Review

2Pipe: It Starts with a Question. Matching You with the Correct Pipeline for MAG Reconstruction

Jeferyd Yepes García and Laurent Falquet *

Posted Date: 15 October 2025

doi: 10.20944/preprints202506.0703.v2

Keywords: metagenomics; metagenome-assembled genome; pipeline; workflow manager

 doi:10.20944/preprints202506.0703.v2

Review

2Pipe: It Starts with a Question. Matching You with the Correct Pipeline for MAG Reconstruction

Jeferyd Yepes-García [1,2] and Laurent Falquet [1,2,*]

[1] Department of Biology, University of Fribourg, Fribourg, Canton of Fribourg, 1700, Switzerland

[2] Swiss Institute of Bioinformatics, Lausanne, Vaud, 1015, Switzerland

* Correspondence: should be addressed to L.F (laurent.falquet@unifr.ch)

Abstract

Whole Genome Sequencing (WGS) has boosted our ability to explore microbial diversity by enabling the recovery of Metagenome-Assembled Genomes (MAGs) directly from environmental DNA. As a result, the vast availability of sequencing data has prompted the development of numerous bioinformatics pipelines for MAG reconstruction, along with challenges to identify the most suitable pipeline to perform the analysis according to the user needs. This report briefly discusses the computational requirements of these pipelines, presents the variety of interfaces, workflow managers and package managers they feature, and describes the typical modular structure. Also, it provides a compacted technical overview of 41 publicly available pipelines or platforms to build MAGs starting from short and/or long sequences. Moreover, recognizing the overwhelming number of factors to consider when selecting an appropriate pipeline, we introduce an interactive decision-support web application, 2Pipe, that helps users to identify a suitable workflow based on their input data characteristics, desired outcomes, and computational constraints. The tool presents a question-driven interface to customize the recommendation, a pipeline gallery to offer a summarized description, and a pipeline comparison based on key factors used for the questionnaire. Beyond this and foreseeing the release of novel pipelines in the near future, we include a quick form and detailed instructions for developers to append their workflow in the application. Altogether, this review and the application equip the researchers with a general outlook of the growing metagenomics pipeline landscape and guide the users towards deciding the workflow that best fits their expectations and infrastructure.

Keywords: metagenomics; metagenome-assembled genome; pipeline benchmarking; workflow manager

Introduction

Metagenomics has boosted our ability to study microbial communities by diminishing the need for cultivation and enabling direct DNA sequencing from complex environments such as the human body, soil or aquatic ecosystems [1]. This has been possible thanks to the combination of high-quality and high-throughput sequencing technologies and recent advances in bioinformatics tools, increasing the scope and resolution at which the microbiota can be explored [2]. Moreover, reconstructing Metagenome-Assembled Genomes (MAGs) has enabled the genomic characterization of uncultured microorganisms, the discovery of previously unknown species, the inference of the community's metabolic and functional potential, the establishing ecological interactions, and the detection of evolutionary mechanisms [2,3].

Considering the ecological importance of the MAGs, some authors have designed specific genomic criteria to determine whether a recovered bin (draft genome) truly represents a MAG or not. For instance, the Minimum Information about MAGs (MIMAG) guidelines establish that MAGs can be classified into three quality tiers: high-quality drafts (HQ, ≥90% completeness and ≤5% contamination, presence of rRNA genes and tRNAs), medium-quality drafts (MQ, ≥50%

 doi:10.20944/preprints202506.0703.v2

completeness and ≤10% contamination), and low-quality drafts (below medium-quality thresholds) [4]. MAGs can also be divided into SMAGs (Species-assigned MAGs, MAGs for which a species can be assigned) and HMAGs (Hypothetical MAGs, MAGs that are supposedly genomes of novel species) according to the *genome heterogeneity spectrum* proposed by Setubal (2021) [5].

In a simplified manner, MAGs are obtained through bioinformatics pipelines that include quality control, assembling and binning the sequences, and the annotation of each recovered genome [6] (**Figure 1**). These pipelines are then responsible for the correct MAG assembly and have a key-role at extracting meaningful information about the structure and function of microbial communities [1]. Through their orchestrated workflow, they simplify and standardize the common tasks that are required to achieve HQ MAGs, reducing the occurrence of manual errors by improving reproducibility [7]. Nonetheless, pipeline choice may not be a trivial decision given that it should be based on the alignment between of user needs and workflow key factors such as the type of sequencing data they handle (short or long reads, or both), analytical functions (i.e., co-assembly, sequential co-assembly, taxonomic profiling, eukaryotic recovery), and computational environment (e.g., availability of local resources, High Performance Computing (HPC) infrastructure, or web-based tools). Therefore, pipeline selection can quickly become an overwhelming process and challenge researchers with a vast landscape of options, delaying the start of the analysis or even not obtaining the expected results since the incorrect workflow was chosen.

Here, we describe the general workflow followed by bioinformatics pipelines to recover MAGs directly from metagenomics data, discussing important aspects the pipelines feature such as the tools they encompass and the type of data they can handle. We also succinctly highlight major considerations regarding pipeline execution, storage needs and computational infrastructure. Likewise, we provide a compact overview of 41 publicly available pipelines, suites or platforms that enable MAG reconstruction and/or annotation starting from short and/or long sequences. Finally, considering the main practical features of each pipeline and aiming at aiding researchers in navigating the ecosystem of workflows, we also introduce 2Pipe, a decision-support web application designed to match metagenomics community users with the most suitable MAG pipeline based on their input data, technical requirements, bioinformatics experience and preferred interface.

1. Pipeline workflow, tools and benchmarks

The traditional computational workflow to build and annotate MAGs involves several steps [6]; **Figure 1** introduces the general series of steps to potentially achieve MQ or HQ MAGs, along with some common software integrated by the pipelines. In brief, it begins with quality control, where low-quality reads and contaminants are removed [8,9]; when required, some pipelines include the option to discard host organism sequences [10]. This is followed by the assembly step, where reads are extended to create contiguous sequences, also called contigs. The contigs are then grouped into bins that ideally represent individual genomes, based on sequence composition, coverage patterns, among other genomic features [11]. Optionally, the bins are subjected to a process of refinement when researchers consider it necessary [12,13]. Afterwards, these bins are evaluated for common metrics such as completeness and contamination to assess their quality, and hence determine whether they constitute MAGs or not, using the criteria previously mentioned [14]. In some cases, the workflows can encompass dereplication tools or modules that attempt to curate the MAG set by clustering them according to their genomic similarity, and thus selecting a representative MAG from each cluster [15]. To conclude with the workflow, the MAGs are then taxonomically affiliated and functionally annotated to assign biological meaning, extracting insights related to their identity and potential roles within their microbial communities [16,17]. A detailed description of the tools for each step of the workflow is provided by Yang *et al.* (2021) [6], and Wajid *et al.* (2022) [18] present an overview of the typical analysis pipeline and software using an interesting music analogy.

We present on **Table 1** the tools and third-party software for quality control, assembly, binning, refinement, taxonomic classification, and functional annotation each of the pipeline documented here encompasses. Additionally, a detailed description of the main workflow for each of them can be found in ***Additional File 1***, where important technical considerations such as the type of input (short

 doi:10.20944/preprints202506.0703.v2

reads, long sequences or both), tools employed at each step, advantages, limitations and/or special features they depict are presented.

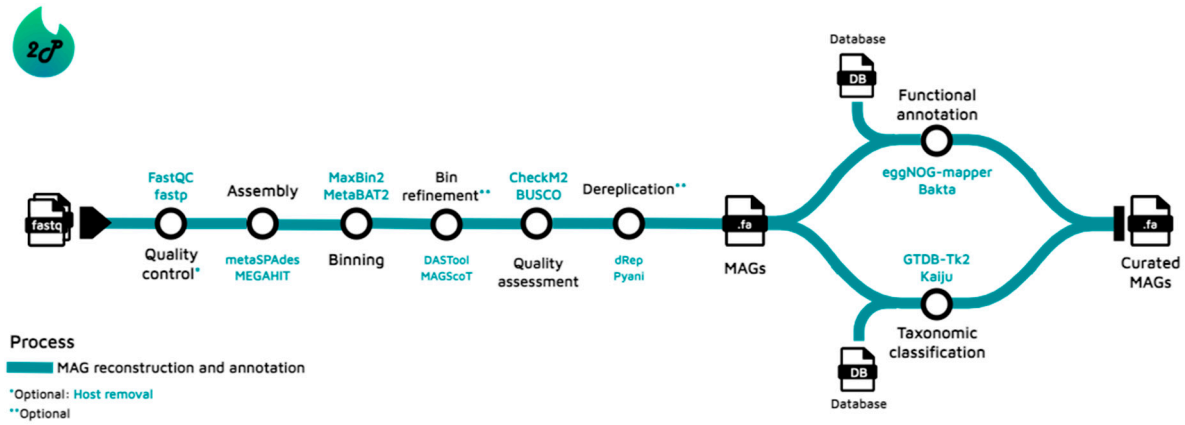

Figure 1. Usual bioinformatics workflow followed to perform MAG recovery, classification and annotation. Some common tools incorporated by the pipelines are highlighted.

As previously mentioned, the MAG reconstruction workflow is triggered with the quality control of the raw reads to ensure the accuracy and integrity of downstream analyses. Usually, the reads received from the sequencing facility contain sequencing errors, low-quality bases, adapters, and contaminant sequences (e.g., host or environment DNA) that can lead to fragmented assemblies or chimeric bins if not properly removed [6,10]. These issues are addressed by filtering and trimming, if required, the raw reads using tools like Trimmomatic [9], fastp [8], Cutadapt [19] or BBTools [20]. In the case of contamination removal, tools such as KneadData [21], Bowtie2 [22], Minimap2 [23], BWA [24] or Kraken (either v1 or v2) [25,26] are commonly used to screen and remove host-derived or non-target reads. For long-read data (Oxford Nanopore known as ONT or Pacific Biosciences known as PacBio), Filtlong [27], Nanofilt [28], and Porechop [29] are used for length filtering, quality trimming, and adapter removal. The pipeline quality control and contamination removal modules are often complemented by FastQC [30] or MultiQC [31], the standard methods to evaluate the overall quality and report it; NanoPack2 and pycoQC [32] provide detailed quality summaries for long reads. In a recent report, Gao et al. (2025) [10] compared many available tools for removing host contamination, namely, KneadData, Bowtie2, KMCP [33], BWA, KrakenUniq [34] and Kraken2, highlighting the superior performance depicted by Bowtie2 in terms of resource usage, whilst Kraken2 demonstrated the shortest execution times; for accuracy, Bowtie2, KneadData and BWA outperformed the rest of the tools.

Furthermore, the assembly step represents the core of the process since it reconstructs longer contiguous sequences from the high-quality reads. Notably, assembling metagenomics datasets faces complex challenges due to varying species abundance, uneven coverage, and the presence of closely related organisms [35]. The short-read assemblers rely mainly on two strategies: overlap-layout-consensus (OLC), which aligns overlapping reads to build contigs, and the more widely used De Bruijn graph method, which decomposes reads into k-mers and represents them as nodes and edges in a graph [35]. MEGAHIT [36], metaSPAdes [37] and IDBA-UD [38] are examples of tools that implement the De Bruijn graph approach, incorporating heuristics to address coverage variation and strain complexity. In contrast, assemblers for long-read data such as metaFlye [39], Canu [40] and hifiasm [41] are designed to apply graph-based algorithms optimized for higher error rates and uneven depth. In some cases, hybrid strategies are employed, combining long reads for structural resolution with accurate short reads for polishing or error correction, as implemented in tools like OPERA-MS [42] and hybridSPAdes [43].

Table 1. Software and tools incorporated by each pipeline or web-based platform.

	Pipeline/Platform	Quality Control Preprocessing	Assembly*	Binning	Quality Assessment	Bin Refinement	Taxonomic Annotation**	Functional Annotation**	Other
1	**Ancient DNA** [44]	FastQC [30], fastp [8], BBTools [20]	Bowtie2, MEGAHIT [36]	CONCOCT [45], MaxBin [46], MetaBAT [47]	CheckM [48]	DASTool [12]	GTDB-Tk [17]		mapDamage2 [49]
2	**Anvi'o**$^{\Delta}$ [50]	Illumina-utils [51]	metaSPAdes [37], MEGAHIT, IBDA-UD [38]	MetaBAT2 [52], CONCOCT, MaxBin2 [53], BinSanity [54]		DASTool	KrakenUniq [34], Centrifuge [55]	DIAMOND [56] (NCBI COG [57]), Pyrodigal [58], HMMER [59]	
3	**Aviary** [60]	FastQC, Filtlong [27], NanoPack2 [28], SingleM [61]	metaSPAdes, MEGAHIT, metaFlye [39], Unicycler [62]	MetaBAT2, MetaBAT, MaxBin2, VAMB [63], CONCOCT, Rosella [64]	CheckM, metaQUAST [65], CoverM [66]	DASTool	GTDB-Tk	Prodigal [67], DIAMOND(eggNOG [68])	Lorikeet [69]
4	**BugBuster** [70]	fastp, Bowtie2 [22]	MEGAHIT	METABAT2, SemiBin2 [71], COMEBin [72]	CheckM2 [73]	MetaWRAP-native module [74]	GTDB-Tk2 [75]	Prodigal, MetaCerberus [76]	Kraken2 [26], Sourmash [77], deepARG [78]
5	**BV-BRC**$^{\Delta}$ [79]	TrimGalore [80], BBTools, BLAST [81]	metaSPAdes, MEGAHIT	PATRIC metagenome binning service [82]	EvalG and EvalCon [83]		RASTtk [84]		VIGOR4 [85], Mat_Peptide [86]
6	**DATMA** [87]	Trimmomatic [9], FastQC, FLASH2 [88], BWA [24]	metaSPAdes, Velvet [89], MEGAHIT	CLAME [90]	CheckM		BLAST, Kaiju [91]	Prodigal, GeneMark [92]	Krona [93]
7	**EasyMetagenome** [94]	KneadData [21], HostPurge [94], FastQC	metaSPAdes, MEGAHIT	MetaWRAP-native [74] module	CoverM, CheckM2	MetaWRAP-native module	GTDB-Tk2	MetaProdigal [95], eggNOG-mapper [96]	dRep [97], Kraken2, Bracken [98], HUMAnN3 [21]
8	**EasyNanoMeta** [99]	fastp, Minimap2 [23], SAMtools [100], Porechop	metaFlye, OPERA-MS [42], metaSPAdes,	SemiBin2, MetaBAT2, MaxBin2,	CheckM2		GTDB-Tk2, PhyloPhlAn [104]	Prokka [105]	Kraken2, Centrifuge

