Protein k-mers enable assembly-free microbial metapangenomics

This manuscript (permalink) was automatically generated from taylorreiter/2021-paper-metapangenomes@1e5b083 on March 11, 2022.

Authors

- **Taylor E. Reiter**
  0000-0002-7388-421X · taylorreiter · ReiterTaylor
  Department of Population Health and Reproduction, University of California, Davis · Funded by Grant XXXXXXXX

- **N. Tessa Pierce-Ward**
  0000-0002-2942-5331 · bluegenes · saltyscientist
  Department of Population Health and Reproduction, University of California, Davis · Funded by NSF 1711984

- **Luiz Irber**
  0000-0003-4371-9659 · luizirber · luizirber
  Graduate Group in Computer Science, UC Davis; Department of Population Health and Reproduction, University of California, Davis

- **Olga Borisovna Botvinnik**
  0000-0003-4412-7970 · olgabot · olgabot
  Data Sciences Platform, Chan Zuckerberg Biohub

- **C. Titus Brown**
  0000-0001-6001-2677 · ctb
  Department of Population Health and Reproduction, University of California, Davis

Abstract

Introduction

Short read metagenomic sequencing has expanded our knowledge of microbial communities and diversity [1,2,3]. In particular, metagenome assembly and genome binning or annotation have produced catalogs of metagenome-assembled genomes and genes, revealing new species and functional potentially previously unobserved in cultured organisms [1,2,4].

Along with advances in metagenome sequencing and analysis, the concept of metapangenomics has arisen as a framework for understanding how sets of metagenome-derived genes that are attributable to a group of organisms correlate with parameters in the environments in which they are sampled from [5,6,7]. Metapangenomic methods borrow heavily from pangenome analysis. Pangenomes comprise all genomic elements – usually open reading frames or genes – found within a group of organisms and reflect the metabolic and ecological plasticity of that group [8,9]. The pangenome is divided into core and accessory genes, where core genes are shared by almost all members in the group and accessory genes are not. Core genes often encode primary metabolism or other functions necessary for a group to live in a given environment [10], while accessory genes encode functions that facilitate adaptation to changing environments [9]. The size of the pangenome reflects the diversity of the organisms in a pangenome (population size, number of organisms sampled) as well as the ability of those organisms to adapt to different niches [8]. Open pangenomes are those which increase indefinitely in size when adding new genomes, while closed pangenomes do not.

While pangenomes are traditionally inferred from isolate genomes, metapangenomics extends the ecological framework of pangenomics to metagenomes. Metapangenomics gives insight into the genes that support specific environmental adaptations by applying pangenome methods to metagenome assembled genomes (MAGs) [6], or by mapping metagenomes against isolate-inferred pangenomes [5]. Both methods give valuable insight into the presence and distribution of functional content in natural microbial communities, but either may introduce biases associated with unknown sequencing content [11]. MAGs are often incomplete or unrecoverable due to low sequencing coverage or large amounts of variation (SNPs, indels, rearrangements, horizontal gene transfer, sequencing error, etc.), both of which cause short read assemblers to produce unbinnable short contiguous sequences. Unbinned sequences are disproportionately comprised of genomic islands and plasmids [12], hot spots for evolution that support microbial adaptation to changing environments [13]. In contrast, read mapping against isolate-inferred pangenomes may miss functional content present in the metagenome but missing from references, especially for species under represented or unrecorded in reference databases.

These issues are not exclusive to metapangenome inference, and many recently developed analysis strategies overcome some of these biases. These techniques largely rely on k-mers, words of length *k* in DNA or protein sequences. Metagenome k-mer profiles contain all sequences in a metagenome, including those which may not assemble or bin, or which aren't in reference databases. Long k-mers are also taxonomy-specific, where increasing k-mer length leads to sub-species discriminatory power [14] (CITE: TESSA). These properties have popularized the use of k-mers for metagenome analysis, primarily through lightweight sketching and compact de Bruijn assembly graphs (cDBGs). Lightweight sketching facilitates fast and accurate sequence comparisons between potentially large data sets through random but consistent sub-sampling [15,16]. cDBGs maintain connectivity between k-mers and organize them into species-specific neighborhoods [17,18].

To more fully represent the functional potential in metapangenomes, we present an analysis approach that relies on amino acid k-mers and assembly graph queries to estimate microbial (meta)pangenomes. This approach for metapangenome estimation is minimally reliant on reference databases and is assembly-free.

Results

In an effort to reconstruct metapangenomes without loss of information from assembly and binning [12,17,18,19,20,21], we demonstrate a pipeline that relies on k-mers and assembly graphs for metapangenome estimation (**Figure 1**). We first show that amino acid k-mers accurately estimate microbial pangenomes by comparing amino acid profiles of proteomes (translated coding domain sequences) against the proteomes themselves (**Figure 1 A**). To derive amino acid k-mers directly from shotgun metagenome reads, we next demonstrate the accuracy of a tool called orpheum for open reading frame prediction from short sequencing reads (**Figure 1 B**). We use assembly graph genome queries to retrieve species-specific reads from the metagenome, predict open reading frames from those reads using orpheum, and build a metapangenome using protein k-mers (**Figure 1 C**). We then apply this method to species present over time in a time series metagenomes from a human gut microbiome.

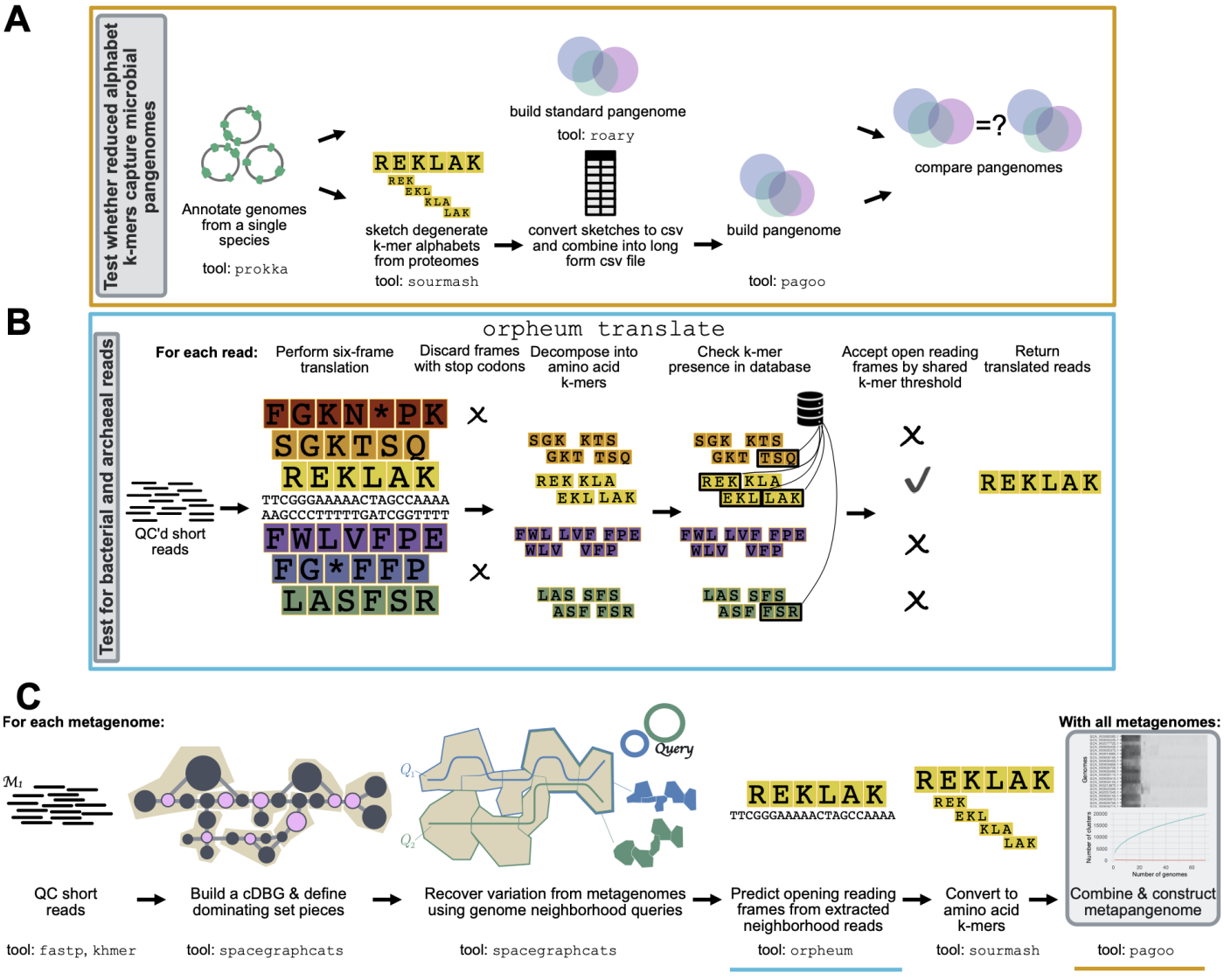

Figure 1: Overview of the pipeline used to build metapangenomes. Approaches that were developed or tested in this manuscript are outlined in grey. **A)** We tested whether degenerate k-mer alphabets could accurately represent bacterial and archaeal pangenomes. Using genomes annotated with prokka, we compared pangenomes built with roary, a field-standard pipeline, against pangenomes built with degenerate k-mer alphabet sketches. **B)** We tested whether

open reading frames could be predicted directly from short sequencing reads using the tool orpheum. This panel is modified from [22]. **C)** We combined this approaches with metagenome assembly graph genome queries to estimate metapangenomes directly from metagenomes without assembly or binning. The blue and orange lines correspond to steps tested in panels **A** and **B**.

Reduced alphabet k-mers accurately estimate characteristics of microbial pangenomes

Pangenomes from isolates are typically built by assembling each isolate genome and predicting genes (open reading frames), clustering gene sequences from all genomes into a non-redundant set, and estimating the presence/absence or abundance of each gene in each genome. To determine whether bacterial and archaeal pangenomes could be constructed from reduced alphabet k-mers, we compared pangenomes estimated from genes against those estimated from k-mers (amino acid, dayhoff, and hydrophobic-polar). We compared pangenomes from 23 species belonging to 23 phyla in the GTDB taxonomy [23], with pangenome size ranging from 20-972 genomes (mean = 203 genomes, median = 44 genomes) (**Figure S 5**). For each pangenome, we compared the total number of genes to the total number of k-mers, and the number of unique genes to the number of distinct k-mers within each genome. We also tested the similarity of presence/absence profiles between pangenomes constructed with different methods using the Mantel test.

For these three metrics, performance varied minimally across encodings and k-mer sizes, but varied dramatically between different pangenomes: both k-mers and genes are highly correlated for some pangenomes and are not correlated for others (**Figure S 6**). We investigated pangenomes more closely to determine the source of the poor correlations and found that they were caused by the presence of many frameshifted proteins, one of many potential criteria for exclusion of GenBank genomes from RefSeq. For example, *Leptospira interrogans* had an R^2 of 0.12 between the total number of genes and k-mers in genomes in the pangenome, but 21 of 317 genomes contained frameshifted proteins. Removing these genomes increased the R^2 to 0.87 (**Figure 2 A**). This trend was consistent across pangenomes, where pangenomes with one or more frameshift-excluded genome had significantly lower R^2 values between total number of genes and k-mers per genome than pangenomes without (Welch Two Sample t-test, estimate = -0.36, p = 0.003) (**Figure 2 B**). Other RefSeq exclusion criteria did not impact the correlation between the total genes and k-mers per genome for a given pangenome.

Using pangenomes that contained no genomes excluded from RefSeq for containing many frameshifted proteins (n = 13), we found that k-mer size encoding had little impact on the accuracy of pangenome estimation with k-mers (**Figure 2 C**). This is likely because the genomes of the same species are closely related, so any reduced alphabet k-mer is sufficient to overcome minor genomic variations such as those introduced by codon degeneracy or evolutionary drift (CITE?). The one exception was for nucleotide k-mers (k = 31), which did not correlate as strongly with gene-based pangenomes. This supports the use of reduced alphabet k-mer encodings over nucleotide k-mers for construction of pangenomes. Given that neither encoding nor k-mer size impacted these performance metrics, we selected protein k-mers with k = 10 to complete the rest of our analysis as this combination was the only combination to fall among the top five performers across all three metrics. In addition, protein k-mers of length 10 have recently been shown to perform well for comparisons across variable taxonomic distances (CITE: TESSA).

