Detection of low-abundance bacterial strains in metagenomic datasets by eigengenome partitioning

Brian Cleary[1,2], Ilana Lauren Brito[2–4], Katherine Huang[2], Dirk Gevers[2], Terrance Shea[2], Sarah Young[2] & Eric J Alm[2–4]

Analyses of metagenomic datasets that are sequenced to a depth of billions or trillions of bases can uncover hundreds of microbial genomes, but naive assembly of these data is computationally intensive, requiring hundreds of gigabytes to terabytes of RAM. We present latent strain analysis (LSA), a scalable, *de novo* pre-assembly method that separates reads into biologically informed partitions and thereby enables assembly of individual genomes. LSA is implemented with a streaming calculation of unobserved variables that we call eigengenomes. Eigengenomes reflect covariance in the abundance of short, fixed-length sequences, or *k*-mers. As the abundance of each genome in a sample is reflected in the abundance of each *k*-mer in that genome, eigengenome analysis can be used to partition reads from different genomes. This partitioning can be done in fixed memory using tens of gigabytes of RAM, which makes assembly and downstream analyses of terabytes of data feasible on commodity hardware. Using LSA, we assemble partial and near-complete genomes of bacterial taxa present at relative abundances as low as 0.00001%. We also show that LSA is sensitive enough to separate reads from several strains of the same species.

Marine, soil, plant and host-associated microbial communities have all been shown to contain vast reservoirs of genomic information, much of which is from species that cannot be cultured in a laboratory[1–8]. Because a single metagenomic sample can include thousands or millions of species[3], researchers frequently sequence billions of bases in order to capture sufficient genomic coverage of a representative portion of a given population. Deconvolving a hidden mixture of unknown genomes from hundreds of gigabytes to terabytes of data is a substantial computational challenge[9].

A suite of tools has been developed to enable analyses of metagenomic datasets. Tools such as MetAMOS[10], MetaVelvet[11], Meta-IDBA[12], Ray Meta[13] and diginorm with khmer[14,15] relax the assumptions of single-genome de Bruijn assemblers to allow multiple coverage/multiple strain assembly and have produced better results compared with standard de Bruijn assemblies, such as those produced by Velvet[16]. However, early meta-assemblers cannot scale to terabyte data sets; in practice it can be a challenge to find the compute (RAM) resources to process even a single, 100-Gb sample.

Several meta-assemblers that use a combination of data reduction, data compression and partitioning have been designed to scale to larger datasets[13–15,17]. Diginorm and khmer[14,15], for example, reduce the effective size of a dataset by eliminating redundancy of very high-coverage reads, compressing data with a probabilistic de Bruijn graph and partitioning data using graph connectivity. Once partitioned, short reads can be assembled and analyzed with relatively small amounts of RAM. Other methods, such as Ray Meta[13], leverage distributed architectures to parallelize assembly computations across many nodes. However, these tools result in multiple small contigs, and further, it is not clear which contigs originate from the same species.

Covarying patterns of contig depth across samples can be used to infer biological linkage. Previous studies demonstrated the power of a pooled analysis of multiple samples by using contig depth covariance to reconstruct genomes of low-abundance species *de novo*[18–22]. These methods work remarkably well, but require that assembly occurs before data partitioning. This introduces a bottleneck because datasets larger than ~100 Gb cannot be assembled in RAM on most computers. Metagenomic studies with a focus on large collections, complex communities, or on not abundant and rare species, could easily generate data that surpass these limits. Therefore, analytical methods should have memory requirements that are independent of sequence depth.

We present a method for metagenomic read partitioning that can use covariance information to cluster sequences in fixed memory. Inspired by latent semantic analysis and associated algorithms in natural language processing[23,24], we use a deconvolution algorithm to identify clusters of hashed *k*-mers that represent hidden variables. A streaming singular value decomposition (SVD) of a *k*-mer abundance matrix operates in fixed memory and defines two sets of latent, orthogonal vectors that we refer to as eigensamples and eigengenomes. Each of these vector sets is analogous to the principle components of sample and sequence space, respectively. Eigengenomes defined by this decomposition are used to cluster *k*-mers and partition reads. Short reads that originate on the same physical fragment of DNA should partition together, regardless of sequence composition or graph connectivity.

We use LSA to analyze a subset of samples from the Human Microbiome Project[25] spiked with mock reads from reference

genomes, and to analyze three metagenomic collections containing 10 Gb[18], 300 Gb and 4 Tb of data. Highly distributed partitioning was carried out in fixed memory (independent of input scale), and implicit biological associations were validated by observing partition-specific enrichment of reference genomes known to be present in the data.

RESULTS

We assessed five aspects of LSA performance: (i) the degree to which LSA recruits reads from a single genome to a common partition; (ii) the ability to separate sequences from related genomes into different partitions; (iii) computational resource consumption as a function of sequencing depth and number of samples; (iv) the fidelity of assembled partitions with respect to known references; and (v) the enrichment of very low abundance genomes in deeply sequenced data. We also discuss the potential for new genome discovery and identification of ecological relationships.

Partitioning reads from data spiked with a single genome

We first sought to test to what extent reads from a single genome were grouped together in a single partition. Mock reads from a reference *Salmonella bongori* genome were spiked into 30 subsampled gut metagenomes from the Human Microbiome Project (HMP)[25] at an average abundance of 1.8% (see Online Methods and **Supplementary Table 1**). The subsampled HMP data, which did not include any detectable *Salmonella* genomes, served as metagenomic background. We quantified the fraction of spiked *S. bongori* reads recruited to a single partition. Running the LSA algorithm on these samples produced 451 partitions of the initial 600 million reads using a maximum of 25 Gb of RAM. Out of a total of ~20 million spiked *S. bongori* reads, more than 99% ended up in a single partition.