		[29], BEDTools [101]	MetaPlatanus [102], NextPolish [103]	CONCOCT, VAMB					
9	**Eukfinder** [106]	Bowtie2, Trimmomatic	metaSPAdes	MyCC [107], Metaxa2 [108]			Centrifuge, PLAST [109]		
10	**EURYALE (MEDUSA)** [110,111]	FastQC, fastp, Bowtie2, MultiQC [31]	MEGAHIT				Kaiju, Kraken2	DIAMOND (NCBI nr [112])	Krona
11	**Galaxy**$^{\Delta}$ [113]	FastQC, Seqtk [114], Trimmomatic	metaSPAdes	MaxBin2			GTDB-Tk2, CAT [115]	Prokka	Kraken [25]
12	**GEN-ERA** [116]	fastp, FastQC	SPAdes [117], metaSPAdes, Canu [40], metaFlye, Pilon [118], RagTag [119]	MetaBAT2, CONCOCT	CheckM, GUNC [120], CheckM2, EukCC [121], BUSCO [122], Physeter [123], Kraken, QUAST [124]		AMAW [125], BRAKER2 [126], GTDB-Tk	Prodigal, Mantis [127], Anvi'o scripts (KEGG [128])	OrthoFinder [129]
13	**HiFi-MAG** [130]			MetaBAT2, SemiBin2	CheckM2	DASTool	GTDB-Tk2		
14	**IDseq**$^{\Delta}$ [131]	Trimmomatic, STAR [132], Bowtie2, CD-HIT [133]	SPAdes, Bowtie2				GSNAPL [134], RAPsearch2 [135]		
15	**IMG/M**$^{\Delta}$ [136]			SemiBin2	CheckM		GTDB-Tk	Prodigal, GeneMarkS-2 [137], HMMER (NCBI COG, Pfam [138], TIGRFAMs [139])	EukCC, SignalP [140], TMHMM [141]
16	**JAMS** [142]	Trimmomatic, Bowtie2	MEGAHIT, SPAdes				Kraken2	Prokka, InterProScan [143]	Samtools, BEDTools
17	**KBase**$^{\Delta}$ [144]	FastQC, Trimmomatic, Cutadapt [19]	metaSPAdes, MEGAHIT, IBDA-UD	MetaBAT2, CONCOCT, MaxBin2	CheckM	DASTool	RASTtk, GTDB-Tk	Prokka, dbCAN3 [145], DRAM [146]	OMEGGA [147], ModelSEED2 [148], Kaiju, FastANI [149], dRep, FastTree2 [150], Muscle5 [151]
18	**MAGNETO** [152]	fastp, Bowtie2, FastQscreen	MEGAHIT, Simka [154]	MetaBAT2	CheckM		GTDB-Tk,	Prodigal, Linclust [155], CD-HIT,	mOTUs [156], dRep

		[153]					eggNOG-mapper		
19	**MAGO** [157]	FastQC, fastp	metaSPAdes, MEGAHIT, IBDA-UD	MaxBin2, MetaBAT, CONCOCT, BinSanity	CheckM		GTDB-Tk	Prokka	Roary [158], ezTree [159], FastANI
20	**Mapler** [160]	FastQC	metaMDBG [161], hifiasm [41], metaFlye, OPERA-MS, Minimap2	MetaBAT2	CheckM2, metaQUAST		GTDB-Tk2, Kraken2		KAT [162]
21	**MetaGEM** [163]	fastp	MEGAHIT, BWA	MetaBAT2, CONCOCT, MaxBin2	MetaWRAP–native module		GTDB-Tk	Prokka	Roary, CarveMe [164], SMETANA [165], MEMOTE [166], GRiD [167]
22	**MetaGenePipe** [168]	Trimmomatic, TrimGalore, FastQC	MEGAHIT				DIAMOND (SwissProt [169])	Prodigal, HMMER [170] (KOfam [171])	BLAST
23	**Metagenome-Atlas** [172]	BBTools	MEGAHIT, metaSPAdes	MetaBAT2, MaxBin2, VAMB	BUSCO, CheckM, CheckM2	DASTool	GTDB-Tk	Prodigal, eggNOG, DRAM	dRep
24	**Metagenomics-Toolkit** [173]	fastp, Porechop, Filtlong, NanoPack2, KMC [174], Nonpareil [175]	metaFlye, metaSPAdes, MEGAHIT, Assembler Resource Estimator [173]	MetaBAT2, MetaCoAG [176], Metabinner [177]	CheckM	MAGScoT [13]	MMSeqs2 taxonomy [178], GTDB-Tk2	Prodigal, Prokka, RGI [179]	CarveMe, SMETANA, MEMOTE, gapseq [180], Pyani [181], SANS [182]
25	**Metaphor** [183]	FastQC, fastp, MultiQC	MEGAHIT	VAMB, MetaBAT2, CONCOCT	metaQUAST	DASTool	DIAMOND (NCBI COG)	Prodigal, Prokka	
26	**metagWGS** [184]	FastQC, Cutadapt, Sickle [185], SAMtools, BWA	metaSPAdes, MEGAHIT, hifiasm, metaFlye	MetaBAT2, CONCOCT, MaxBin2	metaQUAST	Binette [186]	GTDB-Tk2	Prodigal, eggNOG-mapper	dRep, Kaiju
27	**MetaWRAP** [74]	FastQC, TrimGalore	metaSPAdes, MEGAHIT	MetaBAT2, CONCOCT, MaxBin2	CheckM	MetaWRAP-native module	Kraken, BLAST	Prokka	Kraken, Blobology [187]

28	**MG-TK** [188]	Trimmomatic, Porechop, Kraken, Kraken2, SDM [189]	SPAdes, MEGAHIT, Flye [190], metaMDBG	MetaBAT2, SemiBin2, MetaDecoder [191]	CheckM, CheckM2		GTDB-Tk	Prodigal, DIAMOND (KEGG CAZy [192], eggNOG)	mOTUs2 [193], MetaPhlAn [194], Freebayes [195], riboFinder [196], BCFtools [100]
29	**MGnify**$^{\Delta}$ [197]	Trimmomatic, Biopython [198]	metaSPAdes				DIAMOND (UniRef90 [199])	Prodigal, FragGeneScan [200], InterProScan, eggNOG-mapper, HMMER [59]	mOTUs2, antiSMASH [201]
30	**MOSHPIT**$^{\Delta}$ [202]	Cutadapt, Bowtie2	SPAdes, MEGAHIT	MetaBAT2	QUAST, BUSCO	Sourmash	Kraken2, Kaiju	eggNOG-mapper, DIAMOND (eggNOG, CAZy)	
31	**MUFFIN** [203]	fastp, Filtlong	SPAdes, Flye, Unicycler	MetaBAT2, CONCOCT, MaxBin2	CheckM	MetaWRAP-native module	Sourmash (GTDB [204])	eggNOG-mapper	Salmon [205], Trinity [206]
32	**NanoPhase** [207]	Filtlong	metaFlye, Racon [208], medaka [209]	MetaBAT2, MaxBin2	CheckM, QUAST	MetaWRAP-native module	GTDB-Tk	Prodigal, DIAMOND (UniProtKB [210])	
33	**nf-core/mag** [211]	fastp, AdapterRemoval [212], Bowtie2, BBTools, Trimmomatic, FastQC, Porechop, Filtlong, NanoPack2	MEGAHIT, metaSPAdes, Flye, metaMDBG, hybridSPAdes [43]	MetaBAT2, CONCOCT, MaxBin2	BUSCO, CheckM, CheckM2, GUNC, QUAST	DASTool	GTDB-Tk2, CAT	Prodigal, Prokka, MetaEuk [213]	Kraken2, MultiQC, Centrifuge, PyDamage [214] geNomad [215], Tiara [216]
34	**ngs-preprocess MpGAp Bacannot** [217]	Porechop, Nanopack2, pycoQC [32], fastp	SPAdes, Flye, Canu, Unicycler, Shovill [218], HASLR [219], Raven [220], Shasta [221], wtdbg2 [222], Pilon					Prokka, antiSMASH, KOfamScan [171], KEGGDecoder [223], Bakta [16], Barrnap [224]	AMRFinderPlus [225], CARD-RGI, BEDTools, Phigaro [226], VFDB [227], PlasmidFinder [228], MLST [229], Platon [230], PHASTER [231], ARGminer [232], ResFinder [233]

35	**nIMP3** [234]	BWA, Samtools, BBTools, FastQC, Kraken2, SortMeRNA [235]	MEGAHIT						mOTUs, MultiQC, MetaPhlAn4 [21], Salmon, gffquant [236], kallisto [237]
36	**SnakeMAGs** [238]	Illumina-utils, Trimmomatic, Bowtie2	MEGAHIT	MetaBAT2	CheckM, GUNC, CoverM		GTDB-Tk2		
37	**SPIRE** [239]	NGLess [240]	MEGAHIT, BWA, Samtools	MetaBAT2	CheckM2, GUNC		GTDB-Tk2	Prodigal, eggNOG-mapper	Barrnap, RGI [179], ABRicate [241] (MEGARes [242], VFDB), Seqtk, Macrel [243], Mash [244]
38	**SqueezeMeta** [245]	PRINSEQ [246], Trimmomatic, SAMtools	MEGAHIT, SPAdes, Canu, Flye	MetaBAT2, CONCOCT, MaxBin2	CheckM, CheckM2, CompareM [247]	DASTool	GTDB-Tk2	Prodigal, MUMmer [248], HMMER, Barrnap	DIAMOND (NCBI COG, KEGG), SQMtools [249], POGENOM [250]
39	**Sunbeam** [251]	Trimmomatic, Cutadapt, Komplexity [251], BWA	MEGAHIT					Prodigal, BLAST, DIAMOND	Kraken
40	**VEBA** [252]	KneadData, fastp, BBTools, Bowtie2, NanoPack2, Minimap2	metaSPAdes, SPAdes, rnaSPAdes [253], MEGAHIT, Flye, metaFlye	MetaBAT2, CONCOCT, MaxBin2, SemiBin2	CheckM, Tiara, CheckV [254], BUSCO, CoverM	Binette	GTDB-Tk2, MetaEuk, geNomad, VirFinder [255]	Prodigal, DIAMOND (UniRef50/90, MIBiG [256], VFDB, CAZy) HMMER (Pfam, NCBIfam-AMR [225], AntiFam [257], KOfam), MicrobeAnnotator [258]	antiSMASH, Muscle5, FastTree2, FastANI, sylph [259], HUMAnN3
41	**WGSA2+/LoRA**[Δ] [260]	KneadData, fastp, Kraken2	metaSPAdes, metaFlye, MiniMap2, Samtools	MetaBAT2	CheckM, CheckM2		GTDB-Tk2	Prodigal, eggNOG-mapper, MinPath [261]	SortMeRNA, Krona, Trinity, AMRFinderPlus

* Not all the tools included here are assemblers, some of them are alignment or polishing tools that the pipeline's assembly module includes. ** If a database is necessary, it is mentioned in parenthesis. [Δ] We describe here the main workflow to recover MAGs published on these platforms or suites. However, they may offer many more services, tools or pipelines to meet any other need that the users demand.

 doi:10.20944/preprints202506.0703.v2

To this date, some authors have attempted to provide a comprehensive and unbiased benchmark of the most popular assemblers using different datasets that vary in complexity. For instance, Goussarov *et al.* (2024) [262] developed a comparison among short, long and hybrid assemblers using a complex mock metagenome with more than 200 bacterial strains, demonstrating that metaSPAdes can achieve superior performance in terms of assembly fragmentation and chimerism when using Illumina reads; while, Canu depicted the best metrics (chimerism and fragmentation) for ONT data. A similar conclusion regarding short-read assemblers was presented by Meyer *et al.* (2021) [263], where although MEGAHIT and metaSPAdes showed similar performance, metaSPAdes delivers fewer fragmented assemblies using simulated mouse gut sequences that enclosed more than 540 species. During the analysis of datasets enclosing mixed real metagenomic reads and reads from known genomes, Wang *et al.* (2019) [264] reported MEGAHIT as the most efficient assembler, while metaSPAdes outperformed MEGAHIT, IDBA-UD and Faucet [265] in terms of integrity and continuity at the species-level, and it showed the overall best performance at the strain-level.

In the case of hybrid assembly, Brown *et al.* (2021) [266] showed boosted contiguity and reduced assembly errors with either hybridSPAdes or OPERA-MS, although yielding frequent misassemblies during in-silico spike-in experiments using real and simulated reads. Nevertheless, assemblies obtained with these hybrid same tools were less complete and more fragmented than long-read only assemblies using the same dataset of more than 200 bacterial strains above-mentioned [262]. As a result, Goussarov *et al.* (2024) suggest constructing the assembly using long reads complemented with short-read polishing, when the coverage is sufficient.

Accompanying the core of the pipelines, binning tools also represent an important step to reconstruct as accurately as possible the genomes present in the microbial communities. Classical binning strategies can be divided into different categories: *i)* algorithms based on genomic composition (mainly k-mer frequencies and GC content); *ii)* approaches using read depth (coverage) profiles across multiple samples to link contigs with similar abundance patterns; and, *iii)* combined strategies that integrate both sequence composition and coverage signals [6]. Classical tools based on these strategies such as MetaBAT2 [52], MaxBin2 [53], and CONCOCT [45] have been widely incorporated into the workflows given their efficiency and robustness. Nevertheless, more recent methods leverage machine learning and semi-supervised approaches to improve resolution in more complex environments such as soil or ocean [267]. SemiBin2 [71] represents an example of these recent strategies as it uses deep learning with semi-supervised contrastive learning to incorporate both intrinsic sequence information and external reference genomes. Another example is represented by COMEBin [72], which employs graph neural networks to integrate contrastive multi-view representation learning, coverage and a clustering algorithm.

Similar to the assembly case, there have been efforts to benchmark the performance of the available binning tools. In a recent report, Han *et al.* (2025) [11] used different combinations of short, long and hybrid data to compare the outcomes from 10 binners, finding that deep-learning based tools (COMEBin, SemiBin2) were almost always among the top three high-performance binners regardless of the combination of the contig provenance. Through comparisons among less tools, Cansdale & Chong (2024) [268] showed that CONCOCT generated more high-quality bins than MetaBAT2 using a simple gut metagenome, while Meyer *et al.* (2021) [263] reported homogeneous results among CONCOCT, MetaBAT2 and MaxBin2, with the MAG completeness slightly increased by CONCOCT at the expense of genome purity. Contrastingly, Groopm2 [269] and MetaBAT2 provided the best performance metrics in recall, purity and the number of high-quality genome bins as recovering MAGs from CAMI (Critical Assessment of Metagenome Interpretation) datasets [270]. In addition, Yepes-García & Falquet (2024) [271] used environmental metagenomics samples (rice soil) to show how MetaBinner stands out for the greater number of bins recovered as compared with MetaBAT2 and SemiBin2, albeit only 10% of these were at least MQ MAGs.