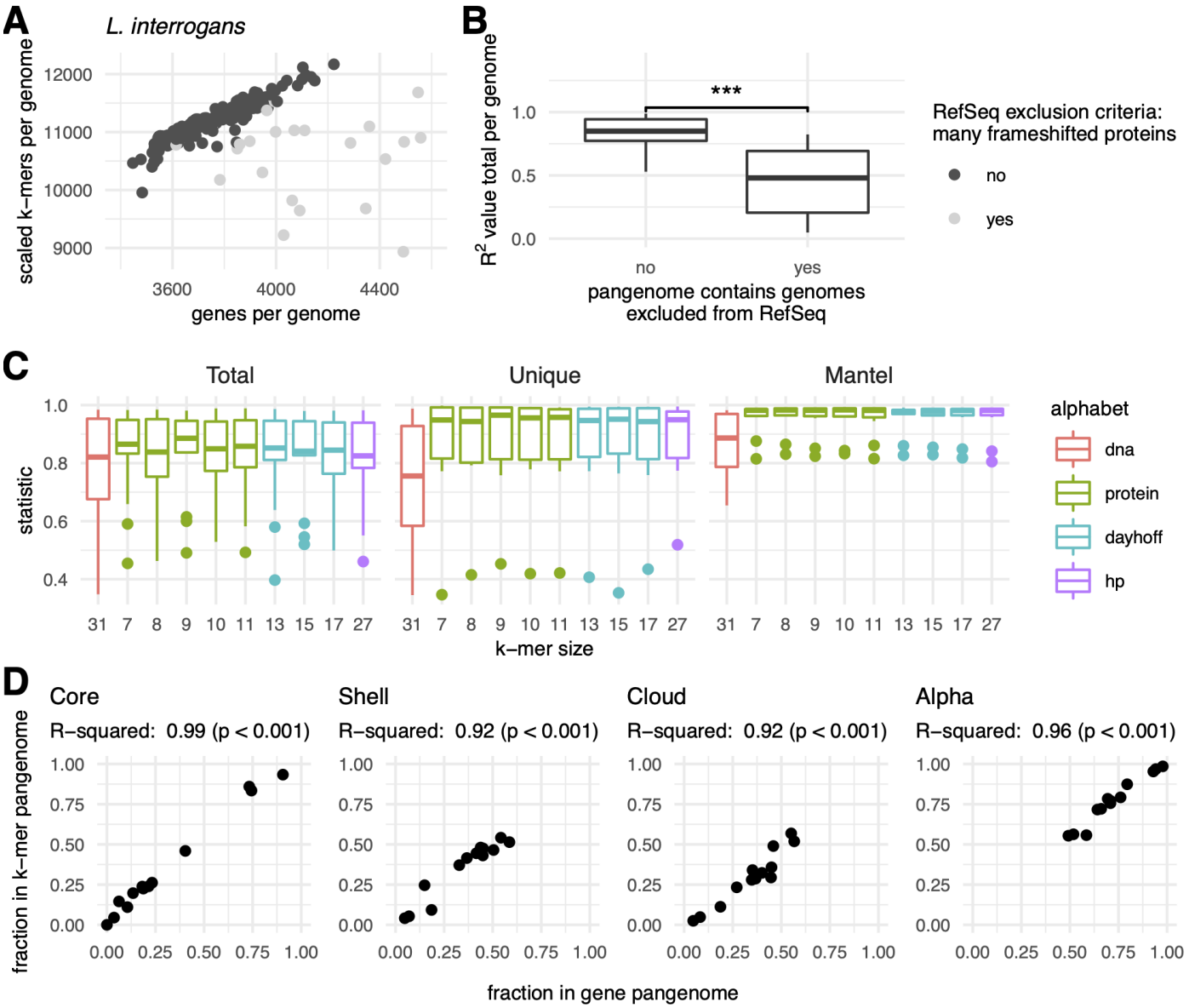

Figure 2: Amino acid k-mers accurately estimate microbial pangenomes. A, B) Genomes that are excluded from RefSeq for having many frameshifted proteins reduce similarity between gene- and k-mer-based pangenomes. **A)** Scatter plot of the total number of genes and k-mers per genome for the species *Leptospira interrogans*, where each point represents a single genome in the pangenome. Removing genomes flagged with RefSeq exclusion criteria "many frameshifted proteins" improves the correlation between these variables. **B)** Box plot of R^2 values between the total number of genes and k-mers per genome. Pangenomes that contain genomes with the RefSeq exclusion criteria of "many frameshifted proteins" have significantly lower R^2 values. **C)** Box plots representing the distribution of R^2 values for linear models (Total, Unique) or statistic values for mantel tests (Mantel) calculated for each pangenome. Only pangenomes that do not contain genomes with the RefSeq exclusion criteria "many frameshifted proteins" are plotted. K-mer size corresponds to the number of amino acid sequences used for the k-mer for all k-mers except $k = 31$, which corresponds to the number of nucleotides. *Total* corresponds to correlations between the total number of distinct genes and k-mers in a genome. *Unique* corresponds to correlations between the number of unique genes and k-mers in genome. *Mantel* corresponds to mantel tests between the gene and k-mer presence-absence matrices. **D)** Pangenome metrics strongly correlate between gene- and k-mer-based pangenomes. Pangenome categories core, shell, and cloud refer to genes or k-mers shared between the majority (>95%), some, or singleton genomes in the pangenome. Alpha is a value from Heaps law used to estimate whether a pangenome is open or closed.

We next investigated whether other pangenome metrics were well correlated between our k-mer-based and the gene-based method roary (see Methods for details). For these 13 pangenomes, the percent of k-mers or genes predicted to be part of the core, shell, or cloud pangenome was strongly correlated (**Figure** 2 D). The content of the core genomes was also similar between pangenomes built with different methods. Focusing on genes or k-mers shared between all genomes in a pangenome, and limiting our inquiry to pangenomes with at least five genes shared between all genomes (n = 11), we found that the core k-mers contained an average of 83.9% (SD = 15.4%) of sequences in the core

genes, while the core genes contained an average of 73.5% (SD = 16.9%) of the sequences in the core k-mers. This indicates congruence in the functional content represented by the core fractions of both pangenome types. Lastly, we compared whether pangenomes would be designated as open or closed by calculating the alpha value for the Heaps law model [24]. Alpha values were strongly correlated between gene- and k-mer based pangenomes (**Figure 2** D).

Taken together, these results show that reduced alphabet k-mers can accurately estimate key characteristics of pangenomes from bacterial and archaeal genomes.

K-mer methods accurately predict open reading frames in short sequencing reads

We next sought to determine whether open reading frames could be accurately predicted directly from short sequencing reads, as this would enable k-mer-based pangenome analysis without assembly. Without accurate open reading frame prediction, reads would need to be translated into all six translation frames prior to k-mer decomposition. This would inflate the number of k-mers and decrease similarity between genomes.

We evaluated whether orpheum, a tool recently developed to predict open reading frames in Eukaryotic short reads [22], could also perform this task in bacterial and archaeal sequences. Orpheum predicts open reading frames by comparing reduced alphabet k-mers in six frame translations of short sequencing reads against those in a database (Jaccard containment) and assigns an open reading frame as coding if containment exceeds a user-defined threshold [22]. To evaluate orpheum, we constructed a database containing all k-mers in coding domain sequences from genomes in GTDB rs202. Using representative genomes from the 23 species above, as well as 20 additional RefSeq genomes not in the GTDB rs202 database, we simulated short sequencing reads either from coding domain sequences or non-coding sequences and used these reads to test orpheum.

Using default parameters, orpheum accurately separated coding from non-coding reads when reads were simulated from genomes in GTDB (**Figure 3 A**). On average, 5.3% (SD = 2.8%) of reads that were coding were predicted to be non-coding, while 4.9% (SD = 1.5%) of reads that were non-coding were predicted to be coding. For reads simulated from genomes not in GTDB, orpheum recovered the majority of coding reads when genomes of the same species were in the database (**Figure 3 A,B**). On average, 30.2% (SD = 27.1%) of reads that were coding were predicted to be non-coding, while 4.8% (SD = 5.5%) of reads that were non-coding were predicted to be coding. Accuracy decreased with increasing taxonomic distance between the query genome and the closest relative in the database (**Figure 3 B**).

For genomes that had at least species-level representatives in GTDB, the largest source of error was non-coding reads being predicted as coding (**Figure 3 A**). We hypothesized that these reads originated from pseudogenes as these sequences would likely not be annotated as coding in the genomes from which the reads were simulated from, but may retain some k-mers contained in the database. To assess this hypothesis, we used annotation files produced by the NCBI Prokaryotic Genome Annotation Pipeline (PGAP), which annotates pseudogenes, for the 23 genomes for which these files were available [25,26]. On average, 12.4% (SD = 13.8%) of non-coding reads that were predicted to be coding fell within pseudogenes annotated by the PGAP pipeline. We then BLASTed a subset of the remaining non-coding reads that were predicted to be coding against the NCBI nr database. All reads we investigated had at least one match at 100% identity to protein sequences in the database, suggesting our test genomes contained additional pseudogenes not annotated by PGAP, or that the software we used to predict open reading frames missed some coding sequences (see Methods).

Because this method of open reading frame prediction cannot distinguish pseudogenes, it may not be appropriate for species with many pseudogenes.

Some coding sequences were also predicted to be non-coding. We hypothesized that this was caused by sequencing error introduced into the simulated reads. We mapped the simulated reads against the coding domain sequences from which they were derived and calculated mapping error rates. While all reads mapped, the error rate was higher for reads that were predicted to be non-coding than those predicted to be coding (Welch Two Sample t-test, estimate = 0.00523, $p < 0.001$).

Protein k-mers from predicted open reading frames in the simulated short sequencing reads recapitulated similarity between genomic coding domain sequences. We estimated the Jaccard similarity between genomes using protein k-mers ($k = 10$) from annotated coding domain sequences, and compared this against Jaccard similarity between genomes using protein k-mers from predicted open read frames in the simulated short sequencing reads. Genomes that were most similar in one matrix were also most similar in another matrix (Mantel statistic = 0.9975, $p < 0.001$). The average similarity among all pairwise comparisons for the coding domain sequences was 2.6%, and this decreased to 2.5% when using the open reading frames predicted from reads. This demonstrates that information recovered from open reading frame prediction from short read is similar to that derived directly from the genome sequence.

The majority of predictive capability originated from species-level databases. We performed ORF prediction using just species-level databases for genomes that had at least a species-level representative in GTDB, and compared this against ORF prediction using the full GTDB database. On average, there was no change between the percent of reads derived from coding domain sequences when a species-level database was used versus when all of GTDB was used to predict open reading frames (**Figure 7**).

Decreasing the Jaccard containment threshold increased the sensitivity and specificity of ORF prediction when there are no closely related genomes in the database (**Figure 3 C, Table 1**). The Jaccard containment threshold controls the final prediction of coding vs. non-coding, as well as the the number of open reading frames which a read is translated into. On average, increasing the rank of the closest taxonomic relative in the database by one taxonomic level decreased the optimal Jaccard containment threshold by 0.13.

Table 1: Jaccard containment thresholds that maximize the Youden's index depending on the taxonomic rank of the closest relative in GTDB.

Jaccard threshold	closest rank	mean sensitivity	mean specificity	mean Youden's index
0.47	genome	0.988	0.971	0.959
0.39	species	0.941	0.961	0.902
0.17	genus	0.790	0.862	0.653
0.07	family	0.593	0.878	0.471

Overall, these results show that open reading frames can be accurately determined from short sequencing reads when closely related proteomes are available.