Assembly of the primary *S. bongori* partition contained the reference genome, along with portions of several non-*Salmonella* HMP background genomes. Although 4.9 Mb of the assembled partition aligned back to the *S. bongori* reference, a total of 18 Mb were assembled from reads in this partition. So, although this experiment successfully demonstrated that LSA can group reads from one genome in a single partition, a higher resolution analysis is needed to determine if LSA can isolate individual genomes.

The relatedness of reads in a single partition can vary depending on the overall resolution of partitioning. At low resolution, relatedness might capture reads from organisms that covary, whereas at an intermediate resolution, partitions might contain a single

genome, or a small set of genomes, that largely overlap in sequence. High-resolution partitioning could separate very similar genomes, or identify variable, genome fragments.

Separating reads from closely related strains

We hypothesized that using LSA at high resolution could produce partitions that separate strains of the same species into individual genomes. To test this hypothesis, we spiked mock reads from a clade of *Salmonella* reference genomes into a collection of 50 subsampled HMP gut metagenomes. We included two strains of *S. bongori* and eight strains of *S. enterica*, all of which were absent from the HMP samples. Pairwise distances between the strains, calculated as the MUM index[26], ranged from very similar (0.005) to moderately divergent (0.679) (see tree in **Fig. 1**, and multiple alignment in **Fig. 2**). Altogether, spiked reads from these strains accounted for 5% of the total read count in the entire collection, and the average abundance of each strain was 0.50% (min. 0.20%, max. 0.79%) (**Supplementary Table 2**).

LSA of the pooled data produced 2,543 partitions, each containing disjoint subsets of the original 2 billion reads. The partition most enriched in *Salmonella* sequences contained 9,817,505 total reads, of which 9,801,109 (99.83%) were derived from the spiked-in genomes. The two *S. bongori* genomes together accounted for the vast majority (9,770,839) of these reads, whereas 30,270 reads were from other species of spiked-in *Salmonella*, and 16,396 reads came from the HMP background. This represents a near-perfect separation of the 10.2 million *S. bongori* reads from the overall background of 2 billion reads. Overall, the 15 partitions most enriched in *Salmonella* sequences accounted for nearly half of all *Salmonella* mock reads. Interestingly, the partition with the most mock reads was not among the most enriched in *Salmonella*, as only 19% of the reads in this partition were from the spiked-in genomes. Rather than representing a single species, this partition seemed to represent a set of reads that were
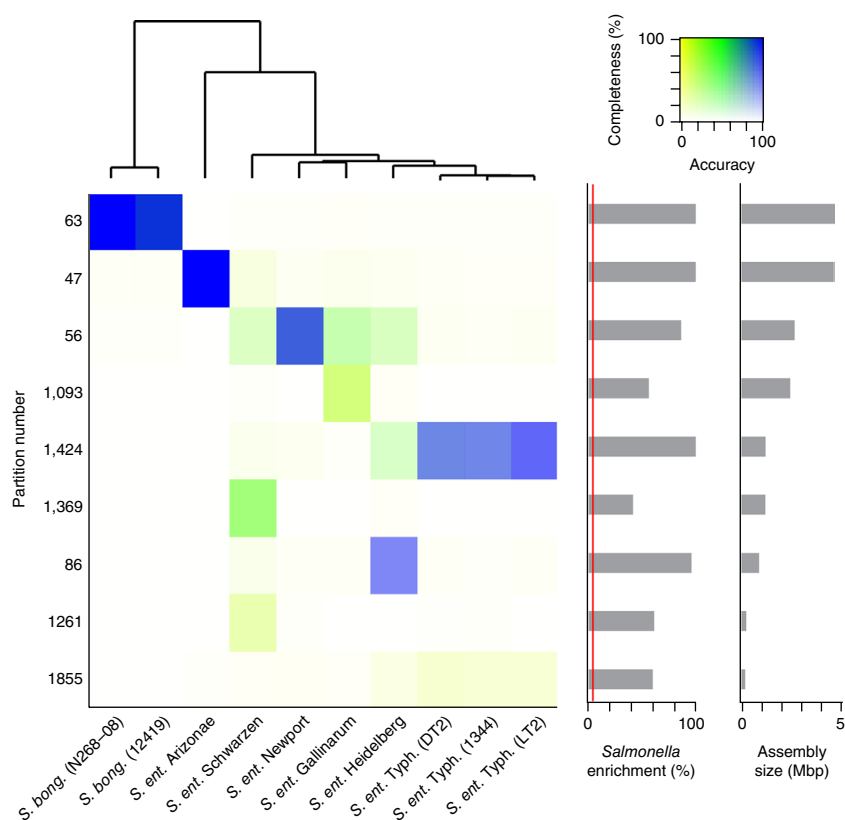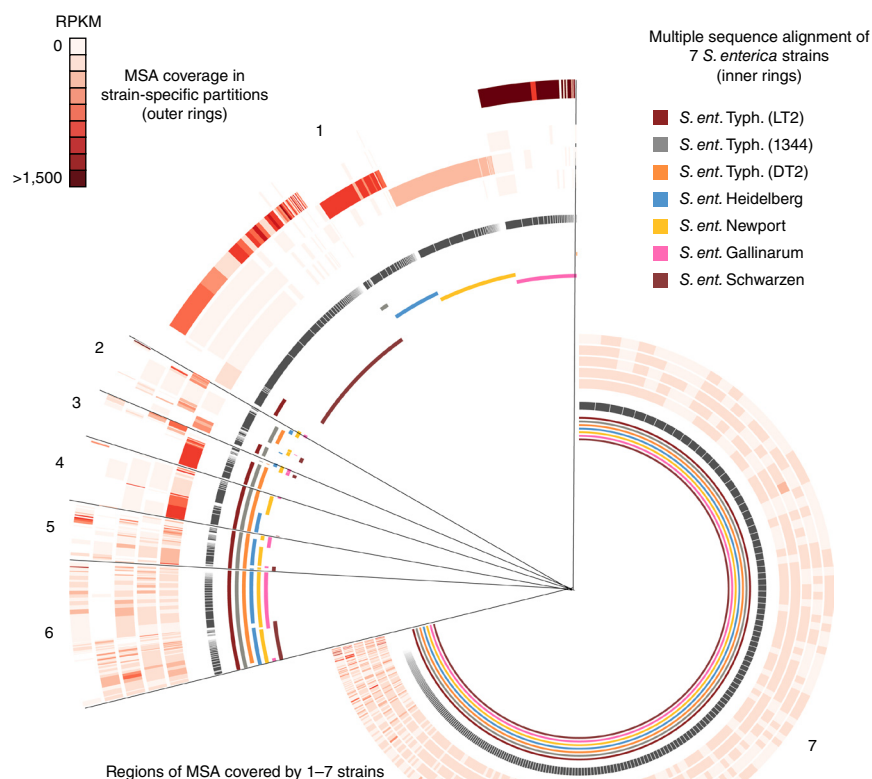