Moreover, the inclusion (or enabling) of tools within the workflows to recover a non-redundant and high-quality MAG set is determinant. Several pipelines incorporate bin refinement modules or tools to improve the quality of the bin set as they reduce contamination, increase completeness, and

 doi:10.20944/preprints202506.0703.v2

may recover mis-binned contigs [12,13,97]. The tools in charge of this task take as input the bins from different binning software to provide the best possible version of each bin and potential MAG. Among the existing tools for bin refinement, MAGScoT [13] is claimed by the developers as the piece of software with the best performance, as compared to DASTool [12] and the MetaWRAP-binning module [74], in terms of MAG quantity and quality using simulated marine and human gut datasets. Nonetheless, Han *et al.* (2025) [11] showed how MetaWRAP achieved the highest rank score (custom ranking score developed for the study) followed closely by MAGScoT, although this former tool demanded 10 times less memory and carried the bin refinement in one tenth of a fraction of the time required by MetaWRAP.

Contamination estimation tools aid in the main goal of ensuring the reliability of the MAGs, with representative tools such as CheckM [48], BUSCO [122] and CheckM2 [73] that infer completeness and contamination based on single-copy marker genes from specific lineages or deep learning models. However, a benchmarking study [14] showed that CheckM may underestimate contamination, mainly if sequences from distantly related taxa are present, as it reported contamination values between 1% and 2% when the true contamination introduced by the researchers was 11%. In contrast, in the same study, the authors found that tools integrating phylogenomic signals or read classification strategies like GUNC [120], Kraken2 [26], Physeter [123] and Forty-Two [272], achieved contamination estimations closer to the true values and performed overall better at detecting inter-domain contamination. Further, within the CheckM2 paper itself, the developers demonstrated its greater accuracy to detect genome contamination conferred by unusual lineages and to predict genome completeness.

Similarly, some pipelines could include dereplication strategies after quality assessment, typically based on Average Nucleotide Identity (ANI) with the aim of curating the MAG set and selecting the *best* representative MAG in each cluster of MAGs. However, enabling the execution of these dereplication tools [97,149,181], as well as the parameter configuration should be always thought thoroughly as discussed by Evans & Denef (2020) [15], who analyzed the advantages and drawbacks of running de-replication procedures. Briefly, these authors highlighted how dereplication maintains high quality of genomic databases and enhances coverage pattern estimations; however, dereplication may lead to a loss of information on variability in the auxiliary gene content among representatives from the same species.

One of the final stages when building MAGs is represented by reporting the taxonomic affiliation of each genome. The most common tool included within the workflows (**Table 1**) is GTDB-Tk [17] in either version 1 or 2 [75], since it demonstrated that its phylogeny-based approach achieves high agreement (around 90%) with manually curated classifications in the GTDB, while GTDB-Tk v2 further optimized performance by reducing memory requirements without compromising accuracy. Beyond this, the report describing the capabilities of CAT (Contig Annotation Tool) and BAT (Bin Annotation Tool) [115], included a benchmark against GTDB-Tk that demonstrated very similar performance as BAT and GTDB-Tk provided the same final MAG annotations.

Other classifiers not particularly designed to annotate MAGs can be included within the workflows such as MetaPhlAn4 [194], Kraken [25], Kraken2 [26], Centrifuge [55], Kaiju [91], among others, through the re-formatting of the draft genomes to make them suitable as input for these tools. There have been several efforts to benchmark taxonomic classifiers in a wide variety of scenarios and using different types of data [10,273–279]; however, these studies contrasting their performance and precision have shown variable results. For instance, Kraken2 in combination with Bracken exhibited superior precision, sensitivity, F1 score, and overall sequence classification of a custom in-silico mock community within a comparison against MetaPhlAn and Kaiju [273]; similar results were described by Timilsina *et al.* (2025) [274], who reported the highest accuracy and broad sensitivity achieved by Kraken2/Bracken [98] in simulated microbial communities as compared against MetaPhlAn4 and Centrifuge. Meanwhile, Irankhah *et al.* (2024) [275] observed how MetaPhlAn4 exhibited higher precision in identifying species in a simulated dataset, outperforming Kraken2, Bracken and Centrifuge. In contrast, when attempting to classify long reads (ONT), Kraken2 and Centrifuge

 doi:10.20944/preprints202506.0703.v2

demonstrated low to very low precision for all defined mock communities (DMS) considered in the study [276]. Similarly, Centrifuge depicted the worst performer at classifying sequences belonging to a mock community built from human fecal samples, within the study that introduced the tool DeepMicrobes [277].

To complete the final stages of the MAG reconstruction, functional annotation serves to reveal metabolic potential and ecological roles of microbial communities, with a remarkably high number of options available [280]. The selection of these tools depends on the study goal, and it is usually a conscious decision made by the researchers. For more than 10 years Prokka [105] has remained as standard for rapid genome annotation, predicting coding sequences, rRNAs, and tRNAs, and assigning functions through curated databases. Nevertheless, more elaborated tools like eggNOG-mapper [96] have emerged to provide large-scale functional annotation, and the Distilled and Refined Annotation of Metabolism pipeline (DRAM) [146] offers detailed metabolic summaries. Web-based systems like RASTtk [84] (implemented within the Bacterial and Viral Bioinformatics Resource Center, BV-BRC [79]) and MGnify [197] can achieve quick and reliable annotations, whilst for specialized functional insights, tools like antiSMASH [201], KOfamKOALA [171] and dbCAN3 [145] are often incorporated into the workflows.

As suggested within the previous statements, taxonomic and functional annotation steps heavily rely on existing databases, highlighting the importance of these information resources. In the case of taxonomic classification, the Genome Taxonomy Database (GTDB) [204] provides a phylogenetically consistent framework for prokaryotic and archaeal taxonomy, whilst nucleotide and protein repositories like UniRef [199] and Swiss-Prot [169] offer curated sequences that serve reliable standards for accurate assignments. On the functional prediction side, the Kyoto Encyclopedia of Genes and Genomes (KEGG) [128] and its ortholog collection (KOfam [171]) enables the reconstruction of metabolic pathways, while Pfam [138] catalogs protein domains and families that help identify conserved protein functions. In the same sense, the database for evolutionary genealogy of genes: Non-supervised Orthologous Groups (eggNOG) [68] covers orthologous groups linked to functional categories including Cluster of Orthologs Groups (COG) [57], KEGG, and Gene Ontology (GO) terms [281]. Other specialized databases are represented by the Carbohydrate-Active enZYmes database (CAZy [192]) and the database of proteolytic enzymes, their substrates and inhibitors [282]. Please note that this is not a comprehensive review, and hence we suggest further reading of the works by Zeller & Huson (2022) [283] and Lin *et al.* (2024) [280], who explored and compared computational methods and classification systems, including databases, for protein function prediction.

Finally, considering the pipelines single tool, benchmarking them can be more difficult as they include many pieces of software that make setting a groundline for comparisons specially challenging. Notwithstanding, there are a few works where the whole pipeline execution has been benchmarked, for instance, Churcheward *et al.* (2022) [152], who tested their pipeline performance (MAGNETO) against similar workflows such as nf-core/mag, Metagenome-Atlas and MetaWRAP. These authors recovered a superior number of HQ MAGs from human gut microbiomes (Integrative Human Microbiome Project) through MetaWRAP operated in either single-assembly with single binning or co-assembly with co-binning approach (see the next section for a detailed explanation of these approaches). Meanwhile, Yepes-García & Falquet (2024) [271], starting from sequences belonging to a mock community, depicted slight differences in terms of genome completeness, contamination and number of MAGs taxonomically annotated at species level among MetaWRAP, nf-core/mag, SnakeMAGs and Metagenome-Atlas. nf-core/mag reached the highest percentages of MQ and HQ MAGs; DATMA was also included in this study, although it performed poorly as only 40% of the MAGs were assigned a proper taxonomic classification and not a single MQ or HQ MAG was recovered.

2. Practical and technical considerations for pipeline execution

As high-throughput sequencing technologies have grown in the past years, the availability of MAG-centered pipelines has been quickly expanded to handle and integrate different data types and

 doi:10.20944/preprints202506.0703.v2

computational strategies [173,184,252]. Specifically, recent pipelines have been designed or have evolved to assemble and bin short reads (normally Illumina), long reads (mainly ONT and PacBio) or a blend of both technologies to maximize high base accuracy, depth, contiguity and structural information [184,252]. Short reads synthesized through DNA nanoball sequencing (DNBSQ) [284] or long reads derived from CycloneSEQ [285] can be eventually processed by some pipelines [211,217]. Differences or similarities among these MAG-reconstruction approaches based on the type of sequence used as input have been studied by Goussarov *et al*. (2024) [262], and *Kim et al.* (2021) [286] analyzed the variations in terms of genome recovery between Illumina and MGI platforms.

Among the several steps a pipeline is composed (**Figure 1**), assembly and binning tools are the main responsible for the scaling up in the hardware demands, especially when handling datasets with several samples encompassing millions of short-read sequences [6]. Moreover, these tools can be executed in different configurations such as co-assembly and co-binning as these strategies can increase overall MAG recovery rate and quality [287]. Briefly, co-assembly refers to the possibility of performing the metagenome assembly after merging user-specified samples to enhance the coverage, capturing a higher fraction of the diversity [287], while co-binning establishes the possibility of binning contigs using coverage information across multiple samples simultaneously after single or co-assembly [11]. Co-binning is advantageous at exploring coverage across samples and improving separation of closely related genomes [63]. Despite the desirable benefits co-assembly can bring to the analysis, it is computationally intensive and increases the probability of generating fragmented assemblies [152], although sequential co-assembly has emerged recently as an efficient alternative that enhances both time and memory requirements by the assembler [288]. Similarly, co-binning can be sensitive to uneven sequencing depth, requires high-quality coverage profiles and can be affected by low diversity among samples [152]. Vosloo *et al.* (2021) [287] and Han *et al.* (2025) [11] have demonstrated how superior performance can be achieved by applying co-assembly and/or co-binning.

On the other hand, the workflow execution varies in terms of computational demands, where small-scale datasets can be processed on high-end workstations, whilst large or complex metagenomes often require access to HPC clusters or cloud-based environments (Azure, Amazon Web Services or AWS, Google Cloud, Terra, among others). Beyond the sample-specific computational requirements, most metagenomics pipelines rely on external reference databases to perform taxonomic classification, functional annotation, and quality assessment of MAGs. Commonly used databases, namely, RefSeq [289], GTDB, UniProt [210], KEGG and eggNOG are large and require substantial local storage that ranges from tens to hundreds of gigabytes. For instance, the latest GTDB release (R226) exceeds 140 GB, while comprehensive functional annotation pipelines like DRAM can demand up to 500 GB to exploit its full potential. Being so, MAG building is a demanding process that needs adequate disk space, CPU capacity and memory availability.

For researchers without access to HPC resources, web-based platforms such as KBase [290], MGnify [197], Galaxy [113], BV-BRC [79], among others, can assist them by carrying out analysis execution in their servers. In addition, these platforms aid users without a strong experience in command line interface (CLI) interaction since they provide user-friendly interfaces where users can upload raw reads and run predefined workflows. Being so, these platforms eliminate the need for command-line interaction and offer built-in visualization applications and databases for downstream interpretation; a complete landscape of web-based applications is compiled by Achudhan *et al.* (2024) [291] and Chivian *et al.* (2023) [144].

Furthermore, given the MAG pipeline evolution in complexity, involving multiple tools, dependencies and steps, the use of workflow managers has become the standard to ensure reproducibility, scalability, and portability [292]. Specifically, workflow managers ease pipeline step definition in a modular and automated architecture to orchestrate entire analyses, tracking software versions, managing intermediate files, restarting the process if interrupted, handling multiple samples as input and enabling parallel processing in a reproducible manner. Some representatives of these helpful orchestrators are Snakemake [293], Nextflow [294] and Workflow Definition Language

 doi:10.20944/preprints202506.0703.v2

(WDL) [295] whose design, implementation, benefits and scope have been reviewed in some reports [292–294,296]; also, important guidelines for pipeline design based on workflow managers have been published by Roach *et al.* (2022) [297], Reiter *et al.* (2020) [298] and Ahmed *et al.* (2021) [7]. Advantageously, containerization platforms such as Docker, Singularity and Seqera Containers, or package managers like Conda or the Python Package Index (PyPI) complement workflow orchestrators by offering a flexible and reproducible solution for software and dependency management [299]. As a result, this combination allows users to run the analysis, without system conflicts, specific versions of the software and libraries.

In contrast, beyond the MAG assembly and annotation, some pipelines feature interesting options that complement the analysis and provide a wider understanding about the microbial community. The range of these special options is wide, and therefore they must be carefully selected. In this sense, read-based taxonomic profiling [1] is one of the most common offerings by the pipelines, as this process does not rely on the main workflow and can be executed in parallel. Furthermore, some pipelines can incorporate tools or modules to recover viral or eukaryotic MAGs [252], and it is even possible to find pipelines mostly focused on this type of MAGs [106]. Another popular extra option is represented by the possibility of establishing genome-scale metabolic models (GEMs) among the built MAGs [163,173]. However, in many cases some workflows can be considered as *unique* since they include options that no other pipeline encompasses. Examples of these *rare* features are the possibility to assemble plasmids [173], genotype recovery [60], RNA-seq transcriptome analysis [203], controlled resource allocation [173] and an alternative assembly and binning order, where the reads are first grouped (binning) and then assembled in batches [87].

On **Table 2**, we present a summarized overview of the technical features and methodological factors each workflow presents, and hence these same pipeline aspects are also the basis for the questionnaire presented on 2Pipe. Methodological factors include the ability to assemble short reads, long sequences or both in a hybrid approach, the possibility to request a co-assembly and/or co-binning natively, whether the user can input multiple samples or not, if the pipeline includes a bin refinement tool, and special functionalities they may incorporate. In the same sense, technical features are described by which kind of resources the user is planning to use for the pipeline execution, the interface they feel more comfortable working with, the workflow manager they expect to orchestrate the data flow, and the software/package technology management available within each workflow. We assigned one of the following (non-mutually exclusive) labels in order to classify them: short-read centered or long-read focused (if their main input is short or long reads), dual (if they can handle both long and short reads, but they do not perform hybrid assembly), hybrid (pipelines able to assemble short and long reads together), web-based (pipelines offered by online platforms or suites) or special (pipelines designed for a specific purpose).

Table 2. Technical and operational features for each pipeline or web-based platform.