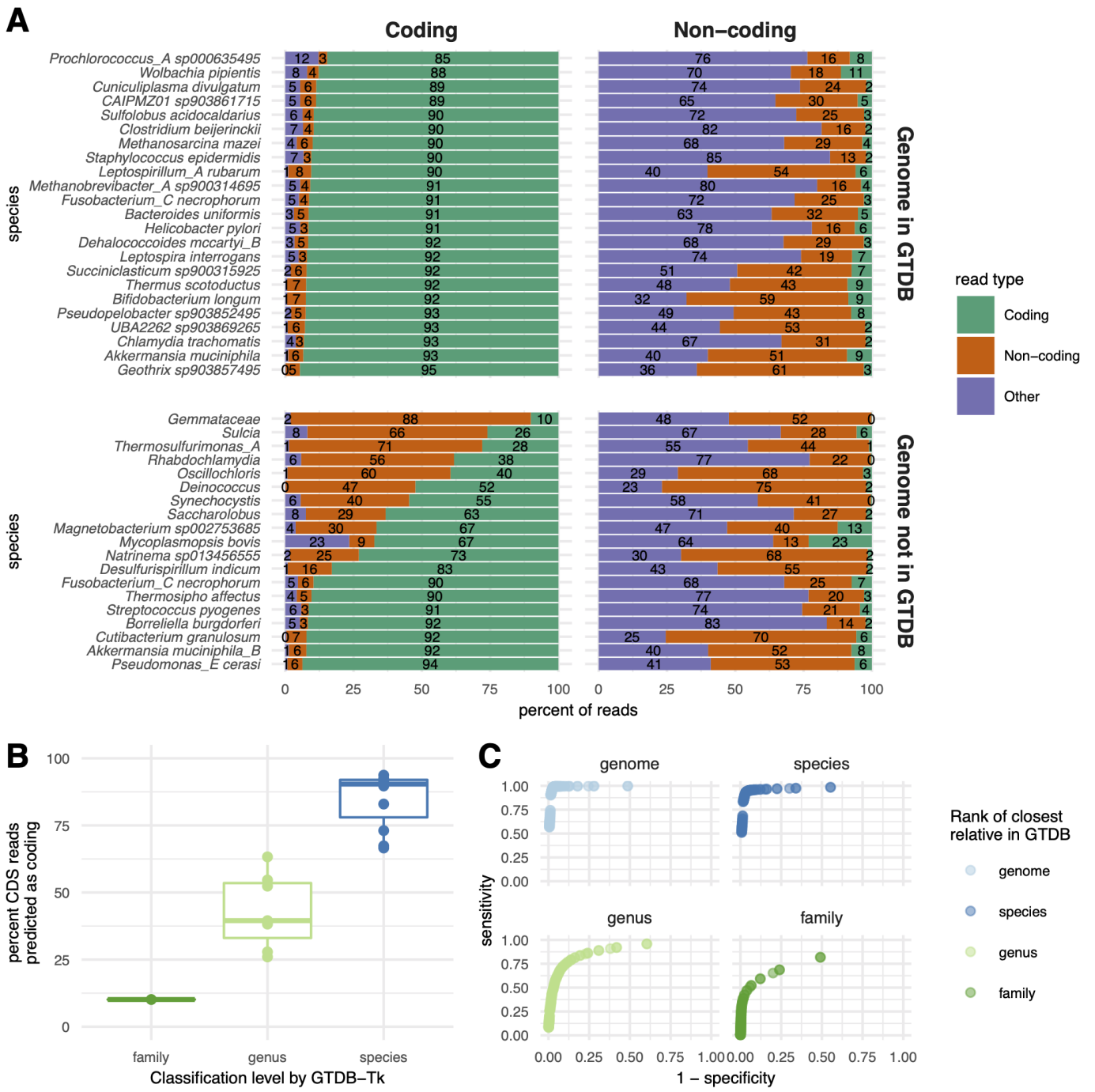

Figure 3: Orpheum correctly assigned short sequencing reads as coding or non-coding and selects the correct open reading frame. A) Percent of simulated coding or non-coding sequences predicted as coding, non-coding, or discarded based on quality metrics (see methods). Genomes are split by those in GTDB and those not in GTDB. Genomes not in GTDB are labelled by taxonomic assignment from GTDB-tk. Predictions were made using default parameters (Jaccard containment = 0.5). **B)** Boxplots of the percent of coding reads that were recovered by Orpheum, separated by the level of taxonomic assignment achieved by GTDB-Tk. Orpheum recovers more coding sequences when there are closely related genomes in the database. **C)** Receiver operating curves for the Jaccard containment thresholds. Curves are separated by the level of taxonomic assignment achieved by GTDB-Tk, and values are averaged across all genomes that fell within those categories. The best Jaccard threshold decreases when there are fewer closely related genomes in the database.

K-mer-based metapangenomics combined with assembly graphs reveals strain dynamics

Given that amino acid k-mers accurately estimated pangenomes, and that the correct open reading frame could be predicted reliably from short sequencing data, we next combined these approaches to

perform metapangenome analysis from short read shotgun metagenomes. We used 12 metagenomes from a single individual sampled over the course of a year by the Integrated Human Microbiome Project (iHMP) [27]. The individual was diagnosed with Crohn's disease, a sub type of inflammatory bowel disease characterized by inflammation along the gastrointestinal tract. The individual received three courses of antibiotics over the year and each course was separated by weeks without antibiotics (**Figure 4 A**).

We estimated the metapangenome for each species that was detected in all 12 metagenomes and that accounted for at least 2% of reads across metagenomes, for a total of six metapangenomes (**Figure 4 A**). To obtain all sequencing reads that originated from genomes of these species, we performed assembly graph genome queries [18]. Assembly graphs contain all sequences in a metagenome, and assembly graph queries return sequences in the metagenome that are either in the query or nearby to the query in the graph. Assembly graph genome queries return sequencing reads that originate from genomes in the metagenome that have as little as 0.1 Jaccard similarity (approximately 93% average nucleotide identity (ANI) (CITE: TESSA)) to the query genome [18]. After retrieving reads in this way, we predicted open reading frames using orpheum. We used species-level databases as these were successful in the context of isolate genomes not in the database (see above) and because they would be more likely to filter out reads beyond the species boundary (95% ANI [28]) that were returned by assembly graph queries. Using the predicted amino acid sequences, we built k_{aa}-mer metapangenomes for each of the six species (**Figure 1 C, 8, Table 3**).

We compared these metapangenomes against reference pangenomes built using genomes of the same species in GTDB and against *de novo* metapangenomes built from MAGs of the same species that were assembled and binned from these samples (see Methods). Almost all sequences from the reference pangenome occurred within the k_{aa}-mer metapangenomes (**Figure 4 B**), indicating we recovered the majority of sequencing variation contained within the reference pangenome. Further, a large fraction of sequences were shared between the *de novo* metapangenome and the k_{aa}-mer metapangenome (**Figure 4 B**), indicating we also recover the majority of variation captured by assembly and binning.

A large fraction of k-mers were only represented in the k_{aa}-mer metapangenome (**Figure 4 B**). To determine whether these sequences represented true biological variation from our query species and not contamination from other species, we next iteratively mapped the reads that were used to build the k_{aa}-mer metapangenome against the reference pangenome and the *de novo* metapangenome (**Figure 4 C**). The majority of reads mapped against the reference pangenomes (mean = 92.2%, SD = 12.4%), a few of the unmapped reads mapped against the *de novo* metapangenome (mean = 0.8%, SD = 1.3%), and 7.1% (SD = 12.3%) of reads did not map. We repeated this process a second time but mapped in amino acid space. Mapping in amino acid space improves sensitivity over nucleotide mapping [29]. The fraction of reads that mapped increased by an average of 1.7% (Welch Two Sample Paired t-test, estimate = 1.9, $p < 0.001$), accounting for 94.6% of total reads and indicating that a substantial fraction of distinct sequences in the k_{aa}-mer metapangenome represent diverged sequences with similar amino acid sequences.

On average, 5.4% (SD = 12%) of reads from the k_{aa}-mer metapangenome were unaccounted for after mapping against other (meta)pangenomes. We assembled and annotated these reads, and BLASTed the 88 resultant protein sequences against the NCBI nr database. Of 56 predicted genes with a BLAST hit, 76.8% matched sequences from the same species or closest specified lineage level as the top hit. This suggests our method recovers functional sequences even if those sequences are not in MAGs or in reference databases (in this case, in NCBI nr but not GTDB).

Visualizing the k_{aa}-mer metapangenomes alongside sequencing depth information, we observed dynamics in the presence of species (**Figure 8 A**) or strains (**Figure 8 B**) in response to antibiotic administration. The fluctuation of the presence of species indicates that antibiotic administration

impacted the community structure of the gut microbiome, as is expected [30]. This is exemplified by periodic blooms of *Enterocloster bolteae*, an organism associated with disturbance succession [31].

Similarly, we detected changes in accessory k_{aa}-mers, the majority of which appeared or disappeared on or after the start of metronidazole administration at week 13 (**Figure 8 B**). Metronidazole targets anaerobic bacteria via reduction by pyruvate:ferredoxin oxidoreductase system which creates an electron sink that produces free radicals that are toxic to cells [32]. Metronidazole treatment disproportionately impacts the presence of anaerobes in the gut microbiome [33]. We hypothesized that fluctuations in accessory k_{aa}-mers reflected strain-level turn over in the community. To confirm this, we compared the nucleotide k-mer content in each query neighborhood against the GTDB database to determine which strains were present (**Figure 4 D**). We used a k-mer size of 51, as this is indicative of strain-level similarity (CITE: metapalette, tessa). In each of the three species we investigated, we identified different patterns of strain fluctuations. In *Bacteroides uniformis*, only one strain (BIOML-A27) was present until week 25, when another strain appears (BIOML-A2). In *Parabacteroides distasonis*, the dominant strain switches at week 13 (20_3 to OF01-14), coincident with other strains appearing (BIOML-A21, BIOML-A26, BIOML-A32). Lastly, in *Phocaeicola vulgatus*, the dominant strain does not change through the time course, but multiple other strains are detected with one strain switch occurring at week 13 (H23 and AM38-19 to VPI-4496.2).

Taken together, these results demonstrate that k_{aa}-mer metapangenomes built from metagenome assembly graph neighborhoods can capture species and strain dynamics in microbiome communities.

