148–54.
135. Nielsen HB, Almeida M, Juncker AS, *et al*. Identification and assembly of genomes and genetic elements in complex metagenomic samples without using reference genomes. *Nat Biotechnol* 2014;**32**:822–8.
136. Federhen S. The NCBI taxonomy database. *Nucleic Acids Res* 2012;**40**:D136–43.
137. Cochrane G, Karsch-Mizrachi I, Takagi T, *et al*. The international nucleotide sequence database collaboration. *Nucleic Acids Res* 2016;**44**:D48–50.
138. Balvočiūtė M, Huson DH. SILVA, RDP, Greengenes, NCBI and OTT—how do these taxonomies compare? *BMC Genomics* 2017;**18**:114.
139. Rosselló-Móra R, Amann R. Past and future species definitions for Bacteria and Archaea. *Syst Appl Microbiol* 2015;**38**: 209–16.
140. Lan R, Reeves PR. *Escherichia coli* in disguise: molecular origins of Shigella. *Microbes Infect* 2002;**4**:1125–32.
141. Taylor JW. One Fungus = One Name: DNA and fungal nomenclature twenty years after PCR. *IMA Fungus* 2011;**2**:113–20.
142. Federhen S. Type material in the NCBI taxonomy database. *Nucleic Acids Res* 2015;**43**:D1086–98.
143. Lapage SP, Sneath P, Lessel EF, *et al*. *International Code of Nomenclature of Bacteria: Bacteriological Code, 1990 Revision*. Washington, DC: ASM Press, 2010.

144. Murray RG, Stackebrandt E. Taxonomic note: implementation of the provisional status Candidatus for incompletely described procaryotes. *Int J Syst Bacteriol* 1995;**45**:186–7.
145. Konstantinidis KT, Rosselló-Móra R. Classifying the uncultivated microbial majority: a place for metagenomic data in the Candidatus proposal. *Syst Appl Microbiol* 2015;**38**:223–30.
146. Parker CT, Tindall BJ, Garrity GM. International code of nomenclature of prokaryotes. *Int J Syst Evol Microbiol* 2015. doi: 10.1099/ijsem.0.000778.
147. Federhen S, Clark K, Barrett T, *et al*. Toward richer metadata for microbial sequences: replacing strain-level NCBI taxonomy taxids with BioProject, BioSample and Assembly records. *Stand Genomic Sci* 2014;**9**:1275–7.
148. Mende DR, Waller AS, Sunagawa S, *et al*. Assessment of metagenomic assembly using simulated next generation sequencing data. *PLoS One* 2012;**7**:e31386.
149. Paez-Espino D, Eloe-Fadrosh EA, Pavlopoulos GA, *et al*. Uncovering Earth's virome. *Nature* 2016;**536**:425–30.
150. Roux S, Hallam SJ, Woyke T, *et al*. Viral dark matter and virus-host interactions resolved from publicly available microbial genomes. *Elife* 2015;**4**:e08490.
151. Simmonds P, Adams MJ, Benko M, *et al*. Consensus statement: virus taxonomy in the age of metagenomics. *Nat Rev Microbiol* 2017;**15**:161–8.
152. Simmonds P. Methods for virus classification and the challenge of incorporating metagenomic sequence data. *J Gen Virol* 2015;**96**:1193–206.
153. Benson DA, Cavanaugh M, Clark K, *et al*. GenBank. *Nucleic Acids Res* 2017;**45**:D37–42.
154. Merchant S, Wood DE, Salzberg SL. Unexpected cross-species contamination in genome sequencing projects. *PeerJ* 2014;**2**:e675.
155. Tatusova T, Ciufo S, Federhen S, *et al*. Update on RefSeq microbial genomes resources. *Nucleic Acids Res* 2015;**43**:D599–605.
156. Brister JR, Ako-Adjei D, Bao Y, *et al*. NCBI viral genomes resource. *Nucleic Acids Res* 2015;**43**:D571–7.