	Pipeline/Platform	Category	Short reads	Long reads*	Hybrid Assembly	Multiple samples	Co-assembly and/or Co-binning**	Bin refinement	Infrastructure ***	Interface ∇	Workflow manager	Software execution	Special features	Last update °	Number of citations °	License®
1	**Ancient DNA** [44]	Special	Yes	No	No	No	No	Yes	Local, HPC	CLI		Local	Ancient DNA identification	2024	0	Not specified
2	**Anvi'o** [50]	Short-read centered	Yes	No	No	Yes	Yes	Yes	Local, HPC	CLI/GUI		Conda	Visualization module	2025	678	GNU GPL v3
3	**Aviary** [60]	Hybrid	Yes	Yes	Yes	Yes	No	Yes	Local, HPC, CC	CLI	Snakemake	Conda	Genotype recovery	2025	NA	GNU GPL v3
4	**BugBuster** [70]	Short-read centered	Yes	No	No	Yes	No	Yes	Local, HPC, CC	CLI	Nextflow	Docker	Taxonomic profiling, antimicrobial resistance gene prediction	2025	0	Not specified
5	**BV-BRC** [79]	Web-based	Yes	No	No	Yes	No	No	External	GUI		External	Taxonomic profiling, Viral MAGs	2024	783	MIT License
6	**DATMA** [87]	Short-read centered	Yes	No	No	No	No	No	Local, HPC	CLI	COMP Superscalar [300]	Local	Reads first grouped (binning) and assembled in batches	2020	4	GNU GPL v3
7	**EasyMetagenome** [94]	Short-read centered	Yes	No	No	Yes	Yes	Yes	Local, HPC	CLI		Conda	Taxonomic profiling	2024	14	GNU GPL v3

8	**EasyNanoMeta** [99]	Long-read focused	No	Yes (ONT)	Yes	Yes	No	No	Local, HPC	CLI		Conda, Singularity	Taxonomic profiling	2024	0	GNU GPL v3
9	**Eukfinder** [106]	Special	Yes	Yes	No	No	No	No	Local, HPC	CLI		Conda	Eukaryotic MAGs	2025	1	MIT License
10	**EURYALE (MEDUSA)** [110,111]	Short-read centered	Yes	No	No	Yes	No	No	Local, HPC, CC	CLI	Nextflow	Conda, Singularity, Docker		2024	7	MIT License
11	**Galaxy** [113]	Web-based	Yes	Yes	Yes	No	No	Yes	External	GUI		External	Taxonomic profiling	2024	1168	Academic Free License v3
12	**GEN-ERA** [116]	Dual	Yes	Yes (ONT)	No	Yes	No	No	Local, HPC, CC	CLI	Nextflow	Singularity	Metabolic modeling	2024	7	GNU GPL v3
13	**HiFi-MAG** [130]	Long-read focused	No	Yes (PacBio)	No	Yes	No	Yes	Local, HPC, CC	CLI	Snakemake	Conda		2025	8	BSD-3-Clause-Clear
14	**IDseq** [131]	Web-based	Yes	Yes (ONT)	No	No	No	No	External	GUI		External	Viral MAGs	2025	347	MIT License
15	**IMG/M** [136]	Web-based	NA	NA	NA	No	No	No	External	GUI		External	Eukaryotic MAGs	2025	268	IMG Expert Review Submission Agreement
16	**JAMS** [142]	Short-read centered	Yes	No	No	No	No	No	Local, HPC	CLI		Conda	Direct sample comparison	2025	7	GNU GPL v3

17	**KBase** [144]	Web-based	Yes	Yes	Yes	Yes	Yes	Yes	External	GUI		External	Taxonomic profiling, metabolic modeling	2024	63	MIT License
18	**MAGNETO** [152]	Short-read centered	Yes	No	No	Yes	Yes	No	Local, HPC, CC	CLI	Snakemake	Conda	Taxonomic profiling	2025	13	GNU GPL v3
19	**MAGO** [157]	Short-read centered	Yes	No	No	No	No	Yes	Local, HPC	CLI		Singularity, Docker	Phylogenetic tree generation, pangenome analysis	2020	21	Creative Commons BY 4.0
20	**Mapler** [160]	Long-read focused	No	Yes (PacBio)	No	Yes	No	No	Local, HPC, CC	CLI	Snakemake	Conda	Visualization module	2025	0	GNU AGPL v3
21	**MetaGEM** [163]	Short-read centered	Yes	No	No	Yes	No	Yes	Local, HPC, CC	CLI	Snakemake	Conda	Eukaryotic MAGs, Metabolic modeling	2023	99	MIT License
22	**MetaGenePipe** [168]	Short-read centered	Yes	No	No	Yes	Yes	No	Local, HPC, CC	CLI	Workflow Definition Language (WDL) [295]	Singularity		2023	1	Apache License 2.0
23	**Metagenome-Atlas** [172]	Short-read centered	Yes	No	No	Yes	No	Yes	Local, HPC, CC	CLI	Snakemake	Conda		2024	159	BSD-3-Clause-Clear
24	**Metagenomics-Toolkit** [173]	Dual	Yes	Yes (ONT)	No	Yes	No	Yes	Local, HPC, CC	CLI	Nextflow	Docker	Plasmid assembly, metabolic modeling, controlled resource allocation	2025	0	GNU AGPL v3

25	**Metaphor** [183]	Short-read centered	Yes	No	No	Yes	Yes	Yes	Local, HPC, CC	CLI	Snakemake	Conda	Visualization module	2024	13	MIT License
26	**metagWGS** [184]	Dual	Yes	Yes (PacBio)	No	Yes	Yes	Yes	Local, HPC, CC	CLI	Nextflow	Singularity	Taxonomic profiling	2025	2	GNU GPL v3
27	**MetaWRAP** [74]	Short-read centered	Yes	No	No	Yes	Yes	Yes	Local, HPC	CLI		Conda, Docker	Taxonomic profiling	2020	1917	MIT License
28	**MG-TK** [188]	Dual	Yes	No	No	Yes	Yes	No	Local, HPC	CLI		Conda	Taxonomic profiling, strain delineation	2025	99	GNU GPL v2
29	**MGnify** [197]	Web-based	Yes	Yes	Yes	Yes	Yes	No	External	GUI		External	Taxonomic profiling	2025	286	Apache License 2.0
30	**MOSHPIT** [202]	Short-read centered	Yes	No	No	Yes	No	Yes	Local, HPC	CLI		Conda	Taxonomic profiling	2025	1	BSD-3-Clause-Clear
31	**MUFFIN** [203]	Hybrid pipelines	No	Yes (ONT)	Yes	Yes	No	Yes	Local, HPC, CC	CLI	Nextflow	Conda, Docker, Singularity	Metatranscriptome support	2022	34	GNU GPL v3
32	**NanoPhase** [207]	Long-read focused	No	Yes (ONT)	Yes	No	No	Yes	Local, HPC	CLI		Conda		2023	73	MIT License
33	**nf-core/mag** [211]	Hybrid	Yes	No	Yes	Yes	Yes	Yes	Local, HPC, CC	CLI	Nextflow	Conda, Docker, Singularity, Others	Taxonomic profiling, ancient DNA identification	2025	57	MIT License

34	**ngs-preprocess MpGAp Bacannot** [217]	Hybrid	Yes	Yes	Yes	Yes	No	No	Local, HPC, CC	CLI	Nextflow	Conda, Docker, Singularity	Antimicrobial resistance gene prediction, virulence factor annotation, plasmid assembly	2025	2	GNU GPL v3
35	**nIMP3** [234]	Short-read centered	Yes	No	No	Yes	No	No	Local, HPC, CC	CLI	Nextflow	Docker, Singularity	Metatranscriptome support, taxonomic profiling	2024	150	MIT License
36	**SnakeMAGs** [238]	Short-read centered	Yes	No	No	Yes	No	No	Local, HPC, CC	CLI	Snakemake	Conda		2024	6	CeCILL Free Software License Agreement v2.1
37	**SPIRE** [239]	Short-read centered	Yes	No	No	Yes	No	No	Local, HPC, CC	CLI	Nextflow		Antimicrobial resistance gene prediction, virulence factor annotation	2025	41	MIT License
38	**SqueezeMeta** [245]	Hybrid	Yes	Yes	Yes	Yes	Yes	Yes	Local, HPC	CLI		Conda	Taxonomic profiling, metatranscriptome support, visualization module	2025	400	GNU GPL v3
39	**Sunbeam** [251]	Short-read centered	Yes	No	No	Yes	No	No	Local, HPC	CLI	Snakemake	Conda, Docker	Taxonomic profiling	2025	184	GNU GPL v3
40	**VEBA** [252]	Dual	Yes	Yes (ONT	No	Yes	Yes and pseudo-	Yes	Local, HPC	CLI	GenoPype [301]	Conda, Docker	Eukaryotic or Viral MAGs, antimicrobial resistance gene	2025	23	GNU AGPL v3

				or PacBio)			coassembly						prediction, virulence factor annotation			
41	**WGSA2+/LoRA** [260]	Web-based	Yes	Yes (ONT or PacBio)	No	Yes	No	No	External, CC	GUI	AWS environment	External	Visualization module, metatranscriptome support, antimicrobial resistance gene prediction	2025	138	CC0 1.0 Universal

***Long reads:**

- ONT: Oxford Nanopore Technology.
- PacBio: Pacific Biosciences.

****Co-assembly and/or Co-binning:** it highlights if the pipeline counts with options to control co-assembly and/or co-binning.

*****Infrastructure:** It refers to the computational infrastructure where the pipeline can be executed natively.

- HPC: High Performance Cluster.
- CC: Cloud Computing.
- External: Pipelines controlled by the platform or suite and use external resources.

▽**Interface:**

- CLI: Command Line Interface.
- GUI: Graphical User Interface.

◦**Last update and Number of citations:** at the moment of writing this report.

◎**License:** These licenses cover the pipeline code and platforms; the third-party software and tools they encompass may be covered by a different license.

 doi:10.20944/preprints202506.0703.v2

3. 2Pipe: It starts with a question

Considering the pipeline landscape identified in this review, we have developed a decision-support application that concatenates most of the features described for each workflow. 2Pipe is an interactive web application designed to help researchers to identify the most suitable metagenomics pipeline for reconstructing and annotating MAGs. 2Pipe can be used by users with different expertise levels and computational access, simplifying the often-complex selection process by mapping user needs to a curated database of available pipelines.

At the core of 2Pipe is a dynamic, question-driven interface that guides users step by step through a personalized questionnaire. This adaptive form collects information related to the methodological factors and technical features detailed on ***Table 1***. Therefore, every response is used to assign a score to each pipeline based on the presence or absence of specific features that align with the user's input. The recommendation system will then suggest the pipeline with the highest score, as well as the second "best hit" for the user to check in case that the first option does not fulfill their requirements; these suggestions can be as well the starting point for the user to dig into the other sections of 2Pipe. It is worthy to mention that the scoring is weighted, and some features have prevalence as they are definitive for the pipeline suggestion. Specifically, all matching features presented in the questions add one point to the final score, excepting type of reads to analyze (2 points), the need for a GUI (3 points) and the requirement for external computational resources (3 points). These features are prioritized as if selected by the user, the recommendation must reflect them as they cannot simply be bypassed with any other pipeline. The system also includes a protection for cases when the users do not provide at least three answers, asking them to restart the questionnaire. Likewise, in case of a tie among more than two pipelines, the recommendation system will show all of them with the respective matching features.

Aside from the accession to the questionnaire and the response-based recommendation at the end of this, 2Pipe as well encompasses a pipeline gallery, where a visual catalog is displayed offering individual summaries of each pipeline, describing their main characteristics, supporting technology and a direct access to the source code or publication documenting the pipeline. Additionally, 2Pipe makes available an interactive view of ***Table 2*** that includes the possibility of filtering by each feature or by a combination of them, allowing users to directly tailor the search for the pipeline that best suits their needs; the displayed categories are the same key attributes the question-based suggestion system relies on. 2Pipe also incorporates the features presented in ***Table 1***, assisting the user when comparing the pipelines beyond technical aspects. Also, these tools and external software are organized in a gallery that allows the user to match pipelines that use them, which is useful if the user is looking for a specific software combination that a given pipeline can offer.

On the other hand, given the importance pipeline and tool benchmarking represents, 2Pipe provides an exclusive page where the reports cited in this work comparing performance and/or technical features are introduced. This page is divided into sections according to the tools benchmarked in the papers namely assemblers, binners, bin-refinement tools, contamination-estimation software, complete pipelines, workflow managers and taxonomic classifiers. Moreover, we include sections for reviews, tutorials and protocols for manual MAG reconstruction and key-papers that set interesting discussions around MAG recovery.

The source code for 2Pipe is available at the repository https://github.com/jeffe107/2pipe, and foreseeing the possibility of new pipelines being released in the near future, we provide a quick form for developers to include their workflow into 2Pipe's recommendation system, pipeline gallery and table comparison. Complementary, at the GitHub repository, developers can find a simple template and detailed instructions for the inclusion of their pipeline through a pull request.

Conclusion

The rapid evolution of sequencing technologies has boosted the availability of metagenomics datasets that demand bioinformatics tools adjusted to the user requirements to achieve cutting-edge

 doi:10.20944/preprints202506.0703.v2

analysis, including MAG reconstruction. As a result, in the past 10 years a rise in the number of MAG reconstruction pipelines available has been observed, and the selection of the proper pipeline for the analysis has become an essential step during the execution of metagenomics projects. This review offers a compacted description of 41 publicly available pipelines or platforms, with special focus on their capabilities and distinctive features to serve as a valuable resource for researchers navigating this overwhelming landscape. Expanding the scope of a classical review, we streamlined the selection process by introducing 2Pipe, an interactive decision-support web application that aligns the user needs with the most convenient workflow for their analysis and allows a general overview of the pipeline landscape with its gallery and pipeline-comparison sections. Finally, this review and its accompanying application provide a unified framework that simplifies the decision-making process, releasing part of the burden and uncertainty when setting a metagenomics data analysis project.

Consent for publication: There is no conflict to consent for publication.

Availability of data and materials: 2Pipe is hosted under the domain https://2pipe.app/. The source code is available at https://github.com/jeffe107/2pipe, along with a template to include new pipelines. The quick form to add a new pipeline can be found at https://form.jotform.com/jeffe10789/2pipe-form. For version tracking, 2Pipe v.2.0 release has been deposited at Zenodo, and it can be followed with the identifier https://doi.org/10.5281/zenodo.17334924.

Competing interests: The authors declare no competing interests.

Contributions: JYG performed manuscript writing, data integration, visualization and deposition. LF supervised and oversaw application development and data analysis. All authors participating during the review framework conceptualizing, conceiving the overall work and manuscript preparation.

Acknowledgments: JYG specially thanks the Federal Commission for Scholarships for Foreign Students (FCS) for their support through the Swiss Government Excellence Scholarship.