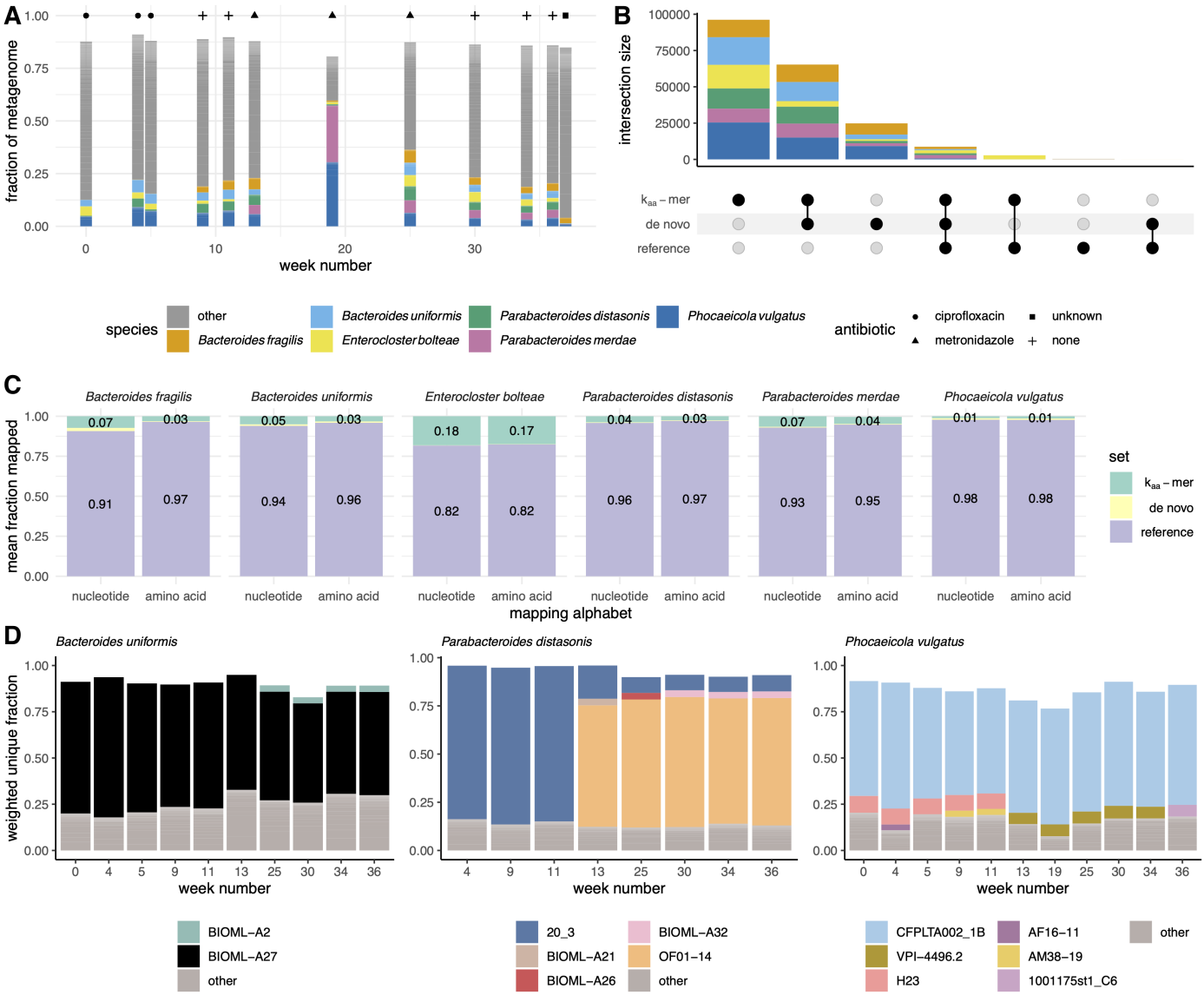

Figure 4: K_{aa}-mer metapangenomes reveal species and strain dynamics in time series gut microbiome metagenomes after antibiotic exposure. A) Antibiotic courses and corresponding gut microbiome profiles for a single individual with Crohn's disease. Fractional abundances are colored by species, with only the six species that accounted for greater than 2% of all metagenome reads displayed. **B)** Upset plot of amino acid k-mers (k = 10) present in the k_{aa}-mer metapangenomes, the *de novo* metapangenomes, and the reference pangenome. Intersections are colored by species. **C)** Bar plots indicating the average fraction of reads used to build the k_{aa}-mer metapangenome that mapped first against the reference pangenome, then against the *de novo* metapangenome, or were unmapped. More reads mapped in amino acid space than in nucleotide space. Only the fraction of reads that mapped to the reference pangenome and the fraction reads that were unmapped are labelled. **D)** Bar plots of the fraction of k_{aa}-mer metapangenome sequences that were anchored to a given strain using the sourmash gather algorithm against the GTDB rs202 database (k = 51). Colors represent strains, which are labelled by their NCBI strain name. Missing fractions depicted as blank space between the bar and one represent novel k-mers not in the database. Only genomes that accounted for greater than 2% of the weighted fractional abundance and that were annotated as the same species are colored. Weeks in which the species was low-abundance are excluded. Starting at at week 13, sequences from previously unobserved strains were detected within each metapangenome. This timing coincides with metronidazole administration.

Discussion

We present a method to perform assembly-free metapangenomics that is minimally reliant on reference databases. We show that pangenome metrics like core, cloud, and shell pangenome fractions can be accurately estimated with long amino acid k-mers. We then demonstrate accurate prediction of open reading frames in highly accurate short sequencing reads by comparing amino acid k-mers in all translation frames against a database of k-mers from all known bacterial and archaeal genomes in GTDB (rs202). Combining these tools enables pangenome estimation directly from quality controlled short sequencing reads. In the context of metagenomes, these approaches enable metapangenome estimation without the need to *de novo* assemble and bin sequences, eliminating common sources of lost sequencing variation [18]. These techniques also reduce the dependence of metapangenomics on complete or comprehensive reference databases, which can be important for understudied environments.

While we leverage open reading frame prediction from short reads and k_{aa}-mer pangenomes in the context of metapangenomes, these approaches have additional applications. Open reading frame prediction with orpheum can be executed on microbial Illumina short read data sets. This may improve functional recall from metagenome short reads from organisms without reference genomes and from communities that have low assembly rates [34], or from metatranscriptomes, however these applications must be validated. Similarly, k_{aa}-mer pangenomes built from sketches may offer benefits over other pangenome construction methods, many of which are predominately tied to exact matching of k-mers between genomes. Exact matching of k-mers between genomes enables additional genomes to be added to the pangenome without having to re-cluster gene sequences. Exact matching also allows direct comparisons to distantly related organisms, which creates a unified framework for genome comparisons even when organisms are distantly related. While we predominately explored k_{aa}-mers of length 10 in this paper, k-mers from other degenerate alphabets like Dayhoff or hydrophobic-polar encodings may allow for accurate comparisons at greater evolutionary distances (CITE: Tessa). When pangenomes are built from FracMinHashes as performed here, the scaled down sampling parameter may also enable even faster pangenome size estimation even for very large collections of genomes. genes in a genome, and could be potentially used a quality control metric for annotated genomes. While developed for the metapangenomics space, this study demonstrates that k_{aa}-mer pangenomes also accurately estimate pangenomes built from isolate genomes. Given that building k_{aa}-mer sketches and exact matching of k_{aa}-mers between genomes is fast, this provides an alternative approach for building pangenomes. Lastly, our results suggest that the number of k_{aa}-mers in a genome strongly correlates with the number of genes; we observed that while the number of genes per genome is increased for genomes with this exclusion criteria, there is no commensurate increase in the number of k_{aa}-mers observed. This suggests that the number of

k_{aa}-mers in a genome could be used to predict the expected range of predicted genes, and could be used as a quality diagnostic criteria.

We posit that the combination of these approaches is potentially most useful in the context of analyzing metagenome assembly graphs. Assembly graphs like compact de Bruijn graphs (cDBG) capture all sequences in a metagenome, including sequences with high strain variation or low coverage, which may not be captured by other analysis methods. A targeted query of an assembly graph, for example with a metagenome-assembled genome bin, can recover all sequencing reads in a metagenome that originate from all genomes of the same species [18]. Often, reads may be longer than unitigs (nodes) in the cDBG in regions of high variation making read retrieval desirable. While recovering these reads and assigning their taxonomic identity through graph queries is useful, many of the recovered reads cannot be assembled due to prolific sequencing variation attributable to strain diversity in the original microbial community. Yet, the sequences represented by these un-assembleable reads often encode functional potential, some of which may be key to a microorganisms functioning within its ecosystem [17,35]. The approaches presented in this paper enable these sequences to be represented in metapangenome estimation. Indeed, as demonstrated by the number of bins generated per species at each time point, many species were observed in a sample for which no bin was produced, often because the species was present at low abundance (**Figure [metap_sfig?]**). K_{aa}-mer metapangenomes rescue these sequences and include them in the analysis.

Interestingly, in some samples, *de novo* metagenome analysis produced multiple MAGs (**Figure [metap_sfig?]**). While not explored here, it may be possible to estimate the number of strains present in a sample using k_{aa}-mer abundances. FracMinHash sketches optionally retain abundance information, so it is possible that this information could be retrieved directly from these sketches.

This method is not without shortcomings, the largest of which is the current lack of integration of k_{aa}-mers with functional annotations. We see three avenues by which this could be ameliorated. First, the underlying cDBG used to produce assembly graph query neighborhoods could be annotated, and these annotations could be paired with k_{aa}-mers. The technology to do this exists (CITE: IBD) [36], however this approach is the least scalable option. Second, using the sequences included in the orpheum species databases, species-level protein databases that include functional annotations could be built and searched using sourmash gather. While rapid, this approach is undesirable given its reliance on existing annotations for genes within a given species. Lastly, a database such as the NCBI nr database could be made searchable in a similar capacity, increasing the breadth of possible matches. This may require new algorithm development to maintain scalability.

While cDBGs are often common intermediary data structures in bioinformatics pipelines (e.g., metagenome assembly), the organization of cDBGs is still under explored. Sequences from a genome or from closely related genomes often co-locate in neighborhoods of the cDBG [17,18], but regions of high sequence conservation and horizontally transferred genes create bridges between otherwise distant neighborhoods in the graph [17]. Even still, species-level neighborhoods are well conserved and can be extracted [[18]; 10.1101/2022.02.25.481923]. As we observed in strain-level plots of query neighborhoods wherein a fraction of the graph fell in the “other category” and belonged to low-abundance strains of the same species or organisms of other closely-related species (**Figure 4 D**), it is unclear the extent to which assembly graph organization breaks down in real metagenomes. Prolific horizontal gene transfer could account for shared sequencing content, either by collapsing neighborhoods in the graph or by increasing the fraction of shared sequences between community members. Deeply sequenced paired long and short reads from the same community could help resolve these questions.

While we observe strain dynamics, the ecological reason for those signals cannot be determined from short read sequences alone; the patterns we observe could be the result of strain turn over, the

presence of multiple strains, horizontal gene transfer between strains, or some combination of all of these influences. Again, long read sequencing, or sequencing from isolate genomes, could resolve these questions, particularly as long read lineage-resolved methods become mainstream [37]. In principle, the methods presented here should extend directly to long read sequences, however this remains a point of future research.

Methods

All code is available at github.com/taylorreiter/2021-panmers (results section 1), github.com/taylorreiter/2021-orpheum-sim (results section 2), and https://github.com/taylorreiter/2021-metapangenome-example (results section 3).

Selection of benchmarking species for pangenome analysis

We selected a species representative for each of the 23 phyla in GTDB rs202 [23]. To select representative species, we first filtered species with fewer than 20 representatives and greater than 1000 representatives. While this approach scales beyond 1000 genomes, we elected to benchmark smaller sets to iterate over the potential parameter space more quickly. Of species remaining after filtering, we selected the species within each phyla that had the largest number of genomes. We downloaded these genomes from GenBank. Species names are recorded in **Figure 5**.

Calculating the gene-based pangenome with roary

To calculate the gene-based pangenome, we first annotated each genome using prokka [38]. We then used the resulting GFF annotations files to calculate the pangenome with roary using default settings [39].

Calculating the k-mer based pangenome with sourmash

To calculate k-mer based pangenomes, we used sourmash `sketch` to generate signatures from the bakta-predicted amino acid sequences (`.faa` files) [40]. We used the protein alphabet (k = 7, 8, 9, 10, 11), dayhoff alphabet (k = 13, 15, 17), and the hydrophobic-polar alphabet (k = 27, 31). All signatures were calculated with a scaled value of 100. The scaled parameter controls the fraction of the total k-mers represented by the sketch; a scaled value of 100 indicates that 1/100th of the distinct k-mers in a genome were included in each sketch. We converted signatures from json format into a genome x hash presence-absence matrix.

Correlating gene-based and k-mer based pangenomes

Using the presence-absence matrices for the gene-based and k-mer-based pangenomes, we correlated total genes/k-mers observed per genome and total unique genes/k-mers observed per genome for each species. We used the `rowSums()` function in R to determine the number of genes/unique genes per matrix, then used the `lm()` function with default parameters to correlate the values. We also used the Mantel test to determine whether genomes that were most similar in the gene presence-absence matrix were also most similar in the k-mer presence-absence matrix. We used the `mantel()` function in the R vegan package to perform this test [41]. We used distance matrices calculated with the `dist()` function using the parameter `method = "binary"` as input to the mantel test.

Generating standard pangenome metrics with pagoo

The pagoo R package provides functions to analyze bacterial pangenomes [42]. We used this package to generate standard pangenome metrics and visualizations. These metrics are based on the presence-absence matrices generated above and include calculation of the core, shell, and cloud genome sizes and estimation of the alpha value in Heaps law for estimation of pangenome openness.

Augmenting benchmarking species set to include genomes not in GTDB for open reading frame prediction

We next generated a benchmarking data set for open reading frame prediction. We selected a genome from each of the 23 species evaluated above, choosing the GTDB rs202 representative genome for each species. Genome accessions are recorded in **Table 4**. Given that open reading frame prediction relies on a database, and we used k-mers in GTDB rs202 to generate this database, we also wanted to select genomes that were not in GTDB to evaluate this method. We determined the bacterial and archaeal genomes that were added to RefSeq after the construction of GTDB rs202 (April 2021-November 2021). From this set, we selected a representative genome from each of the distinct NCBI phyla represented among these genomes, 20 in total. Genome accessions are recorded in **Table 5**. We then ran GTDB-tk on these genomes to predict the GTDB taxonomy of each.

Simulating coding domain sequence and non coding domain sequence reads with polyester

We next created a labelled data set of simulated reads that were generated from either coding domain sequences (CDS) or non-coding regions within each genome. We annotated the genomes with bakta to produce CDS ranges [43], and used polyester to simulate reads from CDS or non-coding regions [44]. We used the default short read error profile within polyester.

Determining short read open reading frames with orpheum

We used the orpheum tool to predict open reading frames from simulated short reads [22]. Orpheum was developed to predict open reading frames in short RNA-seq reads from Eukaryotic organisms without a reference genome or transcriptome sequence [22]. Orpheum perform six-frame translation on nucleotide sequencing reads, calculates k-mers in an amino acid, dayhoff, or hydrophobic-polar encoding at the designated k-mer length, and then estimates the Jaccard similarity between k-mers in each translation frame and a database. It then selects all open reading frames based on a Jaccard similarity threshold, and returns those reads as translated amino acid sequences. Open reading frames are excluded if they contain stop codons, low complexity sequences, or if the read is too short to perform translation. Reads are designated as non-coding if they don't reach the Jaccard similarity threshold and are not excluded for other reasons.