Figure 1 Accuracy and completeness of recovered genomes. The accuracy of *Salmonella*-enriched partitions (rows) with respect to each strain (columns) is depicted on a color scale. Saturation of each color indicates the completeness of each assembly with respect to each strain. Bars in the two right-hand panels indicate the fraction of reads in a partition coming from any *Salmonella* strain (red line = 5%; the background abundance of spiked-in *Salmonella* reads) and the total assembly length. The tree at the top was constructed using MUMi distance between strains. *S. bong.*, *S. bongori*; *S. ent.*, *S. enterica*.

Figure 2 *S. enterica* multiple genome alignment. Multiple sequence alignment (MSA) blocks (gray ring) are ordered by their conservation across 1–7 strains. The inner rings depict portions of each genome that align to each MSA block. Within five *S. enterica*–enriched partitions, the read depth at each MSA block is shown as a heatmap in the outer rings. Partition numbers from the center, outwards are: 1424, 56, 86, 1369 and 1093. RPKM, reads per kilobase per million.

Regions of MSA covered by 1–7 strains

RPKM
0
>1,500

MSA coverage in strain-specific partitions (outer rings)

Multiple sequence alignment of 7 *S. enterica* strains (inner rings)

- *S. ent.* Typh. (LT2)
- *S. ent.* Typh. (1344)
- *S. ent.* Typh. (DT2)
- *S. ent.* Heidelberg
- *S. ent.* Newport
- *S. ent.* Gallinarum
- *S. ent.* Schwarzen

present in roughly equal abundance across all samples. This highlights a more general outcome that can arise when using sequence covariance to separate strains. Genomic regions that are present in all strains can have a distribution that is distinct from the more variable regions, thereby separating the 'core' and 'flexible' genome into different partitions. Currently, manual curation is required to infer complete genomes in these cases, which represents an opportunity for new algorithms to improve these results.

For each partition enriched in one or more of our ten spiked-in genomes, we quantified partition accuracy and completeness with respect to each strain. The accuracy of each partition with respect to a given genome was calculated as the percentage of that partition's assembly covered by reads simulated from the genome in question. Similarly, completeness measured the percentage of a given reference genome covered by each partition. **Figure 1** shows the complete and accurate recovery of *S. bongori* and *S. ent.* Arizonae genomes, as well as the partial separation and recovery of more closely related strains.

Within the species *S. enterica* we identified five partitions that were relatively specific to five of the eight strains included as spiked-in mock reads; partitions 47 (*S. ent.* Arizonae), 56 (*S. ent.* Newport), 86 (*S. ent.* Heidelberg), 1093 (*S. ent.* Gallinarum) and 1349 (*S. ent.* Schwarzengrund). The remaining three strains (all *S. ent.* Typh.) were very similar (maximum MUMi[26] distance 0.022, average nucleotide identity 99.93%), and were found together in a single partition (1424). This indicates a threshold of similarity of ~99.5% average nucleotide identity (ANI), above which related genomes were not resolved. Below this threshold at a maximum similarity of 99.1% ANI, the remaining strains of *S. enterica* were isolated with a median accuracy of 91.55% (ranging from 16.3% to 99.5%) and median completeness of 56.26% (ranging from 49.5% to 99.6%).

Through multiple genome alignment and coverage analysis by partition, we were able to look in detail at the genomic regions captured in each *S. enterica* partition (**Fig. 2**). Partition 1424 (*S. ent.* Typh.) was enriched in reads covering a portion of the multiple sequence alignment conserved across the three *S. ent.* Typh. genomes (region 3 in **Fig. 2**). The remaining *S. enterica* partitions each covered regions of the multiple alignment specific to just one strain; that is, portions of these genomes were separated into strain-specific partitions (region 1 of **Fig. 2**). Of note, these partitions also contain some reads from the core alignment conserved across all seven strains (region 7 of **Fig. 2**), which enabled assembly of much larger regions than the strain-specific fragments.