Additional files: ***Additional File 1*** (.pdf). File containing the detailed summary description for each pipeline considered in this review. It can be found at: https://doi.org/10.5281/zenodo.17335110.

References

1. Navgire, G. S. *et al.* Analysis and Interpretation of metagenomics data: an approach. *Biol. Proced. Online* **24**, 1–22 (2022).
2. Kim, N. *et al.* Genome-resolved metagenomics: a game changer for microbiome medicine. *Exp. Mol. Med.* **56**, 1501–1512 (2024).
3. Lemos, L. N., Mendes, L. W., Baldrian, P. & Pylro, V. S. Genome-Resolved Metagenomics Is Essential for Unlocking the Microbial Black Box of the Soil. *Trends Microbiol.* **29**, 279–282 (2021).
4. Bowers, R. M. *et al.* Minimum information about a single amplified genome (MISAG) and a metagenome-assembled genome (MIMAG) of bacteria and archaea. *Nat. Biotechnol.* **35**, 725–731 (2017).
5. Setubal, J. C. Metagenome-assembled genomes: concepts, analogies, and challenges. *Biophys. Rev.* **13**, 905–909 (2021).
6. Yang, C. *et al.* A review of computational tools for generating metagenome-assembled genomes from metagenomic sequencing data. *Computational and Structural Biotechnology Journal* vol. 19 6301–6314 (2021).
7. Ahmed, A. E. *et al.* Design considerations for workflow management systems use in production genomics research and the clinic. *Sci. Rep.* **11**, 1–18 (2021).
8. Chen, S., Zhou, Y., Chen, Y. & Gu, J. fastp: an ultra-fast all-in-one FASTQ preprocessor. *Bioinformatics* **34**, i884–i890 (2018).
9. Bolger, A. M., Lohse, M. & Usadel, B. Trimmomatic: a flexible trimmer for Illumina sequence data. *Bioinformatics* **30**, 2114–2120 (2014).

 doi:10.20944/preprints202506.0703.v2

10. Gao, Y. *et al.* Benchmarking short-read metagenomics tools for removing host contamination. *GigaScience* **14**, giaf004 (2025).
11. Han, H., Wang, Z. & Zhu, S. Benchmarking metagenomic binning tools on real datasets across sequencing platforms and binning modes. *Nat. Commun.* **16**, 2865 (2025).
12. Sieber, C. M. K. *et al.* Recovery of genomes from metagenomes via a dereplication, aggregation and scoring strategy. *Nat. Microbiol.* **3**, 836–843 (2018).
13. Christoph, M., Rühlemann, R., Wacker, E. M., Ellinghaus, D. & Franke, A. MAGScoT: a fast, lightweight and accurate bin-refinement tool. *Bioinformatics* **38**, 5430–5433 (2022).
14. Cornet, L. & Baurain, D. Contamination detection in genomic data: more is not enough. *Genome Biol.* **23**, 1–15 (2022).
15. Evans, J. T. & Denef, V. J. To Dereplicate or Not To Dereplicate? *mSphere* **5**, (2020).
16. Schwengers, O. *et al.* Bakta: Rapid and standardized annotation of bacterial genomes via alignment-free sequence identification. *Microb. Genomics* **7**, 000685 (2021).
17. Chaumeil, P. A., Mussig, A. J., Hugenholtz, P. & Parks, D. H. GTDB-Tk: a toolkit to classify genomes with the Genome Taxonomy Database. *Bioinformatics* **36**, 1925–1927 (2020).
18. Wajid, B. *et al.* Music of metagenomics—a review of its applications, analysis pipeline, and associated tools. *Funct. Integr. Genomics* **22**, 3–26 (2022).
19. Martin, M. Cutadapt removes adapter sequences from high-throughput sequencing reads. *EMBnet.journal* **17**, 10–12 (2011).
20. Bushnell, B. BBMap: A Fast, Accurate, Splice-Aware Aligner. LBL Publications, (2014).
21. Beghini, F. *et al.* Integrating taxonomic, functional, and strain-level profiling of diverse microbial communities with bioBakery 3. *eLife* **10**, e65088 (2021).
22. Langmead, B. & Salzberg, S. L. Fast gapped-read alignment with Bowtie 2. *Nat. Methods* **9**, 357–359 (2012).
23. Li, H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* **34**, 3094–3100 (2018).
24. Li, H. Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. Preprint at https://doi.org/10.48550/arXiv.1303.3997 (2013).
25. Wood, D. E. & Salzberg, S. L. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biol.* **15**, R46 (2014).
26. Wood, D. E., Lu, J. & Langmead, B. Improved metagenomic analysis with Kraken 2. *Genome Biol.* **20**, 1–13 (2019).
27. Haveman, N. J. *et al.* Evaluating the lettuce metatranscriptome with MinION sequencing for future spaceflight food production applications. *Npj Microgravity* **7**, 22 (2021).
28. De Coster, W. & Rademakers, R. NanoPack2: population-scale evaluation of long-read sequencing data. *Bioinformatics* **39**, btad311 (2023).
29. Wick, R. R., Judd, L. M., Gorrie, C. L. & Holt, K. E. Completing bacterial genome assemblies with multiplex MinION sequencing. *Microb. Genomics* **3**, e000132 (2017).
30. Simon, A. FastQC A Quality Control tool for High Throughput Sequence Data. FastQC [Online]. Available online at: http://www.bioinformatics.babraham.ac.uk/projects/fastqc/ (2010).
31. Ewels, P., Magnusson, M., Lundin, S. & Käller, M. MultiQC: summarize analysis results for multiple tools and samples in a single report. *Bioinformatics* **32**, 3047–3048 (2016).
32. Leger, A. & Leonardi, T. pycoQC, interactive quality control for Oxford Nanopore Sequencing. *J. Open Source Softw.* **4**, 1236 (2019).
33. Shen, W. *et al.* KMCP: accurate metagenomic profiling of both prokaryotic and viral populations by pseudo-mapping. *Bioinformatics* **39**, btac845 (2023).
34. Breitwieser, F. P., Baker, D. N. & Salzberg, S. L. KrakenUniq: confident and fast metagenomics classification using unique k-mer counts. *Genome Biol.* **19**, 198 (2018).
35. Ayling, M., Clark, M. D. & Leggett, R. M. New approaches for metagenome assembly with short reads. *Brief. Bioinform.* **21**, 584–594 (2020).
36. Li, D. *et al.* MEGAHIT v1.0: A fast and scalable metagenome assembler driven by advanced methodologies and community practices. *Methods* **102**, 3–11 (2016).

 doi:10.20944/preprints202506.0703.v2

37. Nurk, S., Meleshko, D., Korobeynikov, A. & Pevzner, P. A. MetaSPAdes: A new versatile metagenomic assembler. *Genome Res.* **27**, 824–834 (2017).
38. Peng, Y., Leung, H. C. M., Yiu, S. M. & Chin, F. Y. L. IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth. *Bioinformatics* **28**, 1420–1428 (2012).
39. Kolmogorov, M. *et al.* metaFlye: scalable long-read metagenome assembly using repeat graphs. *Nat. Methods* **17**, 1103–1110 (2020).
40. Koren, S. *et al.* Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation. *Genome Res.* **27**, 722–736 (2017).
41. Cheng, H., Concepcion, G. T., Feng, X., Zhang, H. & Li, H. Haplotype-resolved de novo assembly using phased assembly graphs with hifiasm. *Nat. Methods* **18**, 170–175 (2021).
42. Bertrand, D. *et al.* Hybrid metagenomic assembly enables high-resolution analysis of resistance determinants and mobile elements in human microbiomes. *Nat. Biotechnol.* **37**, 937–944 (2019).
43. Antipov, D., Korobeynikov, A., McLean, J. S. & Pevzner, P. A. hybridSPAdes: an algorithm for hybrid assembly of short and long reads. *Bioinformatics* **32**, 1009–1015 (2016).
44. Standeven, F. J., Dahlquist-Axe, G., Speller, C. F., Meehan, C. J. & Tedder, A. An efficient pipeline for creating metagenomic-assembled genomes from ancient oral microbiomes. Preprint at https://doi.org/10.1101/2024.09.18.613623 (2024).
45. Alneberg, J. *et al.* Binning metagenomic contigs by coverage and composition. *Nat. Methods* **11**, 1144–1146 (2014).
46. Wu, Y.-W., Tang, Y.-H., Tringe, S. G., Simmons, B. A. & Singer, S. W. MaxBin: an automated binning method to recover individual genomes from metagenomes using an expectation-maximization algorithm. *Microbiome* **2**, 26 (2014).
47. Kang, D. D., Froula, J., Egan, R. & Wang, Z. MetaBAT, an efficient tool for accurately reconstructing single genomes from complex microbial communities. *PeerJ* **3**, e1165 (2015).
48. Parks, D. H., Imelfort, M., Skennerton, C. T., Hugenholtz, P. & Tyson, G. W. CheckM: assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. *Genome Res.* **25**, 1043–1055 (2015).
49. Jónsson, H., Ginolhac, A., Schubert, M., Johnson, P. L. F. & Orlando, L. mapDamage2.0: fast approximate Bayesian estimates of ancient DNA damage parameters. *Bioinformatics* **29**, 1682–1684 (2013).
50. Eren, A. M. *et al.* Community-led, integrated, reproducible multi-omics with anvi'o. *Nat. Microbiol.* **6**, 3–6 (2020).
51. Eren, A. M., Vineis, J. H., Morrison, H. G. & Sogin, M. L. A Filtering Method to Generate High Quality Short Reads Using Illumina Paired-End Technology. *PLOS ONE* **8**, e66643 (2013).
52. Kang, D. D. *et al.* MetaBAT 2: An adaptive binning algorithm for robust and efficient genome reconstruction from metagenome assemblies. *PeerJ* **2019**, (2019).
53. Wu, Y. W., Simmons, B. A. & Singer, S. W. MaxBin 2.0: an automated binning algorithm to recover genomes from multiple metagenomic datasets. *Bioinformatics* **32**, 605–607 (2016).
54. Graham, E. D., Heidelberg, J. F. & Tully, B. J. BinSanity: unsupervised clustering of environmental microbial assemblies using coverage and affinity propagation. *PeerJ* **5**, e3035 (2017).
55. Kim, D., Song, L., Breitwieser, F. P. & Salzberg, S. L. Centrifuge: rapid and sensitive classification of metagenomic sequences. *Genome Res.* **26**, 1721-1729 (2016).
56. Buchfink, B., Reuter, K. & Drost, H. G. Sensitive protein alignments at tree-of-life scale using DIAMOND. *Nat. Methods* **18**, 366–368 (2021).
57. Galperin, M. Y. *et al.* COG database update 2024. *Nucleic Acids Res.* **53**, D356–D363 (2025).
58. Larralde, M. Pyrodigal: Python bindings and interface to Prodigal, an efficient method for gene prediction in prokaryotes. *J. Open Source Softw.* **7**, 4296 (2022).
59. Finn, R. D., Clements, J. & Eddy, S. R. HMMER web server: interactive sequence similarity searching. *Nucleic Acids Res.* **39**, W29–W37 (2011).
60. Newell, R. J. P., Aroney, S. T. N., Zaugg, J., Sternes, P., Tyson, G. W., & Woodcroft, B. J. Aviary: Hybrid assembly and genome recovery from metagenomes with Aviary (v0.12.0). Zenodo. https://doi.org/10.5281/zenodo.15208119 (2025).

 doi:10.20944/preprints202506.0703.v2

61. Woodcroft, B. J. *et al.* Comprehensive taxonomic identification of microbial species in metagenomic data using SingleM and Sandpiper. *Nat. Biotechnol.* 1–6 (2025).
62. Wick, R. R., Judd, L. M., Gorrie, C. L. & Holt, K. E. Unicycler: Resolving bacterial genome assemblies from short and long sequencing reads. *PLOS Comput. Biol.* **13**, e1005595 (2017).
63. Nissen, J. N. *et al.* Improved metagenome binning and assembly using deep variational autoencoders. *Nat. Biotechnol.* **39**, 555–560 (2021).
64. Newell, R. J. P., Tyson, G. W., & Woodcroft, B. J. . Rosella: Metagenomic binning using UMAP and HDBSCAN (v0.5.3). Zenodo. https://doi.org/10.5281/zenodo.10460259 (2024).
65. Mikheenko, A., Saveliev, V. & Gurevich, A. MetaQUAST: evaluation of metagenome assemblies. *Bioinformatics* **32**, 1088–1090 (2016).
66. Aroney, S. T. N. *et al.* CoverM: read alignment statistics for metagenomics. *Bioinformatics* **41**, btaf147 (2025).
67. Hyatt, D. *et al.* Prodigal: Prokaryotic gene recognition and translation initiation site identification. *BMC Bioinformatics* **11**, 1–11 (2010).
68. Huerta-Cepas, J. *et al.* eggNOG 5.0: a hierarchical, functionally and phylogenetically annotated orthology resource based on 5090 organisms and 2502 viruses. *Nucleic Acids Res.* **47**, D309–D314 (2019).
69. Newell, R. J. P., McMaster, E. S., Craig, P., Boden, M., Tyson, G. W., & Woodcroft, B. J. Lorikeet: strain-resolved metagenome analysis using local reassembly (v0.8.2). Zenodo. https://doi.org/10.5281/zenodo.10275469 (2023).
70. Fuentes-Santander, F., Curiqueo, C., Araos, R. & Ugalde, J. A. BugBuster: a novel automatic and reproducible workflow for metagenomic data analysis. *Bioinforma. Adv.* **5**, vbaf152 (2025).
71. Pan, S., Zhao, X. M. & Coelho, L. P. SemiBin2: self-supervised contrastive learning leads to better MAGs for short- and long-read sequencing. *Bioinformatics* **39**, i21–i29 (2023).
72. Wang, Z. *et al.* Effective binning of metagenomic contigs using contrastive multi-view representation learning. *Nat. Commun.* **15**, 1–14 (2024).
73. Chklovski, A., Parks, D. H., Woodcroft, B. J. & Tyson, G. W. CheckM2: a rapid, scalable and accurate tool for assessing microbial genome quality using machine learning. *Nat. Methods* **20**, 1203–1212 (2023).
74. Uritskiy, G. V., Diruggiero, J. & Taylor, J. MetaWRAP - A flexible pipeline for genome-resolved metagenomic data analysis. *Microbiome* **6**, 1–13 (2018).
75. Chaumeil, P.-A., Mussig, A. J., Hugenholtz, P. & Parks, D. H. GTDB-Tk v2: memory friendly classification with the genome taxonomy database. *Bioinformatics* **38**, 5315–5316 (2022).
76. Figueroa III, J. L., Dhungel, E., Bellanger, M., Brouwer, C. R. & White III, R. A. MetaCerberus: distributed highly parallelized HMM-based processing for robust functional annotation across the tree of life. *Bioinformatics* **40**, btae119 (2024).
77. Irber, L. *et al.* sourmash v4: A multitool to quickly search, compare, and analyze genomic and metagenomic data sets. *J. Open Source Softw.* **9**, 6830 (2024).
78. Arango-Argoty, G. *et al.* DeepARG: A deep learning approach for predicting antibiotic resistance genes from metagenomic data. *Microbiome* **6**, 1–15 (2018).
79. Olson, R. D. *et al.* Introducing the Bacterial and Viral Bioinformatics Resource Center (BV-BRC): a resource combining PATRIC, IRD and ViPR. *Nucleic Acids Res.* **51**, D678–D689 (2023).
80. Krueger, F. Source code for: A wrapper around Cutadapt and FastQC to consistently apply adapter and quality trimming to FastQ files, with extra functionality for RRBS data. https://github.com/FelixKrueger/TrimGalore (2023).
81. Altschul, S. F., Gish, W., Miller, W., Myers, E. W. & Lipman, D. J. Basic local alignment search tool. *J. Mol. Biol.* **215**, 403–410 (1990).
82. Parrello, B., Butler, R., Chlenski, P., Pusch, G. D. & Overbeek, R. Supervised extraction of near-complete genomes from metagenomic samples: A new service in PATRIC. *PLOS ONE* **16**, e0250092 (2021).
83. Parrello, B. *et al.* A machine learning-based service for estimating quality of genomes using PATRIC. *BMC Bioinformatics* **20**, 486 (2019).
84. Brettin, T. *et al.* RASTtk: A modular and extensible implementation of the RAST algorithm for building custom annotation pipelines and annotating batches of genomes. *Sci. Rep.* **5**, 8365 (2015).