We constructed a database from GTDB rs202 using sourmash XXX and using a k-mer size of 10. + [**Tessa?**] any relevant details would be very helpful :)

K_{aa}-mer metapangenome analysis of iHMP metagenomes

We used sourmash, spacegraphcats, and orpheum to perform k_{aa}-mer metapangenome analysis of 12 iHMP time series gut microbiomes captured by short read shotgun metagenomes [45]. We downloaded samples HSM6XRQB, HSM6XRQI, HSM6XRQK, HSM6XRQM, HSM6XRQO, HSM67VF9, HSM67VFD, HSM67VFJ, HSM7CYY7, HSM7CYYD, HSM7CYY9, HSM7CYYB from ibdmdb.org. We adapter and quality trimmed each sample with fastp (parameters `--detect_adapter_for_pe`, `--qualified_quality_phred 4`, `--length_required 31`, and `--correction`), removed human host sequencing reads with bbduk (parameters `k=31`, reference file

https://drive.google.com/file/d/0B3llHR93L14wd0pSSnFULUlhcUk/edit?usp=sharing), and k-mer trimmed reads using khmer `trim-low-abund.py` (parameters `-C 3`, `-Z 18`, `-V`) [46]. We then used sourmash gather to infer the taxonomic profile of each sample, using the GTDB rs202 database (`k` = 31, https://osf.io/w4bcm/) [45]. We summarized the results to species-level using the GTDB taxonomy. We retained species with a cumulative sum of at least 2% (sum of `f_unique_to_query`) across metagenome reads as query genomes. We downloaded each genome from GenBank (**Table 2**) and performed spacegraphcats assembly graph queries with each (parameters `ksize: 31`, `radius: 1`, `paired_reads: true`) [18]. Using the returned reads, we predicted open reading frames using orpheum `translate` (parameters `--jaccard-threshold 0.39`, `--alphabet protein`, `--peptide-ksize 10`) and using species-level GTDB databases. We sketched each set of translated reads using sourmash `sketch` (parameters `protein`, `-p k=10,scaled=100,protein`) [40], converted each sketch to a csv file, and then combined csv files for a single query species across all metagenomes. This long format csv was used as input for the R pangenome package pagoo, using the `pagoo()` function [42]. We used pagoo methods `pg$gg_binmap()`, `pg$summary_stats()`, and `pg$pg_power_law_fit()` to visualize the pangenome, calculate the size of the core, shell, and cloud, and estimate alpha.

Table 2: Query genome GTDB species names and GenBank accessions.

species	accession
Parabacteroides distasonis	GCA_000162535.1
Enterocloster bolteae	GCF_003433765.1
Bacteroides fragilis	GCF_003458955.1
Parabacteroides merdae	GCF_003475305.1
Bacteroides uniformis	GCF_009020325.1
Phocaeicola vulgatus	GCF_009025805.1

Comparing k_{aa}-mer, *de novo*, and reference (meta)pangenomes

We compared the k_{aa}-mer metapangenomes against other (meta)pangenomes. We first constructed a reference pangenome for each query species as detailed in the methods section, "Calculated the gene-based pangenome with roary." We used all genomes in the GTDB rs202 database for a given species (**Table 2**). We then constructed *de novo* metapangenomes for each species. Using quality controlled reads from each sample, we assembled each metagenome separately using megahit with default parameters [47]. We binned the resultant assemblies using metabat2 with parameter `-m 1500` [48]. We assigned GTDB species to each bin using sourmash `gather` (DNA, *k* = 31, scaled = 2000) against the GTDB rs202 database, selecting the species of the best match [16]. We decontaminated each bin using charcoal using default parameters (CITE: charcoal). We annotated each bin using prokka [38]. To compare the sequence content of the (meta)pangenomes, we sketched the protein sequences from the *de novo* and reference (meta)pangenomes using sourmash `sketch` (protein, *k* = 10, scaled = 100). We intersected the hashes in these sketches to assess shared sequencing content, and visualized it using the R complexUpset package [49].

To further compare the sequence content of the k_{aa}-mer metapangenomes to the *de novo* and reference (meta)pangenomes, we mapped the reads used to build the k_{aa}-mer metapangenomes iteratively against the (meta)pangenomes. We first mapped the reads against the reference pangenome using bwa `mem` with default parameters [50]. We used samtools `stat` to determine read mapping statistics, and samtools `view` and `fastq` to extract the unmapped reads [51]. We then mapped the unmapped reads against the *de novo* metapangenome. To prepare the *de novo*

metapangenome for mapping, we clustered nucleotide sequences at 95% using cd-hit-est [52]. We then mapped the unmapped reads using bwa `mem` with default parameters. We extracted unmapped reads using the same procedure as above. To determine whether more reads mapped in amino acid space, we translated the reference and *de novo* (meta)pangenomes into protein sequences using transeq (http://ebi.ac.uk/Tools/emboss/transeq/index.html), and then repeated the same iterative mapping procedure using paladin `align` [29]. To test whether the fraction of mapped reads increased between the two mapping protocols, we used the R function `t.test()` using parameter `paired = TRUE`.

References

1. **A new view of the tree of life**
Laura A Hug, Brett J Baker, Karthik Anantharaman, Christopher T Brown, Alexander J Probst, Cindy J Castelle, Cristina N Butterfield, Alex W Hernsdorf, Yuki Amano, Kotaro Ise, ... Jillian F Banfield
Nature Microbiology (2016-05) https://doi.org/bpkh
DOI: 10.1038/nmicrobiol.2016.48 · PMID: 27572647

2. **A genomic catalog of Earth's microbiomes**
Stephen Nayfach, Simon Roux, Rekha Seshadri, Daniel Udwary, Neha Varghese, Frederik Schulz, Dongying Wu, David Paez-Espino, I-Min Chen, Marcel Huntemann, ... Emiley A Eloe-Fadrosh
Nature Biotechnology (2021-04) https://doi.org/ghjh4b
DOI: 10.1038/s41587-020-0718-6 · PMID: 33169036 · PMCID: PMC8041624

3. **Phage diversity, genomics and phylogeny**
Moïra B Dion, Frank Oechslin, Sylvain Moineau
Nature Reviews Microbiology (2020-03) https://doi.org/ggkq9f
DOI: 10.1038/s41579-019-0311-5 · PMID: 32015529

4. **Community structure and metabolism through reconstruction of microbial genomes from the environment**
Gene W Tyson, Jarrod Chapman, Philip Hugenholtz, Eric E Allen, Rachna J Ram, Paul M Richardson, Victor V Solovyev, Edward M Rubin, Daniel S Rokhsar, Jillian F Banfield
Nature (2004-03) https://doi.org/b85j5j
DOI: 10.1038/nature02340 · PMID: 14961025

5. **Linking pangenomes and metagenomes: the Prochlorococcus metapangenome**
Tom O Delmont, AMurat Eren
PeerJ (2018-01-25) https://doi.org/gczf4x
DOI: 10.7717/peerj.4320 · PMID: 29423345 · PMCID: PMC5804319

6. **Global ecotypes in the ubiquitous marine clade SAR86**
Adrienne Hoarfrost, Stephen Nayfach, Joshua Ladau, Shibu Yooseph, Carol Arnosti, Chris L Dupont, Katherine S Pollard
The ISME Journal (2020-01) https://doi.org/gns4sb
DOI: 10.1038/s41396-019-0516-7 · PMID: 31611653 · PMCID: PMC6908720

7. **Meta-Pangenome: At the Crossroad of Pangenomics and Metagenomics**
Bing Ma, Michael France, Jacques Ravel
The Pangenome (2020) https://doi.org/gns4r8
DOI: 10.1007/978-3-030-38281-0_9 · PMID: 32633911 · ISBN: 9783030382803

8. **Genome analysis of multiple pathogenic isolates of Streptococcus agalactiae : Implications for the microbial "pan-genome"**
Hervé Tettelin, Vega Masignani, Michael J Cieslewicz, Claudio Donati, Duccio Medini, Naomi L Ward, Samuel V Angiuoli, Jonathan Crabtree, Amanda L Jones, AScott Durkin, ... Claire M Fraser
Proceedings of the National Academy of Sciences (2005-09-27) https://doi.org/b9b426
DOI: 10.1073/pnas.0506758102 · PMID: 16172379 · PMCID: PMC1216834

9. **Toward quantifying the adaptive role of bacterial pangenomes during environmental perturbations**
Roth E Conrad, Tomeu Viver, Juan F Gago, Janet K Hatt, Stephanus N Venter, Ramon Rossello-Mora, Konstantinos T Konstantinidis

The ISME Journal (2021-12-09) https://doi.org/gpdvs9
DOI: 10.1038/s41396-021-01149-9 · PMID: 34887548

10. **Ten years of pan-genome analyses**
George Vernikos, Duccio Medini, David R Riley, Hervé Tettelin
Current Opinion in Microbiology (2015-02) https://doi.org/f63bmf
DOI: 10.1016/j.mib.2014.11.016 · PMID: 25483351

11. **Multiple levels of the unknown in microbiome research**
Andrew Maltez Thomas, Nicola Segata
BMC Biology (2019-12) https://doi.org/gnm4t7
DOI: 10.1186/s12915-019-0667-z · PMID: 31189463 · PMCID: PMC6560723

12. **Metagenome-assembled genome binning methods with short reads disproportionately fail for plasmids and genomic Islands**
Finlay Maguire, Baofeng Jia, Kristen L Gray, Wing Yin Venus Lau, Robert G Beiko, Fiona SL Brinkman
Microbial Genomics (2020-10-01) https://doi.org/gns4sc
DOI: 10.1099/mgen.0.000436 · PMID: 33001022 · PMCID: PMC7660262

13. **Toward quantifying the adaptive role of bacterial pangenomes during environmental perturbations**
Roth E Conrad, Tomeu Viver, Juan F Gago, Janet K Hatt, Fanus Venter, Ramon Rosselló-Móra, Konstantinos T Konstantinidis
Microbiology (2021-03-15) https://doi.org/gns4sd
DOI: 10.1101/2021.03.15.435471

14. **MetaPalette: a k -mer Painting Approach for Metagenomic Taxonomic Profiling and Quantification of Novel Strain Variation**
David Koslicki, Daniel Falush
mSystems (2016-06-28) https://doi.org/gg3gbd
DOI: 10.1128/msystems.00020-16 · PMID: 27822531 · PMCID: PMC5069763

15. **Mash: fast genome and metagenome distance estimation using MinHash**
Brian D Ondov, Todd J Treangen, Páll Melsted, Adam B Mallonee, Nicholas H Bergman, Sergey Koren, Adam M Phillippy
Genome Biology (2016-12) https://doi.org/gfx74q
DOI: 10.1186/s13059-016-0997-x · PMID: 27323842 · PMCID: PMC4915045

16. **Lightweight compositional analysis of metagenomes with FracMinHash and minimum metagenome covers**
Luiz Irber, Phillip T Brooks, Taylor Reiter, NTessa Pierce-Ward, Mahmudur Rahman Hera, David Koslicki, CTitus Brown
Bioinformatics (2022-01-12) https://doi.org/gn34zt
DOI: 10.1101/2022.01.11.475838

17. **Genome-resolved metagenomics identifies genetic mobility, metabolic interactions, and unexpected diversity in perchlorate-reducing communities**
Tyler P Barnum, Israel A Figueroa, Charlotte I Carlström, Lauren N Lucas, Anna L Engelbrektson, John D Coates
The ISME Journal (2018-06) https://doi.org/gdms93
DOI: 10.1038/s41396-018-0081-5 · PMID: 29476141 · PMCID: PMC5955982

18. **Exploring neighborhoods in large metagenome assembly graphs using spacegraphcats reveals hidden sequence diversity**

CTitus Brown, Dominik Moritz, Michael P O'Brien, Felix Reidl, Taylor Reiter, Blair D Sullivan
Genome Biology (2020-12) https://doi.org/d4bb
DOI: 10.1186/s13059-020-02066-4 · PMID: 32631445 · PMCID: PMC7336657