Side-by-side comparison of the multiple sequence alignment along with accuracy and completeness results offers a comprehensive view of these partitions. For those partitions that were incomplete, but nonetheless specific to certain strains, the assemblies capture unique portions of each genome. In these cases, the core conserved portion

Table 1 LSA pipeline

Pipeline step	General computational requirements			Number of tasks per collection			Peak memory use per collection		
	Number of tasks	Time per task (h)	RAM/task	FijiCoMP stool 176 samples	HUGE 32 samples	Sharon *et al.*[18] 18 samples	FijiCoMP stool 4 Tb	HUGE 300 Gb	Sharon *et al.*[18] 20 Gb
Hyperplane hashing	Number of reads/millions	1.2	3 Gb	17,516	3,184	561	3 Gb	3 Gb	3 Gb
Hashed *k*-mer counting	Number of samples	2	4 Gb	176	32	18	4 Gb	4 Gb	2 Gb
Global *k*-mer weighting	1	1.7	25 Gb	1	1	1	25 Gb	25 Gb	12 Gb
k-mer abundance matrix	Number of samples	0.45	32 Gb	176	32	18	32 Gb	32 Gb	16 Gb
Streaming SVD[a]	1	78[b]	4 Gb	1	1	1	4 Gb	4 Gb	2 Gb
k-mer clusterindex	1	24	1 Gb	1	1	1	1 Gb	1 Gb	1 Gb
k-mer clustermap	*k*-mer hash size/millions	2.2	1 Gb	2,148	2,148	1,074	1 Gb	1 Gb	1 Gb
k-mer clusterreduce	1	1.1	50 Gb	1	1	1	50 Gb	50 Gb	25 Gb
Read partitioning	Number of reads/millions	6.2	1 Gb	17,516	3,148	561	1 Gb	1 Gb	1 Gb

[a]SVD, singular value decomposition. [b]78 h for the largest data set. Scales with the number of samples.

Figure 3 Latent strain analysis pipeline. Metagenomic samples containing multiple species (depicted by different colors) are sequenced. Every *k*-mer in every sequencing read is hashed to one column of a matrix. Values from each sample occupy a different row. Singular value decomposition of this *k*-mer abundance matrix defines a set of eigengenomes. *k*-mers are clustered across eigengenomes, and each read is partitioned based on the intersection of its *k*-mers with each of these clusters. Each partition contains a small fraction of the original data and can be analyzed independently of all others. SVD, singular value decomposition.

1. Input collection of multiple sequenced metagenomes

Sample 1

Sample *n*

2. Hashing, counting and weighting of *k*-mers
Hashed *k*-mers
(~2 billion columns)

Samples (*n* rows)

$a_{i,j} = $ Conditioned abundance of *k*-mer$_j$ in sample$_i$

3. Streaming SVD defines a set of eigengenomes

$$M = LVR^T$$

Eigengenomes

4. *k*-mer clustering and read partitioning

Partitioned reads

Input reads
+
k-mer clusters

5. Assembly etc. of each partition in parallel

of each genome is found in a separate partition. We considered that allowing reads to be assigned to multiple partitions might allow complete recovery of these strains, but found that this modification reduced strain-specific accuracy (**Supplementary Fig. 1**).

LSA enables analysis of terabytes of data in fixed memory

To assess scalability and computational performance, we ran LSA on three publicly available metagenomic collections that span orders of magnitude in scale. The first collection contains 176 human stool samples from the Fiji Community Microbiome Project (FijiCoMP; terabyte scale), the second contains 32 human stool samples from HUGE (hundreds of gigabytes) and the third contains 18 premature infant gut libraries used in a previously published method for covariance partitioning[18,27] (tens of gigabytes).

Although the number of computing tasks varied considerably across collections, the maximum memory footprint of the entire pipeline was relatively constant. The majority of computation for each dataset was performed in distributed tasks requiring less than 5 Gb of RAM each (**Table 1** and **Fig. 3**), and, importantly, more than 90% of reads were partitioned in every collection. Peak memory consumption occurred in a step running on a single node, and was just 50 Gb for the two larger data sets, and 25 Gb for the smaller data set.

The peak memory consumption of LSA is a function of the hash size chosen for each collection. Our hyperplane-based *k*-mer hashing function is used to map individual *k*-mers to columns in a matrix. Although a matrix with the full space of potential 33-mers (4^{20}) would be intractably large, our hash function was limited to 2^{31} unique values (2^{30} for the smallest data set). This was large enough to accommodate a realistic *k*-mer diversity, while still being small enough to be computable. In the mapping from sequence to matrix column, it is inevitable that two *k*-mers end up mapping to the same column by random chance. However, we have devised the hash function such that *k*-mers of greater similarity in sequence are more likely to collide in the same column. For example, with 33-mers and a hash size of 2^{31}, a mismatch of one letter results in a collision probability of 5.23%, whereas a mismatch of five letters results in a collision probability of 0.08%. Using a smaller hash table makes for faster computations, but also limits the total hashed *k*-mer diversity. The chosen size for the two larger collections (2^{31}) was greater than that for the small collection (2^{30}) to accommodate a greater expectation of microbial diversity and number of unique *k*-mers. Thus, although the memory requirements of LSA were chosen to scale with the size and diversity of genomes present in individual samples, they were invariant with respect to the total number of samples.