 doi:10.20944/preprints202506.0703.v2

85. Wang, S., Sundaram, J. P. & Spiro, D. VIGOR, an annotation program for small viral genomes. *BMC Bioinformatics* **11**, 451 (2010).
86. Larsen, C. N. *et al.* Mat_peptide: comprehensive annotation of mature peptides from polyproteins in five virus families. *Bioinformatics* **36**, 1627–1628 (2020).
87. Benavides, A., Sanchez, F., Alzate, J. F. & Cabarcas, F. DATMA: Distributed Automatic Metagenomic Assembly and annotation framework. *PeerJ* **8**, (2020).
88. Magoč, T. & Salzberg, S. L. FLASH: fast length adjustment of short reads to improve genome assemblies. *Bioinformatics* **27**, 2957–2963 (2011).
89. Zerbino, D. R. & Birney, E. Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res.* **18**, 821–829 (2008).
90. Benavides, A., Isaza, J. P., Niño-García, J. P., Alzate, J. F. & Cabarcas, F. CLAME: a new alignment-based binning algorithm allows the genomic description of a novel *Xanthomonadaceae* from the Colombian Andes. *BMC Genomics* **19**, (2018).
91. Menzel, P., Ng, K. L. & Krogh, A. Fast and sensitive taxonomic classification for metagenomics with Kaiju. *Nat. Commun.* **7**, 11257 (2016).
92. Besemer, J. & Borodovsky, M. GeneMark: web software for gene finding in prokaryotes, eukaryotes and viruses. *Nucleic Acids Res.* **33**, W451–W454 (2005).
93. Ondov, B. D., Bergman, N. H. & Phillippy, A. M. Interactive metagenomic visualization in a Web browser. *BMC Bioinformatics* **12**, 1–10 (2011).
94. Bai, D. *et al.* EasyMetagenome: A user-friendly and flexible pipeline for shotgun metagenomic analysis in microbiome research. *iMeta* **4**, e70001 (2025).
95. Hyatt, D., LoCascio, P. F., Hauser, L. J. & Uberbacher, E. C. Gene and translation initiation site prediction in metagenomic sequences. *Bioinformatics* **28**, 2223–2230 (2012).
96. Cantalapiedra, C. P., Hernández-Plaza, A., Letunic, I., Bork, P. & Huerta-Cepas, J. eggNOG-mapper v2: Functional Annotation, Orthology Assignments, and Domain Prediction at the Metagenomic Scale. *Mol. Biol. Evol.* **38**, 5825–5829 (2021).
97. Olm, M. R., Brown, C. T., Brooks, B. & Banfield, J. F. dRep: a tool for fast and accurate genomic comparisons that enables improved genome recovery from metagenomes through de-replication. *ISME J.* **11**, 2864–2868 (2017).
98. Lu, J., Breitwieser, F. P., Thielen, P. & Salzberg, S. L. Bracken: Estimating species abundance in metagenomics data. *PeerJ Comput. Sci.* **2017**, e104 (2017).
99. Peng, K. *et al.* Benchmarking of analysis tools and pipeline development for nanopore long-read metagenomics. *Sci. Bull.* **70**, 1591–1595 (2025).
100. Danecek, P. *et al.* Twelve years of SAMtools and BCFtools. *GigaScience* **10**, giab008 (2021).
101. Quinlan, A. R. & Hall, I. M. BEDTools: a flexible suite of utilities for comparing genomic features. *Bioinformatics* **26**, 841–842 (2010).
102. Kajitani, R. *et al.* MetaPlatanus: a metagenome assembler that combines long-range sequence links and species-specific features. *Nucleic Acids Res.* **49**, e130 (2021).
103. Hu, J., Fan, J., Sun, Z. & Liu, S. NextPolish: a fast and efficient genome polishing tool for long-read assembly. *Bioinformatics* **36**, 2253–2255 (2020).
104. Asnicar, F. *et al.* Precise phylogenetic analysis of microbial isolates and genomes from metagenomes using PhyloPhlAn 3.0. *Nat. Commun.* **11**, 1–10 (2020).
105. Seemann, T. Prokka: rapid prokaryotic genome annotation. *Bioinformatics* **30**, 2068–2069 (2014).
106. Zhao, D. *et al.* Eukfinder: a pipeline to retrieve microbial eukaryote genome sequences from metagenomic data. *mBio* **16**, e00699-25 (2025).
107. Lin, H.-H. & Liao, Y.-C. Accurate binning of metagenomic contigs via automated clustering sequences using information of genomic signatures and marker genes. *Sci. Rep.* **6**, 24175 (2016).
108. Bengtsson-Palme, J. *et al.* metaxa2: improved identification and taxonomic classification of small and large subunit rRNA in metagenomic data. *Mol. Ecol. Resour.* **15**, 1403–1414 (2015).
109. Van Nguyen, H. & Lavenier, D. PLAST: parallel local alignment search tool for database comparison. *BMC Bioinformatics* **10**, 329 (2009).

 doi:10.20944/preprints202506.0703.v2

110. Cavalcante, J. V. F., Dantas de Souza, I., Morais, D. A. A. & Dalmolin, R. J. S. EURYALE: A versatile Nextflow pipeline for taxonomic classification and functional annotation of metagenomics data. in *2024 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)* 1–7 (2024).
111. Morais, D. A. A., Cavalcante, J. V. F., Monteiro, S. S., Pasquali, M. A. B. & Dalmolin, R. J. S. MEDUSA: A Pipeline for Sensitive Taxonomic Classification and Flexible Functional Annotation of Metagenomic Shotgun Sequences. *Front. Genet.* **13**, (2022).
112. NCBI Resource Coordinators. Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res.* **42**, D7–D17 (2014).
113. The Galaxy Community *et al.* The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2022 update. *Nucleic Acids Res.* **50**, W345–W351 (2022).
114. Li, H. Source code for: Seqtk. https://github.com/lh3/seqtk (2025).
115. von Meijenfeldt, F. A. B., Arkhipova, K., Cambuy, D. D., Coutinho, F. H. & Dutilh, B. E. Robust taxonomic classification of uncharted microbial sequences and bins with CAT and BAT. *Genome Biol.* **20**, 217 (2019).
116. Cornet, L. *et al.* The GEN-ERA toolbox: unified and reproducible workflows for research in microbial genomics. *GigaScience* **12**, 1–10 (2022).
117. Bankevich, A. *et al.* SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing. *J. Comput. Biol.* **19**, 455–477 (2012).
118. Walker, B. J. *et al.* Pilon: An Integrated Tool for Comprehensive Microbial Variant Detection and Genome Assembly Improvement. *PLOS ONE* **9**, e112963 (2014).
119. Alonge, M. *et al.* Automated assembly scaffolding using RagTag elevates a new tomato system for high-throughput genome editing. *Genome Biol.* **23**, 258 (2022).
120. Orakov, A. *et al.* GUNC: detection of chimerism and contamination in prokaryotic genomes. *Genome Biol.* **22**, 1–19 (2021).
121. Saary, P., Mitchell, A. L. & Finn, R. D. Estimating the quality of eukaryotic genomes recovered from metagenomic analysis with EukCC. *Genome Biol.* **21**, 244 (2020).
122. Manni, M., Berkeley, M. R., Seppey, M. & Zdobnov, E. M. BUSCO: Assessing Genomic Data Quality and Beyond. *Curr. Protoc.* **1**, e323 (2021).
123. Cornet, L. *et al.* Consensus assessment of the contamination level of publicly available cyanobacterial genomes. *PLOS ONE* **13**, e0200323 (2018).
124. Gurevich, A., Saveliev, V., Vyahhi, N. & Tesler, G. QUAST: quality assessment tool for genome assemblies. *Bioinformatics* **29**, 1072–1075 (2013).
125. Meunier, L., Baurain, D. & Cornet, L. AMAW: automated gene annotation for non-model eukaryotic genomes. *F1000Research* **12**, 186 (2023).
126. Brůna, T., Hoff, K. J., Lomsadze, A., Stanke, M. & Borodovsky, M. BRAKER2: automatic eukaryotic genome annotation with GeneMark-EP+ and AUGUSTUS supported by a protein database. *NAR Genomics Bioinforma.* **3**, lqaa108 (2021).
127. Queirós, P., Delogu, F., Hickl, O., May, P. & Wilmes, P. Mantis: flexible and consensus-driven genome annotation. *GigaScience* **10**, giab042 (2021).
128. Kanehisa, M., Sato, Y., Kawashima, M., Furumichi, M. & Tanabe, M. KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Res.* **44**, D457–D462 (2016).
129. Emms, D. M. & Kelly, S. OrthoFinder: phylogenetic orthology inference for comparative genomics. *Genome Biol.* **20**, 238 (2019).
130. Portik, D. M. *et al.* Highly accurate metagenome-assembled genomes from human gut microbiota using long-read assembly, binning, and consolidation methods. Preprint at https://doi.org/10.1101/2024.05.10.593587 (2024).
131. Kalantar, K. L. *et al.* IDseq—An open source cloud-based pipeline and analysis service for metagenomic pathogen detection and monitoring. *GigaScience* **9**, giaa111 (2020).
132. Dobin, A. *et al.* STAR: ultrafast universal RNA-seq aligner. *Bioinformatics* **29**, 15–21 (2013).
133. Niu, B., Fu, L., Sun, S. & Li, W. Artificial and natural duplicates in pyrosequencing reads of metagenomic data. *BMC Bioinformatics* **11**, 187 (2010).

 doi:10.20944/preprints202506.0703.v2

134. Wu, T. D. & Nacu, S. Fast and SNP-tolerant detection of complex variants and splicing in short reads. *Bioinformatics* **26**, 873–881 (2010).
135. Zhao, Y., Tang, H. & Ye, Y. RAPSearch2: a fast and memory-efficient protein similarity search tool for next-generation sequencing data. *Bioinformatics* **28**, 125–126 (2012).
136. Chen, I.-M. A. *et al.* The IMG/M data management and analysis system v.7: content updates and new features. *Nucleic Acids Res.* **51**, D723–D732 (2023).
137. Lomsadze, A., Gemayel, K., Tang, S. & Borodovsky, M. Modeling leaderless transcription and atypical genes results in more accurate gene prediction in prokaryotes. *Genome Res.* **28**, 1079–1089 (2018).
138. Mistry, J. *et al.* Pfam: The protein families database in 2021. *Nucleic Acids Res.* **49**, D412–D419 (2021).
139. Haft, D. H. *et al.* TIGRFAMs and Genome Properties in 2013. *Nucleic Acids Res.* **41**, D387–D395 (2013).
140. Petersen, T. N., Brunak, S., von Heijne, G. & Nielsen, H. SignalP 4.0: discriminating signal peptides from transmembrane regions. *Nat. Methods* **8**, 785–786 (2011).
141. Möller, S., Croning, M. D. R. & Apweiler, R. Evaluation of methods for the prediction of membrane spanning regions. *Bioinformatics* **17**, 646–653 (2001).
142. McCulloch, J. A. *et al.* JAMS - A framework for the taxonomic and functional exploration of microbiological genomic data. Preprint at https://doi.org/10.1101/2023.03.03.531026 (2023).
143. Zdobnov, E. M. & Apweiler, R. InterProScan – an integration platform for the signature-recognition methods in InterPro. *Bioinformatics* **17**, 847–848 (2001).
144. Chivian, D. *et al.* Metagenome-assembled genome extraction and analysis from microbiomes using KBase. *Nat. Protoc.* **18**, 208–238 (2023).
145. Zheng, J. *et al.* dbCAN3: automated carbohydrate-active enzyme and substrate annotation. *Nucleic Acids Res.* **51**, W115–W121 (2023).
146. Shaffer, M. *et al.* DRAM for distilling microbial metabolism to automate the curation of microbiome function. *Nucleic Acids Res.* **48**, 8883–8900 (2020).
147. Song, H.S. et al. OMEGGA: A Computationally Efficient Omics-Guided Global Gapfilling Algorithm for Phenotype-Consistent Metabolic Network Reconstruction. U.S. Department of Energy Genomic Science Program, (2023).
148. Faria, J. P. *et al.* ModelSEED v2: High-throughput genome-scale metabolic model reconstruction with enhanced energy biosynthesis pathway prediction. Preprint at https://doi.org/10.1101/2023.10.04.556561 (2023).
149. Jain, C., Rodriguez-R, L. M., Phillippy, A. M., Konstantinidis, K. T. & Aluru, S. High throughput ANI analysis of 90K prokaryotic genomes reveals clear species boundaries. *Nat. Commun.* **9**, 5114 (2018).
150. Price, M. N., Dehal, P. S. & Arkin, A. P. FastTree 2 – Approximately Maximum-Likelihood Trees for Large Alignments. *PLOS ONE* **5**, e9490 (2010).
151. Edgar, R. C. Muscle5: High-accuracy alignment ensembles enable unbiased assessments of sequence homology and phylogeny. *Nat. Commun.* **13**, 6968 (2022).
152. Churcheward, B., Millet, M., Bihouée, A., Fertin, G. & Chaffron, S. MAGNETO: An Automated Workflow for Genome-Resolved Metagenomics. *mSystems* **7**, (2022).
153. Wingett SW and Andrews S. FastQ Screen: A tool for multi-genome mapping and quality control. *F1000Research* **7**, 1338 (2018).
154. Benoit, G. *et al.* Multiple comparative metagenomics using multiset k-mer counting. *PeerJ Comput. Sci.* **2**, e94 (2016).
155. Steinegger, M. & Söding, J. Clustering huge protein sequence sets in linear time. *Nat. Commun.* **9**, 2542 (2018).
156. Sunagawa, S. *et al.* Metagenomic species profiling using universal phylogenetic marker genes. *Nat. Methods* **10**, 1196–1199 (2013).
157. Murovec, B., Deutsch, L. & Stres, B. Computational Framework for High-Quality Production and Large-Scale Evolutionary Analysis of Metagenome Assembled Genomes. *Mol. Biol. Evol.* **37**, 593–598 (2020).
158. Page, A. J. *et al.* Roary: rapid large-scale prokaryote pan genome analysis. *Bioinformatics* **31**, 3691–3693 (2015).