19. **Critical Assessment of Metagenome Interpretation—a benchmark of metagenomics software**
Alexander Sczyrba, Peter Hofmann, Peter Belmann, David Koslicki, Stefan Janssen, Johannes Dröge, Ivan Gregor, Stephan Majda, Jessika Fiedler, Eik Dahms, … Alice C McHardy
Nature Methods (2017-11) https://doi.org/gbzspt
DOI: 10.1038/nmeth.4458 · PMID: 28967888 · PMCID: PMC5903868

20. **Critical Assessment of Metagenome Interpretation - the second round of challenges**
F Meyer, A Fritz, Z-L Deng, D Koslicki, A Gurevich, G Robertson, M Alser, D Antipov, F Beghini, D Bertrand, … AC McHardy
Bioinformatics (2021-07-12) https://doi.org/gk566x
DOI: 10.1101/2021.07.12.451567

21. **Generation of lineage-resolved complete metagenome-assembled genomes by precision phasing**
Derek M Bickhart, Mikhail Kolmogorov, Elizabeth Tseng, Daniel M Portik, Anton Korobeynikov, Ivan Tolstoganov, Gherman Uritskiy, Ivan Liachko, Shawn T Sullivan, Sung Bong Shin, … Timothy PL Smith
Microbiology (2021-05-04) https://doi.org/gns4sf
DOI: 10.1101/2021.05.04.442591

22. **Single-cell transcriptomics for the 99.9% of species without reference genomes**
Olga Borisovna Botvinnik, Venkata Naga Pranathi Vemuri, NTessa Pierce, Phoenix Aja Logan, Saba Nafees, Lekha Karanam, Kyle Joseph Travaglini, Camille Sophie Ezran, Lili Ren, Yanyi Juang, … CTitus Brown
Bioinformatics (2021-07-10) https://doi.org/gns4sg
DOI: 10.1101/2021.07.09.450799

23. **GTDB: an ongoing census of bacterial and archaeal diversity through a phylogenetically consistent, rank normalized and complete genome-based taxonomy**
Donovan H Parks, Maria Chuvochina, Christian Rinke, Aaron J Mussig, Pierre-Alain Chaumeil, Philip Hugenholtz
Nucleic Acids Research (2022-01-07) https://doi.org/gm97d8
DOI: 10.1093/nar/gkab776 · PMID: 34520557 · PMCID: PMC8728215

24. **Comparative genomics: the bacterial pan-genome**
Hervé Tettelin, David Riley, Ciro Cattuto, Duccio Medini
Current Opinion in Microbiology (2008-10) https://doi.org/dp79f2
DOI: 10.1016/j.mib.2008.09.006 · PMID: 19086349

25. **NCBI prokaryotic genome annotation pipeline**
Tatiana Tatusova, Michael DiCuccio, Azat Badretdin, Vyacheslav Chetvernin, Eric P Nawrocki, Leonid Zaslavsky, Alexandre Lomsadze, Kim D Pruitt, Mark Borodovsky, James Ostell
Nucleic Acids Research (2016-08-19) https://doi.org/f82gsk
DOI: 10.1093/nar/gkw569 · PMID: 27342282 · PMCID: PMC5001611

26. **RefSeq: expanding the Prokaryotic Genome Annotation Pipeline reach with protein family model curation**
Wenjun Li, Kathleen R O'Neill, Daniel H Haft, Michael DiCuccio, Vyacheslav Chetvernin, Azat Badretdin, George Coulouris, Farideh Chitsaz, Myra K Derbyshire, AScott Durkin, … Françoise Thibaud-Nissen

Nucleic Acids Research (2021-01-08) https://doi.org/gnrhsn
DOI: 10.1093/nar/gkaa1105 · PMID: 33270901 · PMCID: PMC7779008

27. **The Integrative Human Microbiome Project**
The Integrative HMP (iHMP) Research Network Consortium
Nature (2019-05) https://doi.org/gf3wp9
DOI: 10.1038/s41586-019-1238-8 · PMID: 31142853 · PMCID: PMC6784865

28. **High throughput ANI analysis of 90K prokaryotic genomes reveals clear species boundaries**
Chirag Jain, Luis M Rodriguez-R, Adam M Phillippy, Konstantinos T Konstantinidis, Srinivas Aluru
Nature Communications (2018-12) https://doi.org/gfknmg
DOI: 10.1038/s41467-018-07641-9 · PMID: 30504855 · PMCID: PMC6269478

29. **PALADIN: protein alignment for functional profiling whole metagenome shotgun data**
Anthony Westbrook, Jordan Ramsdell, Taruna Schuelke, Louisa Normington, RDaniel Bergeron, WKelley Thomas, Matthew D MacManes
Bioinformatics (2017-05-15) https://doi.org/gpm3j7
DOI: 10.1093/bioinformatics/btx021 · PMID: 28158639 · PMCID: PMC5423455

30. **Effects of Antibiotics upon the Gut Microbiome: A Review of the Literature**
Theocharis Konstantinidis, Christina Tsigalou, Alexandros Karvelas, Elisavet Stavropoulou, Chrissoula Voidarou, Eugenia Bezirtzoglou
Biomedicines (2020-11-16) https://doi.org/ghqcmk
DOI: 10.3390/biomedicines8110502 · PMID: 33207631 · PMCID: PMC7696078

31. **Identifying genomic and metabolic features that can underlie early successional and opportunistic lifestyles of human gut symbionts**
Catherine Lozupone, Karoline Faust, Jeroen Raes, Jeremiah J Faith, Daniel N Frank, Jesse Zaneveld, Jeffrey I Gordon, Rob Knight
Genome Research (2012-10) https://doi.org/f4bhq4
DOI: 10.1101/gr.138198.112 · PMID: 22665442 · PMCID: PMC3460192

32. **Nitroimidazole drugs-action and resistance mechanisms I. Mechanism of action**
David I Edwards
Journal of Antimicrobial Chemotherapy (1993) https://doi.org/bt9zf9
DOI: 10.1093/jac/31.1.9 · PMID: 8444678

33. **Eradication of Pathogenic Bacteria and Restoration of Normal Pouch Flora: Comparison of Metronidazole and Ciprofloxacin in the Treatment of Pouchitis**
Martijn P Gosselink, Rudolph W Schouten, Leo MC van Lieshout, Willem CJ Hop, Jon D Laman, Johanneke GH Ruseler-van Embden
Diseases of the Colon & Rectum (2004-09) https://doi.org/brhfp8
DOI: 10.1007/s10350-004-0623-y · PMID: 15486751

34. **Comparing and Evaluating Metagenome Assembly Tools from a Microbiologist's Perspective - Not Only Size Matters!**
John Vollmers, Sandra Wiegand, Anne-Kristin Kaster
PLOS ONE (2017-01-18) https://doi.org/f9m25t
DOI: 10.1371/journal.pone.0169662 · PMID: 28099457 · PMCID: PMC5242441

35. **MetaCherchant: analyzing genomic context of antibiotic resistance genes in gut microbiota**
Evgenii I Olekhnovich, Artem T Vasilyev, Vladimir I Ulyantsev, Elena S Kostryukova, Alexander V Tyakht

Bioinformatics (2018-02-01) https://doi.org/gcg2gv
DOI: 10.1093/bioinformatics/btx681 · PMID: 29092015

36. **PathRacer: Racing Profile HMM Paths on Assembly Graph**
Alexander Shlemov, Anton Korobeynikov
Algorithms for Computational Biology (2019) https://doi.org/gg8s3w
DOI: 10.1007/978-3-030-18174-1_6 · ISBN: 9783030181734

37. **Generating lineage-resolved, complete metagenome-assembled genomes from complex microbial communities**
Derek M Bickhart, Mikhail Kolmogorov, Elizabeth Tseng, Daniel M Portik, Anton Korobeynikov, Ivan Tolstoganov, Gherman Uritskiy, Ivan Liachko, Shawn T Sullivan, Sung Bong Shin, … Timothy PL Smith
Nature Biotechnology (2022-01-03) https://doi.org/gn3fkx
DOI: 10.1038/s41587-021-01130-z · PMID: 34980911

38. **Prokka: rapid prokaryotic genome annotation**
T Seemann
Bioinformatics (2014-07-15) https://doi.org/f6b3k4
DOI: 10.1093/bioinformatics/btu153 · PMID: 24642063

39. **Roary: rapid large-scale prokaryote pan genome analysis**
Andrew J Page, Carla A Cummins, Martin Hunt, Vanessa K Wong, Sandra Reuter, Matthew TG Holden, Maria Fookes, Daniel Falush, Jacqueline A Keane, Julian Parkhill
Bioinformatics (2015-11-15) https://doi.org/ggbhmt
DOI: 10.1093/bioinformatics/btv421 · PMID: 26198102 · PMCID: PMC4817141

40. **sourmash: a library for MinHash sketching of DNA**
C Titus Brown, Luiz Irber
The Journal of Open Source Software (2016-09-14) https://doi.org/ghdrk5
DOI: 10.21105/joss.00027

41. **VEGAN, a package of R functions for community ecology**
Philip Dixon
Journal of Vegetation Science (2003-12) https://doi.org/dz9txb
DOI: 10.1111/j.1654-1103.2003.tb02228.x

42. **An object-oriented framework for evolutionary pangenome analysis**
Ignacio Ferrés, Gregorio Iraola
Cell Reports Methods (2021-09) https://doi.org/gm3zts
DOI: 10.1016/j.crmeth.2021.100085

43. **Bakta: rapid and standardized annotation of bacterial genomes via alignment-free sequence identification**
Oliver Schwengers, Lukas Jelonek, Marius Alfred Dieckmann, Sebastian Beyvers, Jochen Blom, Alexander Goesmann
Microbial Genomics (2021-11-05) https://doi.org/gnfrj7
DOI: 10.1099/mgen.0.000685 · PMID: 34739369 · PMCID: PMC8743544

44. **Polyester : simulating RNA-seq datasets with differential transcript expression**
Alyssa C Frazee, Andrew E Jaffe, Ben Langmead, Jeffrey T Leek
Bioinformatics (2015-09-01) https://doi.org/f7rwtr
DOI: 10.1093/bioinformatics/btv272 · PMID: 25926345 · PMCID: PMC4635655

45. **Multi-omics of the gut microbial ecosystem in inflammatory bowel diseases**

IBDMDB Investigators, Jason Lloyd-Price, Cesar Arze, Ashwin N Ananthakrishnan, Melanie Schirmer, Julian Avila-Pacheco, Tiffany W Poon, Elizabeth Andrews, Nadim J Ajami, Kevin S Bonham, ... Curtis Huttenhower
Nature (2019-05) https://doi.org/ggd6wc
DOI: 10.1038/s41586-019-1237-9 · PMID: 31142855 · PMCID: PMC6650278

46. **The khmer software package: enabling efficient nucleotide sequence analysis**
Michael R Crusoe, Hussien F Alameldin, Sherine Awad, Elmar Boucher, Adam Caldwell, Reed Cartwright, Amanda Charbonneau, Bede Constantinides, Greg Edvenson, Scott Fay, ... CTitus Brown
F1000Research (2015-09-25) https://doi.org/9qp
DOI: 10.12688/f1000research.6924.1 · PMID: 26535114 · PMCID: PMC4608353

47. **MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph**
Dinghua Li, Chi-Man Liu, Ruibang Luo, Kunihiko Sadakane, Tak-Wah Lam
Bioinformatics (2015-05-15) https://doi.org/f7fb5z
DOI: 10.1093/bioinformatics/btv033 · PMID: 25609793

48. **MetaBAT 2: an adaptive binning algorithm for robust and efficient genome reconstruction from metagenome assemblies**
Dongwan D Kang, Feng Li, Edward Kirton, Ashleigh Thomas, Rob Egan, Hong An, Zhong Wang
PeerJ (2019-07-26) https://doi.org/gf9pmc
DOI: 10.7717/peerj.7359 · PMID: 31388474 · PMCID: PMC6662567

49. **krassowski/complex-upset: v1.3.3**
Michał Krassowski, Milou Arts, Cyril Lagger
Zenodo (2021-12-07) https://doi.org/gpm8md
DOI: 10.5281/zenodo.3700590

50. **Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM**
Heng Li
arXiv:1303.3997 [q-bio] (2013-05-26) http://arxiv.org/abs/1303.3997

51. **The Sequence Alignment/Map format and SAMtools**
H Li, B Handsaker, A Wysoker, T Fennell, J Ruan, N Homer, G Marth, G Abecasis, R Durbin, 1000 Genome Project Data Processing Subgroup
Bioinformatics (2009-08-15) https://doi.org/ff6426
DOI: 10.1093/bioinformatics/btp352 · PMID: 19505943 · PMCID: PMC2723002