A comparison between LSA and previously published methods was readily available for our analysis of the premature infant gut collection[18]. Each method separated genomes from the most common genera (*Enterococcus*, *Propionibacterium* and *Staphylococcus*), and assembly statistics of the reconstructed genomes were comparable, although our N50s (the largest contig length for which all larger contigs capture 50% of the total assembly) were smaller (e.g., 1.045 Mbp vs. 1.45 Mbp for *Enterococcus fecalis*) (**Supplementary Table 3**).
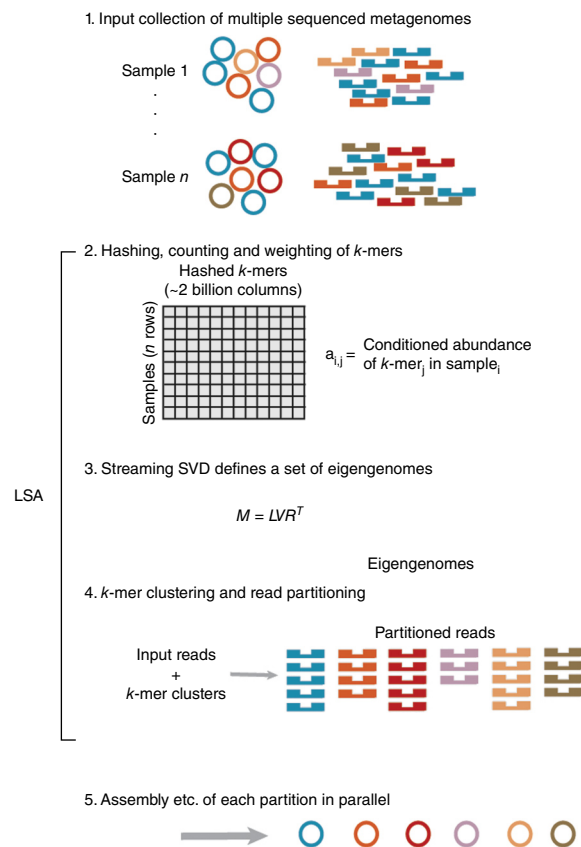
The differences are partially a reflection of the careful, manually curated assembly process undertaken in the previous results, and of the fact that our partitions contain additional, ecologically related sequences (discussed below).

We could not, however, accurately compare the results of this earlier method in the analysis of the 300-Gb and 4-Tb collections, owing to prohibitive RAM requirements in the initial assembly described[18], but there is a similar bottleneck posed at this stage in other methods[20–22]. Indeed, the scale of analysis that we demonstrate with the FijiCoMP data (>10^{10} reads) is ~10× larger than any previously published analyses (**Table 2**), which were either naturally smaller, or downsampled to artificially reduce the effective data size[21]. Moreover, given the fixed memory scaling that we use, it is likely that LSA can scale to even larger data sets without the need for any down-sampling. We also present a step-by-step comparison with the most closely related alternative method, CONCOCT[20]. By assembling first CONCOCT produces a relatively small abundance matrix (roughly contigs by samples), when compared with LSA (roughly *k*-mers by samples; **Table 2**). The difference in number of *k*-mers versus number of contigs highlights a potential memory bottleneck in LSA, which is handled by hyperplane hashing and streaming SVD. Thus we were able to achieve superior results, with regards to genome separation and sensitivity to low-abundance organisms.

Assembled partitions can be aligned to reference genomes

LSA of the 4-Tb FijiCoMP collection produced 4,306 partitions, which were assembled in parallel, and then aligned to a microbial reference database. Assembly length ranged from 77 kbp to 100 Mbp (median 2.2 Mbp), and the fraction of these assemblies that aligned to any reference sequence varied from 38.6% to 90.9% (**Supplementary Table 4**). Analysis of these alignments shows that, at this resolution,

Table 2 Methods comparison[a]

	GroopM[22]	CONCOCT[20]	Nielsen *et al.*[21]	LSA
Demonstrated scale (billions of reads)	~0.1	~1	~1	~10
	CONCOCT		LSA	
k-mer hashing method	None		Complex simplex plus hyperplane hash	
Per-sample abundance vector	Contigs (>1 kbp) plus 4-mers		Hashed 33-mers	
Abundance matrix size	Number of samples by ~10^4		Number of samples by ~10^9	
Matrix decomposition	SVD		Streaming SVD	
k-mer/contig clustering	*N*-dimensional Gaussian mixture model		High-dimensional randomized, modified *k*-means	
Partition-specific analysis	Classification, annotation		Assembly, classification, annotation	

[a]A more in-depth comparison is done for CONCOCT and LSA, the two methods most closely related conceptually.

most partitions contained small fragments from multiple bacterial families. However, when considering only contigs greater in length than the N50 of a given partition, we observed a considerable increase in specificity for most partitions. 344 partitions are relatively specific (>50% of total alignment) to one microbial family after applying this filter, and we find 48 partitions with clear specificity (>80%) for one family (**Supplementary Table 5**). For example, in the most specific partition, 99.5% of contigs >2 kbp (the N50 for that partition) aligned to *Enterobacteriaceae*, and the total length of this alignment was 4.04 Mbp. Using a sequencing depth–based cutoff also increased specificity, but to a lesser degree for the large majority of partitions. Those partitions with bimodal depth distributions would benefit the most from this alternative cutoff, but most partitions were characterized by unimodal distributions.

Completeness of each assembly was also assessed by searching for the presence of a set of conserved housekeeping genes (the AMPHORA set[28]) contained within each filtered assembly. The most specific partition contained a full complement, 31 out of 31 genes, and of the 344 partitions with >50% alignment to one family, 93 contained all 31 AMPHORA genes (**Supplementary Table 6**). These partitions, therefore, achieve a considerable degree of specificity, while still capturing complete genomes according to this proxy measure.

For nonspecific partitions with a mixture of genomes, it is possible that the abundance profiles of the genomes in question were too similar to be separated in the initial, global analysis. We therefore re-ran LSA on five such partitions (**Supplementary Table 7**), isolating the analysis to reads within a single partition each time and adjusting parameters to account for the small size (~1 Gb) of the data set. Starting with a partition that was relatively enriched to begin with

(partition 3094, 72.5% *Streptococcus*), we found that re-running LSA enriched the majority genus from 72.5% abundance to over 90% in five subpartitions; the largest subpartition assembled to a length of 784 kbp, and 724 kbp of this aligned to *Streptococcus salivarius* (97.8% of total alignment). Several other subpartitions were also enriched to >90% *Streptococcus salivarius*, each having assembly length of ~150 kbp. We next examined a nonspecific partition that clearly contained a mixture of genomes. Based on MetaPhyler analysis, partition 3957 contained 42% *Bifidobacterium* by abundance, 23% *Bacteroides* and 11% *Fecalibacterium*. By rerunning LSA on this partition with adjusted parameters, we were able to further resolve genome fragments from each genus. The five largest subpartitions by alignment length were enriched to 91% *Bifidobacterium longum*, 99% *Fecalibacterium prausnitzii*, 30% uncultured organism, 86% *Bacteroides vulgatus* and 93% *Bifidobacterium adolescentis*, respectively (**Supplementary Fig. 2**).