 doi:10.20944/preprints202506.0703.v2

159. Wu, Y.-W. ezTree: an automated pipeline for identifying phylogenetic marker genes and inferring evolutionary relationships among uncultivated prokaryotic draft genomes. *BMC Genomics* **19**, 921 (2018).
160. Maurice, N., Lemaitre, C., Vicedomini, R. & Frioux, C. Mapler: a pipeline for assessing assembly quality in taxonomically rich metagenomes sequenced with HiFi reads. *Bioinformatics* **41**, btaf334 (2025).
161. Benoit, G. *et al.* High-quality metagenome assembly from long accurate reads with metaMDBG. *Nat. Biotechnol.* **42**, 1378–1383 (2024).
162. Mapleson, D., Garcia Accinelli, G., Kettleborough, G., Wright, J. & Clavijo, B. J. KAT: a K-mer analysis toolkit to quality control NGS datasets and genome assemblies. *Bioinformatics* **33**, 574–576 (2017).
163. Zorrilla, F., Buric, F., Patil, K. R. & Zelezniak, A. metaGEM: reconstruction of genome scale metabolic models directly from metagenomes. *Nucleic Acids Res.* **49**, e126 (2021).
164. Machado, D., Andrejev, S., Tramontano, M. & Patil, K. R. Fast automated reconstruction of genome-scale metabolic models for microbial species and communities. *Nucleic Acids Res.* **46**, 7542–7553 (2018).
165. Zelezniak, A. *et al.* Metabolic dependencies drive species co-occurrence in diverse microbial communities. *Proc. Natl. Acad. Sci.* **112**, 6449–6454 (2015).
166. Lieven, C. *et al.* MEMOTE for standardized genome-scale metabolic model testing. *Nat. Biotechnol.* **38**, 272–276 (2020).
167. Emiola, A. & Oh, J. High throughput in situ metagenomic measurement of bacterial replication at ultra-low sequencing coverage. *Nat. Commun.* **9**, 4956 (2018).
168. Shaban, B. et al. MetaGenePipe: An Automated, Portable Pipeline for Contig-based Functional and Taxonomic Analysis. *The Journal of Open Source Software* **8**, 4851 (2023).
169. Poux, S. *et al.* On expert curation and scalability: UniProtKB/Swiss-Prot as a case study. *Bioinformatics* **33**, 3454–3460 (2017).
170. Eddy, S. R. A Probabilistic Model of Local Sequence Alignment That Simplifies Statistical Significance Estimation. *PLOS Comput. Biol.* **4**, e1000069 (2008).
171. Aramaki, T. *et al.* KofamKOALA: KEGG Ortholog assignment based on profile HMM and adaptive score threshold. *Bioinformatics* **36**, 2251–2252 (2020).
172. Kieser, S., Brown, J., Zdobnov, E. M., Trajkovski, M. & McCue, L. A. ATLAS: A Snakemake workflow for assembly, annotation, and genomic binning of metagenome sequence data. *BMC Bioinformatics* **21**, 1–8 (2020).
173. Belmann, P. *et al.* Metagenomics-Toolkit: the flexible and efficient cloud-based metagenomics workflow featuring machine learning-enabled resource allocation. *NAR Genomics Bioinforma.* **7**, lqaf093 (2025).
174. Kokot, M., Długosz, M. & Deorowicz, S. KMC 3: counting and manipulating k-mer statistics. *Bioinformatics* **33**, 2759–2761 (2017).
175. Rodriguez-R, L. M., Gunturu, S., Tiedje, J. M., Cole, J. R. & Konstantinidis, K. T. Nonpareil 3: Fast Estimation of Metagenomic Coverage and Sequence Diversity. *mSystems* **3**, 10.1128/msystems.00039-18 (2018).
176. Mallawaarachchi, V. & Lin, Y. MetaCoAG: Binning Metagenomic Contigs via Composition, Coverage and Assembly Graphs. in *Research in Computational Molecular Biology* (ed. Pe'er, I.) 70–85 (2022).
177. Wang, Z., Huang, P., You, R., Sun, F. & Zhu, S. MetaBinner: a high-performance and stand-alone ensemble binning method to recover individual genomes from complex microbial communities. *Genome Biol.* **24**, 1–18 (2023).
178. Mirdita, M., Steinegger, M., Breitwieser, F., Söding, J. & Levy Karin, E. Fast and sensitive taxonomic assignment to metagenomic contigs. *Bioinformatics* **37**, 3029–3031 (2021).
179. Alcock, B. P. *et al.* CARD 2023: expanded curation, support for machine learning, and resistome prediction at the Comprehensive Antibiotic Resistance Database. *Nucleic Acids Res.* **51**, D690–D699 (2023).
180. Zimmermann, J., Kaleta, C. & Waschina, S. gapseq: informed prediction of bacterial metabolic pathways and reconstruction of accurate metabolic models. *Genome Biol.* **22**, 81 (2021).
181. Pritchard, L., Glover, R. H., Humphris, S., Elphinstone, J. G. & Toth, I. K. Genomics and taxonomy in diagnostics for food security: soft-rotting enterobacterial plant pathogens. *Anal. Methods* **8**, 12–24 (2015).
182. Wittler, R. Alignment- and reference-free phylogenomics with colored de Bruijn graphs. *Algorithms Mol. Biol.* **15**, 4 (2020).

 doi:10.20944/preprints202506.0703.v2

183. Salazar, V. W. *et al.* Metaphor—A workflow for streamlined assembly and binning of metagenomes. *GigaScience* **12**, 1–12 (2023).
184. Mainguy, J. *et al.* metagWGS, a comprehensive workflow to analyze metagenomic data using Illumina or PacBio HiFi reads. Preprint at https://doi.org/10.1101/2024.09.13.612854 (2024).
185. Joshi NA & Fass JN. Source code for: Sickle-A sliding-window, adaptive, quality-based trimming tool for FastQ files. https://github.com/najoshi/sickle (2014).
186. Mainguy, J. & Hoede, C. Binette: a fast and accurate bin refinement tool to construct high quality Metagenome Assembled Genomes. *J. Open Source Softw.* **9**, 6782 (2024).
187. Kumar, S., Jones, M., Koutsovoulos, G., Clarke, M. & Blaxter, M. Blobology: exploring raw genome data for contaminants, symbionts and parasites using taxon-annotated GC-coverage plots. *Front. Genet.* **4**, (2013).
188. Hildebrand, F. *et al.* Dispersal strategies shape persistence and evolution of human gut bacteria. *Cell Host Microbe* **29**, 1167-1176.e9 (2021).
189. Hildebrand, F., Tadeo, R., Voigt, A. Y., Bork, P. & Raes, J. LotuS: an efficient and user-friendly OTU processing pipeline. *Microbiome* **2**, 30 (2014).
190. Kolmogorov, M., Yuan, J., Lin, Y. & Pevzner, P. A. Assembly of long, error-prone reads using repeat graphs. *Nat. Biotechnol.* **37**, 540–546 (2019).
191. Liu, C.-C. *et al.* MetaDecoder: a novel method for clustering metagenomic contigs. *Microbiome* **10**, 46 (2022).
192. Drula, E. *et al.* The carbohydrate-active enzyme database: functions and literature. *Nucleic Acids Res.* **50**, D571–D577 (2022).
193. Milanese, A. *et al.* Microbial abundance, activity and population genomic profiling with mOTUs2. *Nat. Commun.* **10**, 1–11 (2019).
194. Blanco-Míguez, A. *et al.* Extending and improving metagenomic taxonomic profiling with uncharacterized species using MetaPhlAn 4. *Nat. Biotechnol.* 1–12 (2023).
195. Garrison, E. & Marth, G. Haplotype-based variant detection from short-read sequencing. Preprint at https://doi.org/10.48550/arXiv.1207.3907 (2012).
196. Cokelaer, T., Desvillechabrol, D., Legendre, R. & Cardon, M. 'Sequana': a Set of Snakemake NGS pipelines. *J. Open Source Softw.* **2**, 352 (2017).
197. Richardson, L. *et al.* MGnify: the microbiome sequence data analysis resource in 2023. *Nucleic Acids Res.* **51**, D753–D759 (2023).
198. Cock, P. J. A. *et al.* Biopython: freely available Python tools for computational molecular biology and bioinformatics. *Bioinformatics* **25**, 1422–1423 (2009).
199. Suzek, B. E. *et al.* UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. *Bioinformatics* **31**, 926–932 (2015).
200. Rho, M., Tang, H. & Ye, Y. FragGeneScan: predicting genes in short and error-prone reads. *Nucleic Acids Res.* **38**, e191 (2010).
201. Blin, K. *et al.* antiSMASH 7.0: new and improved predictions for detection, regulation, chemical structures and visualisation. *Nucleic Acids Res.* **51**, W46–W50 (2023).
202. Ziemski, M. *et al.* MOSHPIT: accessible, reproducible metagenome data science on the QIIME 2 framework. Preprint at https://doi.org/10.1101/2025.01.27.635007 (2025).
203. Damme, R. van *et al.* Metagenomics workflow for hybrid assembly, differential coverage binning, metatranscriptomics and pathway analysis (MUFFIN). *PLOS Comput. Biol.* **17**, 1–13 (2021).
204. Parks, D. H. *et al.* GTDB: an ongoing census of bacterial and archaeal diversity through a phylogenetically consistent, rank normalized and complete genome-based taxonomy. *Nucleic Acids Res.* **50**, D785–D794 (2022).
205. Patro, R., Duggal, G., Love, M. I., Irizarry, R. A. & Kingsford, C. Salmon provides fast and bias-aware quantification of transcript expression. *Nat. Methods* **14**, 417–419 (2017).
206. Grabherr, M. G. *et al.* Full-length transcriptome assembly from RNA-Seq data without a reference genome. *Nat. Biotechnol.* **29**, 644–652 (2011).
207. Liu, L., Yang, Y., Deng, Y. & Zhang, T. Nanopore long-read-only metagenomics enables complete and high-quality genome reconstruction from mock and complex metagenomes. *Microbiome* **10**, 209 (2022).

 doi:10.20944/preprints202506.0703.v2

208. Vaser, R., Sović, I., Nagarajan, N. & Šikić, M. Fast and accurate de novo genome assembly from long uncorrected reads. *Genome Res.* **27**, 737–746 (2017).
209. Oxford Nanopore Technologies Ltd. Source code for: medaka-Sequence correction provided by ONT Research. https://github.com/nanoporetech/medaka (2025).
210. The UniProt Consortium. UniProt: the Universal Protein Knowledgebase in 2025. *Nucleic Acids Res.* **53**, D609–D617 (2025).
211. Krakau, S., Straub, D., Gourlé, H., Gabernet, G. & Nahnsen, S. nf-core/mag: a best-practice pipeline for metagenome hybrid assembly and binning. *NAR Genomics Bioinforma.* **4**, (2022).
212. Schubert, M., Lindgreen, S. & Orlando, L. AdapterRemoval v2: rapid adapter trimming, identification, and read merging. *BMC Res. Notes* **9**, 88 (2016).
213. Levy Karin, E., Mirdita, M. & Söding, J. MetaEuk—sensitive, high-throughput gene discovery, and annotation for large-scale eukaryotic metagenomics. *Microbiome* **8**, 48 (2020).
214. Borry, M., Hübner, A., Rohrlach, A. B. & Warinner, C. PyDamage: automated ancient damage identification and estimation for contigs in ancient DNA de novo assembly. *PeerJ* **9**, e11845 (2021).
215. Camargo, A. P. *et al.* Identification of mobile genetic elements with geNomad. *Nat. Biotechnol.* **42**, 1303–1312 (2024).
216. Karlicki, M., Antonowicz, S. & Karnkowska, A. Tiara: deep learning-based classification system for eukaryotic sequences. *Bioinformatics* **38**, 344–350 (2022).
217. Almeida, F. M. de, Campos, T. A. de & Pappas, G. J. Scalable and versatile container-based pipelines for de novo genome assembly and bacterial annotation. *F1000Research* **12**, 1205 (2023).
218. Seemann, T. Source code for: Shovill-Assemble bacterial isolate genomes from Illumina paired-end reads. https://github.com/tseemann/shovill (2020).
219. Haghshenas, E., Asghari, H., Stoye, J., Chauve, C. & Hach, F. HASLR: Fast Hybrid Assembly of Long Reads. *iScience* **23**, 101389 (2020).
220. Vaser, R. & Šikić, M. Time- and memory-efficient genome assembly with Raven. *Nat. Comput. Sci.* **1**, 332–336 (2021).
221. Shafin, K. *et al.* Nanopore sequencing and the Shasta toolkit enable efficient de novo assembly of eleven human genomes. *Nat. Biotechnol.* **38**, 1044–1053 (2020).
222. Ruan, J. & Li, H. Fast and accurate long-read assembly with wtdbg2. *Nat. Methods* **17**, 155–158 (2020).
223. Graham, E. D., Heidelberg, J. F. & Tully, B. J. Potential for primary productivity in a globally-distributed bacterial phototroph. *ISME J.* **12**, 1861–1866 (2018).
224. Seemann, T. Source code for: Barrnap-Bacterial ribosomal RNA predictor. https://github.com/tseemann/shovill (2018).
225. Feldgarden, M. *et al.* AMRFinderPlus and the Reference Gene Catalog facilitate examination of the genomic links among antimicrobial resistance, stress response, and virulence. *Sci. Rep.* **11**, 12728 (2021).
226. Starikova, E. V. *et al.* Phigaro: high-throughput prophage sequence annotation. *Bioinformatics* **36**, 3882–3884 (2020).
227. Dong, W. *et al.* An expanded database and analytical toolkit for identifying bacterial virulence factors and their associations with chronic diseases. *Nat. Commun.* **15**, 8084 (2024).
228. Carattoli, A. *et al.* In Silico Detection and Typing of Plasmids using PlasmidFinder and Plasmid Multilocus Sequence Typing. *Antimicrob. Agents Chemother.* **58**, 3895–3903 (2014).
229. Jolley, K. A. & Maiden, M. C. BIGSdb: Scalable analysis of bacterial genome variation at the population level. *BMC Bioinformatics* **11**, 595 (2010).
230. Schwengers, O. *et al.* Platon: identification and characterization of bacterial plasmid contigs in short-read draft assemblies exploiting protein sequence-based replicon distribution scores. *Microb. Genomics* **6**, e000398 (2020).
231. Arndt, D. *et al.* PHASTER: a better, faster version of the PHAST phage search tool. *Nucleic Acids Res.* **44**, W16–W21 (2016).
232. Arango-Argoty, G. A. *et al.* ARGminer: a web platform for the crowdsourcing-based curation of antibiotic resistance genes. *Bioinformatics* **36**, 2966–2973 (2020).