52. **Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences**
W Li, A Godzik
Bioinformatics (2006-07-01) https://doi.org/ct8g72
DOI: 10.1093/bioinformatics/btl158 · PMID: 16731699

Appendix/Supplementary information

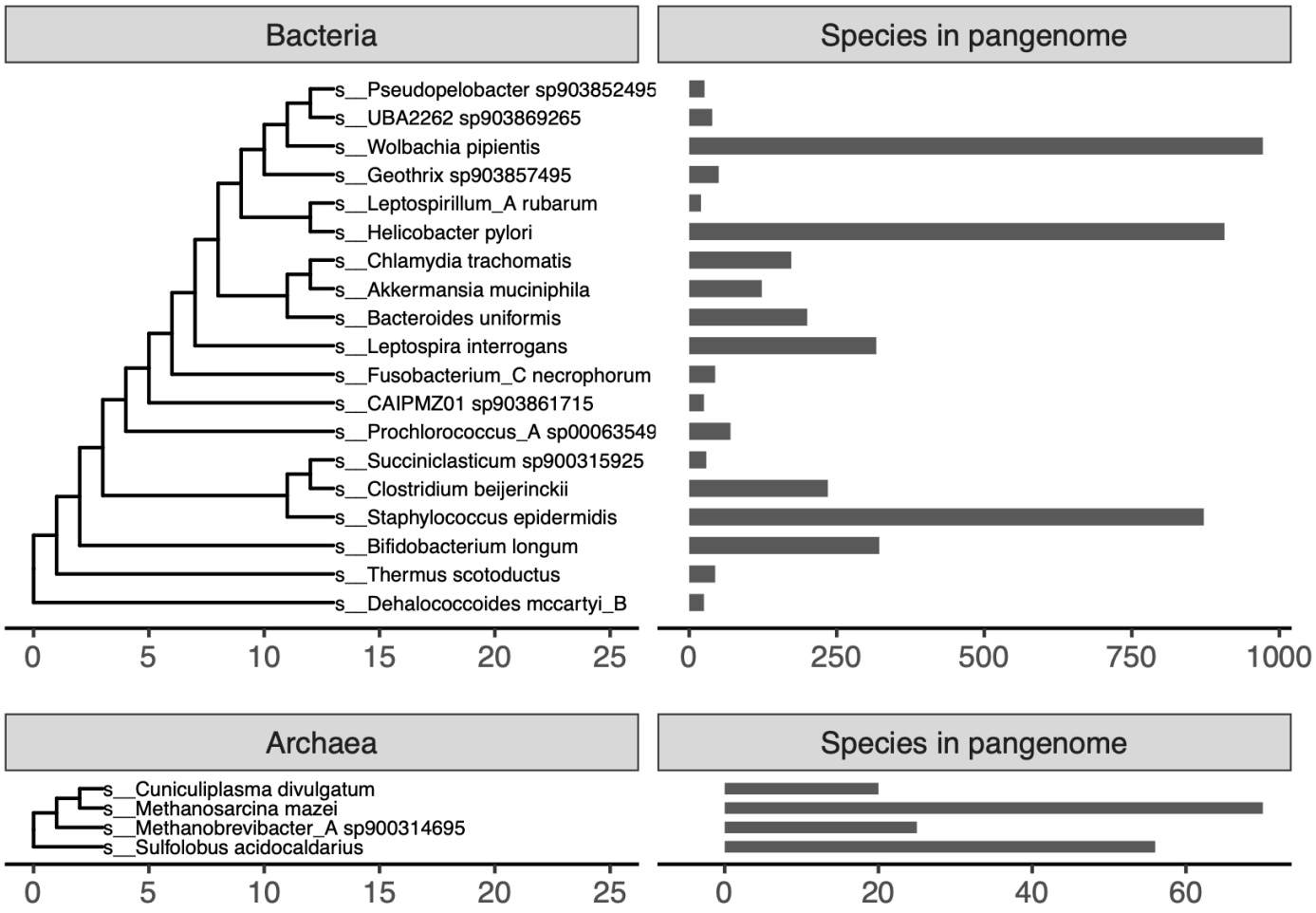

Figure 5: GTDB species used in this paper. These species were used to benchmark pangenome construction with reduced alphabet k-mers and open reading frame prediction from short sequencing reads. The trees are the default GTDB rs202 trees, with tips representing species not used in this paper removed.

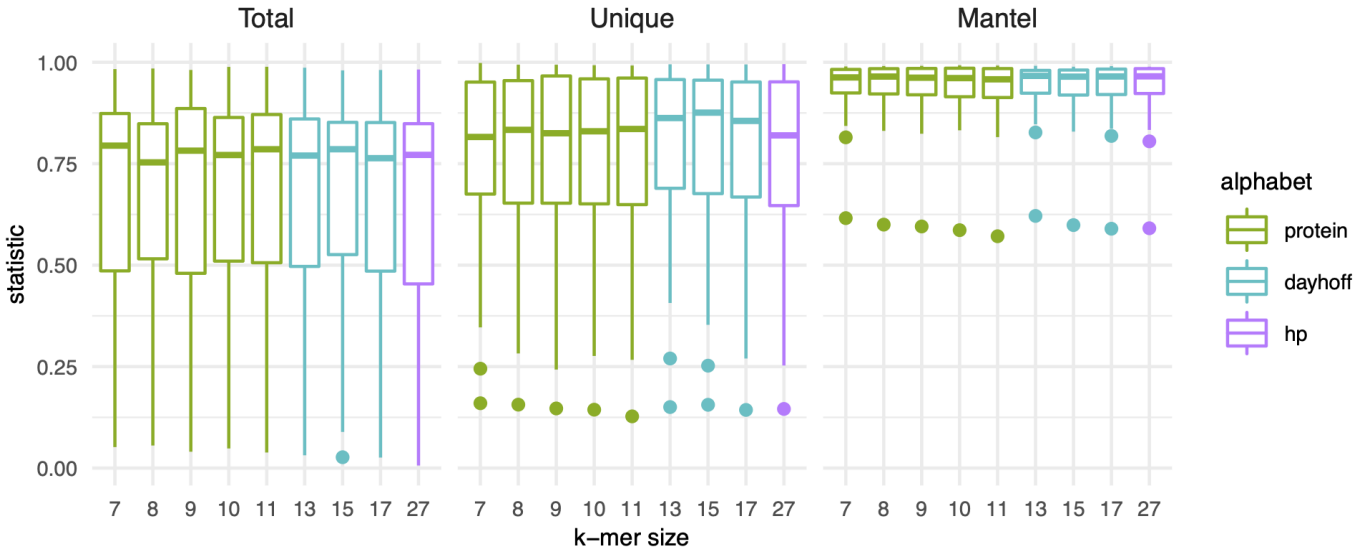

Figure 6: K-mer size and encoding do not impact pangenome estimation with k-mers. Box plots representing the distribution of R^2 values for linear models (Total, Unique) or statistic values for mantel tests (Mantel) calculated for each pangenome. All pangenomes are included, whether they contain genomes with the RefSeq exclusion criteria "many frameshifted proteins" or not. See figure legend for **Figure** 2 for a description of Total, Unique, and Mantel.

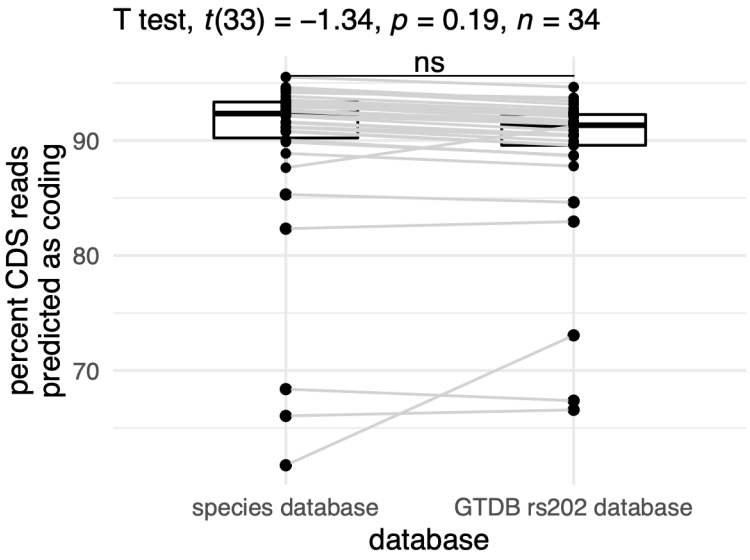

Figure 7: Percent of reads encoding coding domain sequences (CDS) that were predicted to be coding. There is no change between the percent of reads predicted to be derived from coding domain sequences when a species-level database is used versus when all of GTDB is used to predict open reading frames The slight increase observable for some species is a result of different thresholds, where we used 0.39 for the species database and 0.5 for the GTDB rs202 database.

Table 3: Metapangenome estimates for each species. *n* designates the number of metagenomes used to estimate the total, core, shell, cloud, and alpha values. Unlike isolate genomes, metagenomes may contain a fraction of an organism's genome if the metagenome was not sequenced deeply or if an organism was rare. To calculate the core, shell, and cloud fractions and to estimate the openness of the metapangenome, we removed samples with fewer than 10,000 k-mers.

species	n	total	core	shell	cloud	alpha
Bacteroides fragilis	7	24819	56.3%	11.3%	32.4%	0.76
Bacteroides uniformis	9	32197	38.0%	22.3%	39.7%	0.73
Enterocloster bolteae	4	23620	55.8%	18.3%	25.9%	0.66
Parabacteroides distasonis	7	25789	42.4%	30.9%	26.8%	0.74
Parabacteroides merdae	6	19985	63.2%	9.6%	27.1%	0.82
Phocaeicola vulgatus	11	41005	30.3%	20.4%	49.2%	0.65