In every metagenomic dataset we analyzed, the most abundant families were found to have multiple family-specific partitions. For example, among the 344 most specific FijiCoMP partitions, 152 were specific to the most abundant family, *Prevotellaceae*, and 32 of these had complete AMPHORA gene sets. Similarly, in the HUGE data collection, the most abundant family, *Bacteroidaceae*, was enriched to >90% relative abundance in seven separate partitions. These numbers are suggestive of the in-family diversity for the most abundant organisms in each collection and indicate, for FijiCoMP, in particular, that a broad clade of *Prevotellaceae* exists within the population.

To investigate the differences underlying specific and complete partitions within one family, we looked in detail at two *Bacteroidaceae*-dominated partitions in the HUGE collection. Assembly of each
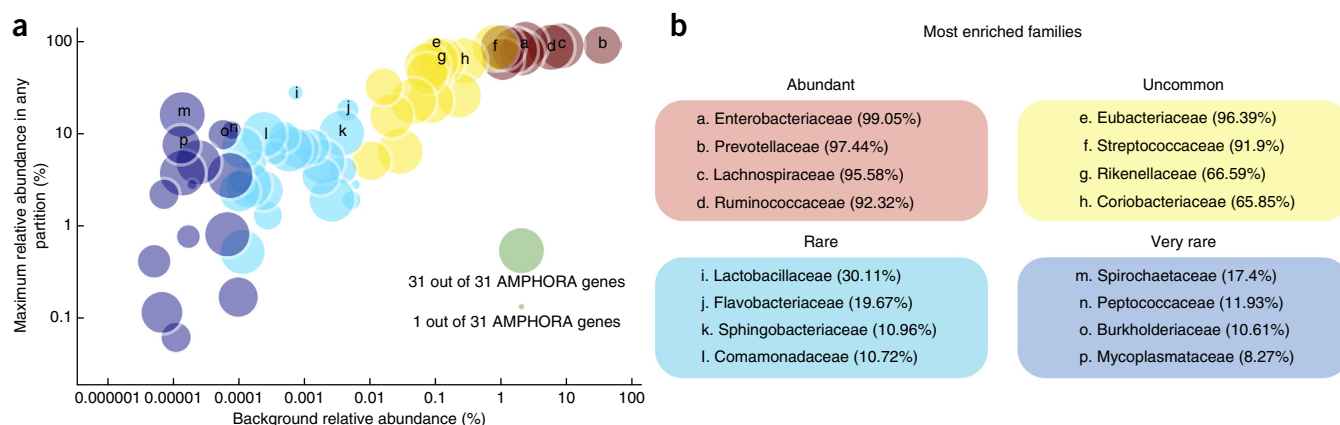

Figure 4 Enrichment of bacterial families spanning six orders of magnitude in abundance. Each circle represents one family in the FijiCoMP stool collection. The *x* axis is the background (unpartitioned) abundance of each family, as determined by species-specific 16S ribosomal DNA. *Y*-axis values are the maximum relative abundance in any one partition, as measured by MetaPhyler analysis of marker genes. Circle size is determined by the number of AMPHORA genes in the assembly of each partition.
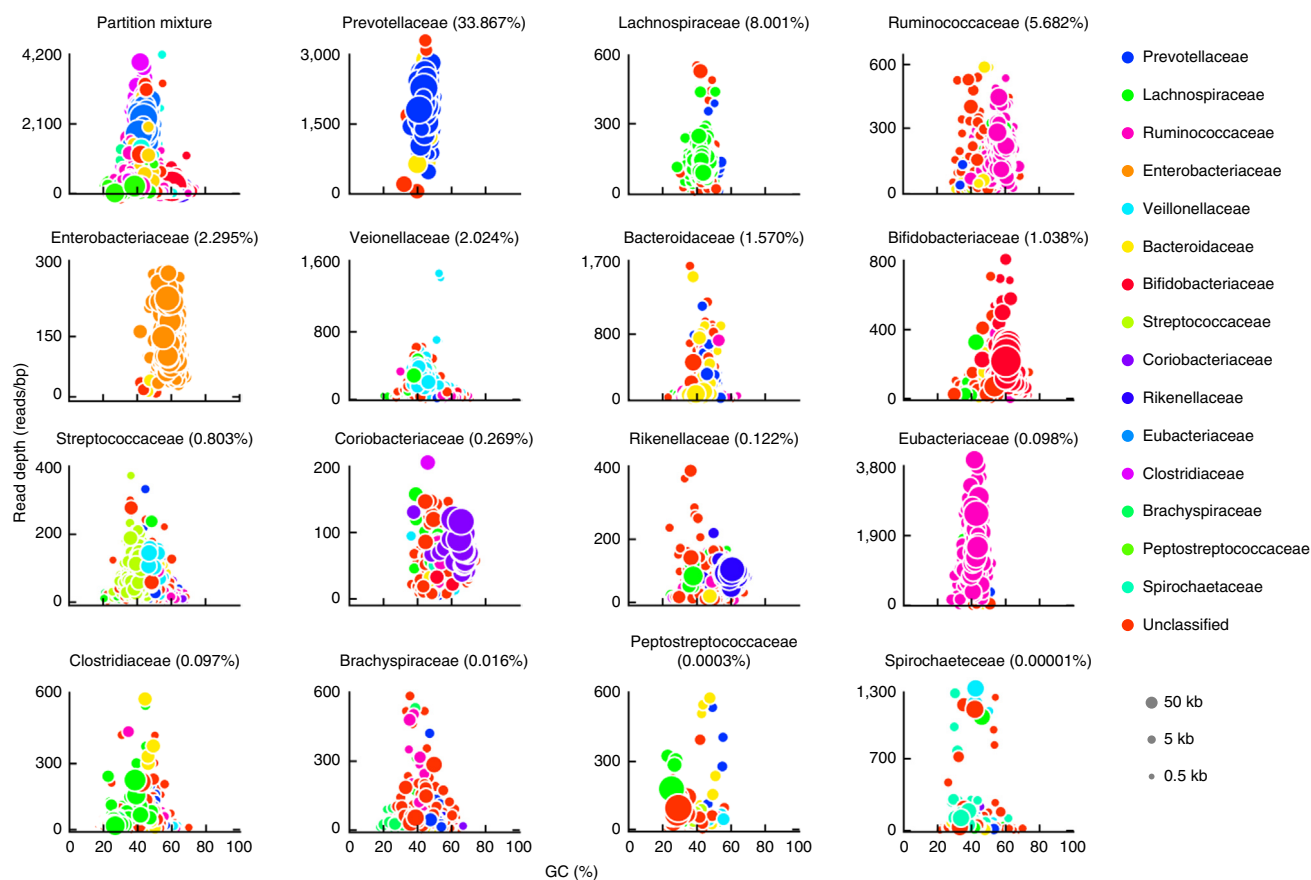
Figure 5 GC content versus contig depth. Plotted are the GC content (*x* axis) and depth (*y* axis) for contigs in partitions representing the top 15 enriched families from the FijiCoMP collection. Size of each circle represents the length of each contig. For each family the background abundance is indicated in parentheses.