 doi:10.20944/preprints202506.0703.v2

233. Florensa, A. F., Kaas, R. S., Clausen, P. T. L. C., Aytan-Aktug, D. & Aarestrup, F. M. ResFinder – an open online resource for identification of antimicrobial resistance genes in next-generation sequencing data and prediction of phenotypes from genotypes. *Microb. Genomics* **8**, 000748 (2022).
234. Narayanasamy, S. *et al.* IMP: a pipeline for reproducible reference-independent integrated metagenomic and metatranscriptomic analyses. *Genome Biol.* **17**, 260 (2016).
235. Kopylova, E., Noé, L. & Touzet, H. SortMeRNA: fast and accurate filtering of ribosomal RNAs in metatranscriptomic data. *Bioinformatics* **28**, 3211–3217 (2012).
236. Schudoma, C. Source code for: gff_quantifier. https://github.com/cschu/gff_quantifier (2023).
237. Bray, N. L., Pimentel, H., Melsted, P. & Pachter, L. Near-optimal probabilistic RNA-seq quantification. *Nat. Biotechnol.* **34**, 525–527 (2016).
238. Tadrent, N. *et al.* SnakeMAGs: a simple, efficient, flexible and scalable workflow to reconstruct prokaryotic genomes from metagenomes. *F1000Research* **11**, 1522 (2023).
239. Schmidt, T. S. B. *et al.* SPIRE: a Searchable, Planetary-scale mIcrobiome REsource. *Nucleic Acids Res.* **52**, D777–D783 (2024).
240. Coelho, L. P. *et al.* NG-meta-profiler: fast processing of metagenomes using NGLess, a domain-specific language. *Microbiome* **7**, 84 (2019).
241. Seemann, T. Source code for: ABRicate-Mass screening of contigs for antimicrobial and virulence genes. https://github.com/tseemann/abricate (2020).
242. Doster, E. *et al.* MEGARes 2.0: a database for classification of antimicrobial drug, biocide and metal resistance determinants in metagenomic sequence data. *Nucleic Acids Res.* **48**, D561–D569 (2020).
243. Santos-Júnior, C. D., Pan, S., Zhao, X.-M. & Coelho, L. P. Macrel: antimicrobial peptide screening in genomes and metagenomes. *PeerJ* **8**, e10555 (2020).
244. Ondov, B. D. *et al.* Mash: fast genome and metagenome distance estimation using MinHash. *Genome Biol.* **17**, 132 (2016).
245. Tamames, J. & Puente-Sánchez, F. SqueezeMeta, a highly portable, fully automatic metagenomic analysis pipeline. *Front. Microbiol.* **10**, 3349 (2019).
246. Schmieder, R. & Edwards, R. Quality control and preprocessing of metagenomic datasets. *Bioinformatics* **27**, 863–864 (2011).
247. Parks, D. H. Source code for: CompareM-A toolbox for comparative genomics. https://github.com/donovan-h-parks/CompareM (2020).
248. Marçais, G. *et al.* MUMmer4: A fast and versatile genome alignment system. *PLOS Comput. Biol.* **14**, e1005944 (2018).
249. Puente-Sánchez, F., García-García, N. & Tamames, J. SQMtools: automated processing and visual analysis of 'omics data with R and anvi'o. *BMC Bioinformatics* **21**, 358 (2020).
250. Sjöqvist, C., Delgado, L. F., Alneberg, J. & Andersson, A. F. Ecologically coherent population structure of uncultivated bacterioplankton. *ISME J.* **15**, 3034–3049 (2021).
251. Clarke, E. L. *et al.* Sunbeam: An extensible pipeline for analyzing metagenomic sequencing experiments. *Microbiome* **7**, 1–13 (2019).
252. Espinoza, J. L. *et al.* Unveiling the microbial realm with VEBA 2.0: a modular bioinformatics suite for end-to-end genome-resolved prokaryotic, (micro)eukaryotic and viral multi-omics from either short- or long-read sequencing. *Nucleic Acids Res.* **52**, e63 (2024).
253. Bushmanova, E., Antipov, D., Lapidus, A. & Prjibelski, A. D. rnaSPAdes: a de novo transcriptome assembler and its application to RNA-Seq data. *GigaScience* **8**, giz100 (2019).
254. Nayfach, S. *et al.* CheckV assesses the quality and completeness of metagenome-assembled viral genomes. *Nat. Biotechnol.* **39**, 578–585 (2021).
255. Ren, J., Ahlgren, N. A., Lu, Y. Y., Fuhrman, J. A. & Sun, F. VirFinder: a novel k-mer based tool for identifying viral sequences from assembled metagenomic data. *Microbiome* **5**, 69 (2017).
256. Zdouc, M. M. *et al.* MIBiG 4.0: advancing biosynthetic gene cluster curation through global collaboration. *Nucleic Acids Res.* **53**, D678–D690 (2025).
257. Eberhardt, R. Y. *et al.* AntiFam: a tool to help identify spurious ORFs in protein annotation. *Database* **2012**, bas003 (2012).

 doi:10.20944/preprints202506.0703.v2

258. Ruiz-Perez, C. A., Conrad, R. E. & Konstantinidis, K. T. MicrobeAnnotator: a user-friendly, comprehensive functional annotation pipeline for microbial genomes. *BMC Bioinformatics* **22**, 1–16 (2021).
259. Shaw, J. & Yu, Y. W. Rapid species-level metagenome profiling and containment estimation with sylph. *Nat. Biotechnol.* **43**, 1348–1359 (2025).
260. Weber, N. *et al.* Nephele: a cloud platform for simplified, standardized and reproducible microbiome data analysis. *Bioinformatics* **34**, 1411–1413 (2018).
261. Ye, Y. & Doak, T. G. A Parsimony Approach to Biological Pathway Reconstruction/Inference for Genomes and Metagenomes. *PLOS Comput. Biol.* **5**, e1000465 (2009).
262. Goussarov, G. *et al.* Benchmarking short-, long- and hybrid-read assemblers for metagenome sequencing of complex microbial communities. *Microbiology* **170**, 001469 (2024).
263. Meyer, F. *et al.* Tutorial: assessing metagenomics software with the CAMI benchmarking toolkit. *Nat. Protoc.* **16**, 1785–1801 (2021).
264. Wang, Z., Wang, Y., Fuhrman, J. A., Sun, F. & Zhu, S. Assessment of metagenomic assemblers based on hybrid reads of real and simulated metagenomic sequences. *Brief. Bioinform.* **21**, 777–790 (2020).
265. Rozov, R., Goldshlager, G., Halperin, E. & Shamir, R. Faucet: streaming de novo assembly graph construction. *Bioinformatics* **34**, 147–154 (2018).
266. Brown, C. L. *et al.* Critical evaluation of short, long, and hybrid assembly for contextual analysis of antibiotic resistance genes in complex environmental metagenomes. *Sci. Rep.* **11**, 3753 (2021).
267. Herazo-Álvarez, J., Mora, M., Cuadros-Orellana, S., Vilches-Ponce, K. & Hernández-García, R. A review of neural networks for metagenomic binning. *Brief. Bioinform.* **26**, bbaf065 (2025).
268. Cansdale, A. & Chong, J. P. J. MAGqual: a stand-alone pipeline to assess the quality of metagenome-assembled genomes. *Microbiome* **12**, 1–10 (2024).
269. Imelfort, M. *et al.* GroopM: an automated tool for the recovery of population genomes from related metagenomes. *PeerJ* **2**, e603 (2014).
270. Yue, Y. *et al.* Evaluating metagenomics tools for genome binning with real metagenomic datasets and CAMI datasets. *BMC Bioinformatics* **21**, 1–15 (2020).
271. Yepes-García, J. & Falquet, L. Metagenome quality metrics and taxonomical annotation visualization through the integration of MAGFlow and BIgMAG. *F1000Research* **13**, 640, (2024).
272. Simion, P. *et al.* A Large and Consistent Phylogenomic Dataset Supports Sponges as the Sister Group to All Other Animals. *Curr. Biol.* **27**, 958–967 (2017).
273. Edwin, N. R., Fitzpatrick, A. H., Brennan, F., Abram, F. & O'Sullivan, O. An in-depth evaluation of metagenomic classifiers for soil microbiomes. *Environ. Microbiome* **19**, 19 (2024).
274. Timilsina, M., Chundru, D., Pradhan, A. K., Blaustein, R. A. & Ghanem, M. Benchmarking Metagenomic Pipelines for the Detection of Foodborne Pathogens in Simulated Microbial Communities. *J. Food Prot.* **88**, 100583 (2025).
275. Irankhah, L., Khorsand, B., Naghibzadeh, M. & Savadi, A. Analyzing the performance of short-read classification tools on metagenomic samples toward proper diagnosis of diseases. *J. Bioinform. Comput. Biol.* **22**, 2450012 (2024).
276. Van Uffelen, A. *et al.* Benchmarking bacterial taxonomic classification using nanopore metagenomics data of several mock communities. *Sci. Data* **11**, 864 (2024).
277. Liang, Q., Bible, P. W., Liu, Y., Zou, B. & Wei, L. DeepMicrobes: taxonomic classification for metagenomics with deep learning. *NAR Genomics Bioinforma.* **2**, lqaa009 (2020).
278. Pusadkar, V. & Azad, R. K. Benchmarking Metagenomic Classifiers on Simulated Ancient and Modern Metagenomic Data. *Microorganisms* **11**, 2478 (2023).
279. Marić, J., Križanović, K., Riondet, S., Nagarajan, N. & Šikić, M. Comparative analysis of metagenomic classifiers for long-read sequencing datasets. *BMC Bioinformatics* **25**, 15 (2024).
280. Lin, B., Luo, X., Liu, Y. & Jin, X. A comprehensive review and comparison of existing computational methods for protein function prediction. *Brief. Bioinform.* **25**, bbae289 (2024).
281. The Gene Ontology Consortium. The Gene Ontology Resource: 20 years and still GOing strong. *Nucleic Acids Res.* **47**, D330–D338 (2019).

 doi:10.20944/preprints202506.0703.v2

282. Rawlings, N. D. *et al.* The MEROPS database of proteolytic enzymes, their substrates and inhibitors in 2017 and a comparison with peptidases in the PANTHER database. *Nucleic Acids Res.* **46**, D624–D632 (2018).
283. Zeller, M. & Huson, D. H. Comparison of functional classification systems. *NAR Genomics Bioinforma.* **4**, lqac090 (2022).
284. Xu, Y. *et al.* A new massively parallel nanoball sequencing platform for whole exome research. *BMC Bioinformatics* **20**, 153 (2019).
285. Liang, H. *et al.* Efficiently constructing complete genomes with CycloneSEQ to fill gaps in bacterial draft assemblies. *Gigabyte* **2025**, gigabyte154-0 (2025).
286. Kim, H.-M. *et al.* Comparative analysis of 7 short-read sequencing platforms using the Korean Reference Genome: MGI and Illumina sequencing benchmark for whole-genome sequencing. *GigaScience* **10**, giab014 (2021).
287. Vosloo, S. *et al.* Evaluating de Novo Assembly and Binning Strategies for Time Series Drinking Water Metagenomes. *Microbiol. Spectr.* **9**, (2021).
288. Lynn, H. M. & Gordon, J. I. Sequential co-assembly reduces computational resources and errors in metagenome-assembled genomes. *Cell Rep. Methods* **5**, (2025).
289. Goldfarb, T. *et al.* NCBI RefSeq: reference sequence standards through 25 years of curation and annotation. *Nucleic Acids Res.* **53**, D243–D257 (2025).
290. Arkin, A. P. *et al.* KBase: The United States Department of Energy Systems Biology Knowledgebase. *Nat. Biotechnol.* **36**, 566–569 (2018).
291. Achudhan, A. B., Kannan, P., Gupta, A. & Saleena, L. M. A Review of Web-Based Metagenomics Platforms for Analysing Next-Generation Sequence Data. *Biochem. Genet.* **62**, 621–632 (2024).
292. Wratten, L., Wilm, A. & Göke, J. Reproducible, scalable, and shareable analysis pipelines with bioinformatics workflow managers. *Nat. Methods* **18**, 1161–1168 (2021).
293. Köster, J. *et al.* Sustainable data analysis with Snakemake. *F1000Research* **10**, 33 (2021).
294. Tommaso, P. D. *et al.* Nextflow enables reproducible computational workflows. *Nat. Biotechnol.* **35**, 316–319 (2017).
295. OpenWDL. Source code for: Specification for the Workflow Description Language (WDL). https://github.com/openwdl/wdl (2025).
296. Ewels, P. A. *et al.* The nf-core framework for community-curated bioinformatics pipelines. *Nat. Biotechnol.* **38**, 276–278 (2020).
297. Roach, M. J. *et al.* Ten simple rules and a template for creating workflows-as-applications. *PLOS Comput. Biol.* **18**, e1010705 (2022).
298. Reiter, T. *et al.* Streamlining data-intensive biology with workflow systems. *GigaScience* **10**, (2021).
299. Kadri, S., Sboner, A., Sigaras, A. & Roy, S. Containers in Bioinformatics: Applications, Practical Considerations, and Best Practices in Molecular Pathology. *J. Mol. Diagn.* **24**, 442–454 (2022).
300. Badia, R. M. *et al.* COMP Superscalar, an interoperable programming framework. *SoftwareX* **3–4**, 32–36 (2015).
301. Espinoza, J. L. Source code for: Genopype-Architecture for creating bash pipelines, in particular, for bioinformatics. https://github.com/jolespin/genopype (2025).