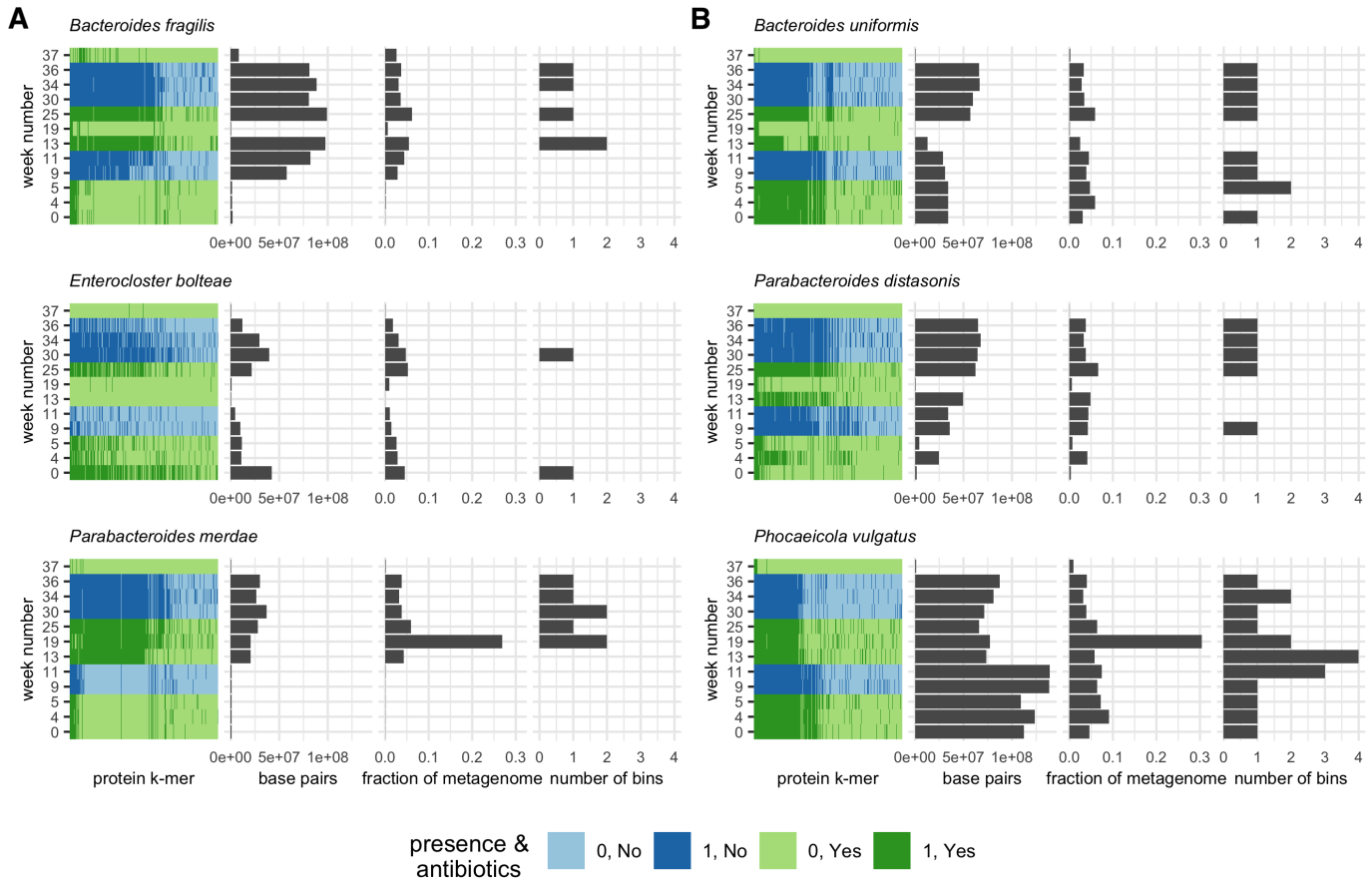

Figure 8: k_{aa}-mer metapangenomes for six species. Each species contains a four-panel figure. The first panel is a binmap plot. Dark colors represent k-mers that are present in each sample. Blue shades represent time points when the sampled individual was not on antibiotics, while green shades represent time points when the individual was on antibiotics. The second panel represents an estimated number of base pairs in the metagenome detected to originate from that species. The third panel represents an estimated fraction of the metagenome assigned to that species. The fourth panel represents the number of bins produced for that species from that sample using a *de novo* metagenome assembly and binning approach. The two values represented in the second and third panels and the species assignations used to infer the value represented in the fourth panel were inferred using the sourmash gather algorithm against the GTDB rs202 database. **A)** Species for which presence-absence fluctuated over the time series. **B)** Species for which strain presence-absence fluctuated over the time series.

Table 4: GTDB genomes used to benchmark orpheum accuracy.

accession	superkingdom	phylum	class	order	family	genus	species	NCBI taxid	NCBI organism name
GCF_000012125.1	d__Bacteria	p__Chlamydiota	c__Chlamydiia	o__Chlamydiales	f__Chlamydiaceae	g__Chlamydia	s__Chlamydia trachomatis	315277	Chlamydia trachomatis A/HAR-13
GCF_000381045.1	d__Bacteria	p__Deinococcota	c__Deinococci	o__Deinococcales	f__Thermaceae	g__Thermus	s__Thermus scotoductus	112339 1	Thermus scotoductus DSM 8553
GCF_900156205.1	d__Bacteria	p__Spirochaetota	c__Leptospirae	o__Leptospirales	f__Leptospiraceae	g__Leptospira	s__Leptospira interrogans	173	Leptospira interrogans

accession	superkingdom	phylum	class	order	family	genus	species	NCBI taxid	NCBI organism name
GCA_900315925.1	d__Bacteria	p__Firmicutes_C	c__Negativicutes	o__Acidaminococcales	f__Acidaminococcaceae	g__Succiniclasticum	s__Succiniclasticum sp900315925	1387507	uncultured Selenomonadales bacterium
GCF_000400665.1	d__Bacteria	p__Fusobacteriota	c__Fusobacteriia	o__Fusobacteriales	f__Fusobacteriaceae	g__Fusobacterium_C	s__Fusobacterium_C necrophorum	143388	Fusobacterium necrophorum subsp. necrophorum
GCA_903857495.1	d__Bacteria	p__Acidobacteriota	c__Holophagae	o__Holophagales	f__Holophagaceae	g__Geothrix	s__Geothrix sp903857495	904990	uncultured Holophagaceae bacterium
GCF_006742205.1	d__Bacteria	p__Firmicutes	c__Bacilli	o__Staphylococcales	f__Staphylococcaceae	g__Staphylococcus	s__Staphylococcus epidermidis	1282	Staphylococcus epidermidis
GCF_002006445.1	d__Bacteria	p__Firmicutes_A	c__Clostridia	o__Clostridiales	f__Clostridiaceae	g__Clostridium	s__Clostridium beijerinckii	1520	Clostridium beijerinckii
GCA_903861715.1	d__Bacteria	p__Patescibacteria	c__Paceibacteria	o__Moranbacterales	f__GWC2-37-73	g__CAIPMZ01	s__CAIPMZ01 sp903861715	771313	uncultured bacterium
GCF_000008025.1	d__Bacteria	p__Proteobacteria	c__Alphaproteobacteria	o__Rickettsiales	f__Anaplasmataceae	g__Wolbachia	s__Wolbachia pipientis	163164	Wolbachia endosymbiont of Drosophila melanogaster
GCF_000830885.1	d__Bacteria	p__Chloroflexota	c__Dehalococcoidia	o__Dehalococcoidales	f__Dehalococcoidaceae	g__Dehalococcoides	s__Dehalococcoides mccartyi_B	1432061	Dehalococcoides mccartyi CG5
GCF_000299235.1	d__Bacteria	p__Nitrospirota_A	c__Leptospirillia	o__Leptospirillales	f__Leptospirillaceae	g__Leptospirillum_A	s__Leptospirillum_A rubarum	1048260	Leptospirillum ferriphilum ML-04
GCA_000635495.1	d__Bacteria	p__Cyanobacteria	c__Cyanobacteriia	o__PCC-6307	f__Cyanobiaceae	g__Prochlorococcus_A	s__Prochlorococcus_A sp000635495	14714722	Prochlorococcus sp. scB243_495K23
GCA_903852495.1	d__Bacteria	p__Desulfobacterota_F	c__Desulfuromonadia	o__Geobacterales	f__Pseudopelobacteraceae	g__Pseudopelobacter	s__Pseudopelobacter sp903852495	214033	uncultured Geobacteraceae bacterium
GCA_903869265.1	d__Bacteria	p__Desulfobacterota	c__Desulfobulbia	o__Desulfobulbales	f__Desulfurivibrionaceae	g__UBA2262	s__UBA2262 sp903869265	340034	uncultured delta proteobacterium

accession	superkingdom	phylum	class	order	family	genus	species	NCBI taxid	NCBI organism name
GCF_000002025.1	d__Bacteria	p__Verrucomicrobiota	c__Verrucomicrobiae	o__Verrucomicrobiales	f__Akkermansiaceae	g__Akkermansia	s__Akkermansia muciniphila	349741	Akkermansia muciniphila ATCC BAA-835
GCF_900478295.1	d__Bacteria	p__Campylobacterota	c__Campylobacteria	o__Campylobacterales	f__Helicobacteraceae	g__Helicobacter	s__Helicobacter pylori	102618	Helicobacter pylori NCTC 11637 = CCUG 17874 = ATCC 43504
GCF_000154205.1	d__Bacteria	p__Bacteroidota	c__Bacteroidia	o__Bacteroidales	f__Bacteroidaceae	g__Bacteroides	s__Bacteroides uniformis	411479	Bacteroides uniformis ATCC 8492
GCF_000196555.1	d__Bacteria	p__Actinobacteriota	c__Actinomycetia	o__Actinomycetales	f__Bifidobacteriaceae	g__Bifidobacterium	s__Bifidobacterium longum	565042	Bifidobacterium longum subsp. longum JCM 1217
GCF_000001228.1	d__Archaea	p__Thermoproteota	c__Thermoprotei	o__Sulfolobales	f__Sulfolobaceae	g__Sulfolobus	s__Sulfolobus acidocaldarius	330779	Sulfolobus acidocaldarius DSM 639
GCF_000970205.1	d__Archaea	p__Halobacteriota	c__Methanosarcinia	o__Methanosarcinales	f__Methanosarcinaceae	g__Methanosarcina	s__Methanosarcina mazei	213585	Methanosarcina mazei S-6
GCA_900314695.1	d__Archaea	p__Methanobacteriota	c__Methanobacteria	o__Methanobacteriales	f__Methanobacteriaceae	g__Methanobrevibacter_A	s__Methanobrevibacter_A sp900314695	253161	uncultured Methanobrevibacter sp.
GCF_900083515.1	d__Archaea	p__Thermoplasmatota	c__Thermoplasmata	o__Thermoplasmatales	f__Thermoplasmataceae	g__Cuniculiplasma	s__Cuniculiplasma divulgatum	1673428	Cuniculiplasma divulgatum

Table 5: RefSeq genomes not in the GTDB rs202 database used to benchmark orpheum accuracy.

accession	NCBI taxid	NCBI organism name
GCF_003428625.2	2303751	Acidipila sp. 4G-K13
GCF_001700755.2	1160719	Cutibacterium granulosum DSM 20700
GCF_001884725.2	336810	Candidatus Sulcia muelleri
GCF_015356815.2	225148	Candidatus Rhabdochlamydia porcellionis
GCF_019599295.1	2866714	Oscillochloris sp. ZM17-4
GCF_020520145.1	936456	Desulfurispirillum indicum
GCF_019175305.1	2286	Saccharolobus shibatae
GCF_018282115.1	2732530	Synechocystis sp. PCC 7338
GCF_018863415.1	2774531	Deinococcus sp. SYSU M49105

accession	NCBI taxid	NCBI organism name
GCF_013456555.2	1710539	Natrinema sp. YPL30
GCF_000167435.2	1314	Streptococcus pyogenes
GCF_018205295.1	859	Fusobacterium necrophorum
GCF_019173545.1	1455061	Candidatus Magnetobacterium casensis
GCF_018398935.1	1123043	Telmatocola sphagniphila
GCF_000145825.2	629264	Pseudomonas syringae Cit 7
GCF_002442595.2	139	Borreliella burgdorferi
GCF_009156025.2	28903	Mycoplasmopsis bovis
GCF_019688735.1	2867247	Thermosulfurimonas sp. F29
GCF_018588215.1	1755816	Thermosipho sp. 1244
GCF_018336995.1	239935	Akkermansia muciniphila

Practical considerations for building k_{aa}-mer metapangenomes

Open reading frame prediction with orpheum

The RAM used to run orpheum is dictated by the database size, as the database is loaded into to memory while it is running. The GTDB rs202 nodegraph was 94 GB in size, and the RAM required to run orpheum never exceed 97GB, which makes database distribution and orpheum execution available on high performance compute clusters and other remote computers. Alternatively, species level databases were ~5 Mb in size, reducing the RAM and CPU time needed to run orpheum.

We demonstrated that orpheum is better able to predict open reading frames in genomes that have species-level representatives in the GTDB database. To asses whether this criteria is satisfied by a query genome without performing genome assembly, we recommend sourmash gather [16]. Sourmash gather will estimate the fraction of sequencing reads in a genome or metagenome that match to genomes in GTDB by comparing long nucleotide k-mers in the query against those in the database [16]. Alternatively, the tool SingleM could be used to perform this task (https://github.com/wwood/singlem). SingleM estimates the taxonomic composition of sequencing reads by identifying fragments of single copy marker genes in short reads and comparing them against a database of taxonomically labelled sequences.

These strategies may also be useful to predetermine the set of species-level databases to use for ORF prediction.

K_{aa}-mer pangenome construction

One consideration is for k_{aa}-mer sketch creation is the scaled value. The scaled parameter controls the fraction on k_{aa}-mers included in each sketch. We have found that a scaled value of 100 works well for comparing proteomes (cite TESSA?). For a subset of pangenomes benchmarked in results section, “K-mer methods accurately predict open reading frames in short sequencing reads”, we tested scaled = 1 and scaled = 100 (n = 8; s__Akkermansia muciniphila, s__Chlamydia trachomatis, s__Faecalibacterium prausnitzii_D s__Gemmiger formicilis, s__Geothrix sp903857495, s__Methanobrevibacter_A-sp900314695, s__Thermus scotoductus, s__UBA2262 sp903869265). We correlated the number of k_{aa}-mers and genes per pangenome, the number of unique k_{aa}-mers and genes per pangenome, and the similarity between genomes using k_{aa}-mers or genes with sketches

generated using a scaled of 1 or a scaled of 100. When then performed a second order correlation to determine if both scaled values produced the same results. All values were strongly significantly correlated with an $R^2 > 0.99$ ($p < 0.001$), indicating that scaled 100 sketches captured the same patterns as scaled 1 sketches. While we have not experimented with the upper bound of the scaled value that will still produce accurate results, using a scaled of 100 substantially decreased compute times.