partition was relatively complete (29 out of 31 genes, and 31 out of 31 AMPHORA genes). To determine if these assemblies represented distinct genomes, we first verified that the per sample coverage profile for each partition was distinct, and then mapped predicted open reading frames from each onto a database containing proteins from an array of *Bacteroides* species. These results show that one partition was enriched in *Bacteroides massillensis*, and the other in *Bacteroides species X4_1_36* (**Supplementary Fig. 3**). Thus, these partitions show species-level resolution of nearly complete genomes.

Low abundance species are enriched using LSA

The massive net sequencing depth from 176 samples in the FijiCoMP collection (over one trillion bases sequenced) enabled us to detect species present at very low abundance. Initial analysis by 16S sequencing detected over 70 bacterial families at relative abundance levels as low as 4×10^{-6}% (**Supplementary Table 8**). Analysis of marker genes in each partition by MetaPhyler[10] indicated statistically significant ($P < 0.001$) enrichment of microbial families across six orders of magnitude in relative abundance (**Fig. 4** and **Supplementary Table 9**).

Family-specific enrichment was further confirmed in our analysis of aligned assemblies. Between the 344 most specific partitions discussed above, 20 microbial families were represented. These included some families not detected by 16S analysis, and some at relative abundance as low as 0.00001% (**Supplementary Table 6**). We examined the specificity of these partitions by plotting the GC content versus depth of each assembled and aligned contig. For the partitions representing the 15 most enriched families in **Supplementary Table 6**, some are clearly dominated by a single

family, whereas others contain two or three genomes with distinct GC content and depth patterns (**Fig. 5**). Although they have reduced specificity for a single genome, these latter partitions are nonetheless useful for identifying covarying organisms.

***De novo* genome discovery**

We identified 333 FijiCoMP partitions with assembly statistics on a par with those discussed above (total assembly between 500 kbp and 10 Mbp, greater than 28 AMPHORA genes, N50 > 500 bp), but with a greater fraction of sequences that could not be identified in reference databases (>30% of assembly unidentified). Given that all partitions and assemblies were created *de novo*, and that we've primarily restricted our analysis to the family level, it is likely that some of these contain reads from uncharacterized organisms. This is especially true of the FijiCoMP collection, considering that we found that 42.28% of 16S sequences could not be classified at the family level.

We used several filters to look for partitions that were most promising in terms of new genome content. First, we excluded partitions with assembly N50 under 1 kbp, and then we eliminated partitions with total assembly length >10 Mbp. The remaining 28 partitions were sorted by the total fraction of assembly aligning to any reference (**Supplementary Table 10**). Analysis of GC content versus contig depth confirmed that many of these partitions likely represent single genomes that do not align to a common reference (**Supplementary Fig. 4**). The most promising of these had a maximum scaffold of 38 kbp and total assembled length of 2.2 Mbp, of which just 53.4% aligned to any reference genome.

Overall, we consider the prospects for *de novo* genome discovery through LSA to be good. If we relaxed our filter for N50 length and examined alignments at a finer resolution than the family level, we could identify many more partitions containing novel genomes. In particular, our analysis of marker genes by MetaPhyler indicated 427 FijiCoMP partitions in which >50% of marker genes could not be classified at the genus level. However, a more thorough post-assembly pipeline would be needed to validate these as nearly complete, novel genomes.

LSA can recover ecological relationships

Single chromosome genomes were not the only genetic elements that we were able to discover. In the smallest collection we analyzed, which included 18 premature infant gut metagenomes from 11 fecal samples, we found phage sequences partitioned with sequences from their known hosts, and plasmids partitioned in isolation.

We were able to posit phage-host relationships by noting both the 'best hit' alignment to a bacterial reference database, and the 'best hit' alignment to a viral reference database. For example, partitions containing *Enterococcus* genomes preferentially recruited *Enterococcus* phage (though one was found to also contain *Staphylococcus phage CNPH82*) (**Supplementary Table 3**). Although we have not demonstrated that these are extracellular phage, as opposed to integrated prophage, and although the implicit association of being found in the same partition is not enough to establish a definitive biological link between phage and host, the ability of LSA to group co-varying genomes at different absolute abundance levels can reveal relationships that might otherwise go unnoticed.

Additionally, several partitions seemed to contain extrachromosomal sequence aligning to plasmids, and assembled into a total length of >100 kbp. Many of these aligned to previously sequenced *Enterococcus* plasmids. Such isolation of plasmid sequence could make it easier to observe the dynamics of variable elements, and enable enhanced resolution of variable genomic content.

Theory

Metagenomic samples can be viewed as a linear mixing of genomic variables. The observed frequency of every *k*-mer in a sample is a function of the abundance of each distinct DNA fragment containing that *k*-mer. By representing a collection of samples as a *k*-mer-by-sample matrix, we enable the application of well-established matrix decomposition techniques. This method is analogous to latent semantic analysis[23], a technique commonly used in natural language processing to uncover the semantic content in a large corpus of documents.

Our approach is motivated by the following supposition. The variance of a given species' abundance across samples imparts a covariance to the read depth at every *k*-mer in that species' genome. A pooled analysis of multiple samples can capture these patterns, and a linear decomposition of *k*-mer abundances could shed light on latent genomic variables.

In LSA this decomposition is an SVD, which factors a matrix, M, into left singular vectors (L), singular values (V), and right singular vectors (R):

$$M = L \times V \times R^T$$

For an abundance matrix in which there is one row per sample and one column per unique hashed *k*-mer, the right singular vectors can be thought of as principle components (or eigenvectors) of *k*-mer abundance covariance; the singular vector corresponding to the largest singular value is analogous to the first principal component.

We refer to this representation of covarying *k*-mers as an eigengenome, and the columns of *R* collectively as the set of eigengenomes.

The amount of memory consumed in a naive calculation of eigengenomes scales with the size of the *k*-mer abundance matrix (number of samples multiplied by number of *k*-mers). For data such as those in FijiCoMP, this would require terabytes of RAM, which is prohibitive. At the same time, we are motivated to include many samples to observe higher resolution covariance patterns. This tension highlights the value of a streaming implementation, which reduces the memory footprint from terabytes to just a few gigabytes of RAM.

DISCUSSION

Covariance-based analysis of metagenomic data can find meaningful relationships between reads that may not be detectable by assembly or compositional analysis. We have shown here that a hyperplane hashing function and a streaming calculation of eigengenomes facilitates assembly for a range of input data sizes, and that the value of this approach is especially great when data are too large to be analyzed in-memory.

A crucial feature of LSA is that partitioning occurs before assembly. The benefits of this are multiple: assembly is not a bottleneck, because data are first split into much smaller pieces; partitioning can be very sensitive to closely related strains, because sequences are split at the level of ~33-mers before potential merging by assembly; parameters for each assembly, or other pre-assembly analysis, can be adjusted for each individual partition; and partitioning can be sensitive to nonabundant strains or other sequences that are likely to assemble poorly. On the other hand, given a completed assembly, clustering contigs is a much easier computational task for the simple reason that there are typically orders of magnitude more *k*-mers than contigs. This difference is magnified in many assembly and clustering algorithms when small or related contigs are eliminated or merged. For example, the number of filtered contigs clustered in CONCOCT[20] is of the order 10^4, whereas the number of hashed *k*-mers clustered in LSA is of the order 10^9 (**Table 2**). Overall, the differences between LSA and methods such as CONCOCT reflect our optimizations for sensitivity and scalability. Our results, particularly on strain-level separation and abundance sensitivity across six orders of magnitude, validate these optimizations.

For any covariance-based analysis, the number of samples greatly affects partitioning sensitivity. In general, having more samples enables better resolution. This is because the number of potential abundance profiles grows exponentially with the number of samples. As genomes are distinguished by having unique profiles across samples, the upper bound on the number of genomes that can be separated increases with the addition of each new sample. Consider, for example, the case where abundance is replaced by a simple presence/absence measure. In this case, the maximum complexity that can be resolved is $2^{(\text{number of samples})}$. The caveat here is that the samples are independent. Making repeated measurements on the same sample, for instance, may not provide much new information with regards to covariation. Thus, the appropriate number of samples for a given community is a function of both sample complexity and independence. If each new sample is independent, a rough guideline is that the number of samples should increase with the log of community complexity.

We have also shown that an iterative procedure of LSA can be used to adaptively resolve genomes. The increased resolution we observed in subpartitioned reads likely reflects two aspects of the isolated analysis. First, the hash function is recreated from reads within the partition, which can effectively expand *k*-mers that were merged in the original hash function into unique values. Second, subtle differences in

abundance profiles may be more apparent, representing a greater fraction of total variation, when limiting the analysis to a particular subset of reads. In general, some degree of adaptive partitioning is likely necessary for any data set, as every genome will not be distinguished to the same degree. We therefore suggest the following guidelines in partitioning a new data set. First, reads may be initially partitioned at a low to moderate resolution. Then the contents of each partition can be assessed using marker gene analysis (e.g., with MetaPhyler), or by assembly and alignment to a reference database. Some partitions may have clear enrichments for reference genomes, whereas others may have minimal alignments to any reference sequences. For the latter, downstream analysis should focus on characterizing a potentially novel genome. In those partitions containing a mixture of alignments, iterative LSA with adjusted parameters should be used to create an enhanced resolution partitioning. This may be particularly useful for strain-level deconvolution into whole genomes. In these cases, assemblies from a lower-resolution partition could effectively be shared among the adaptively split children, thereby providing the complement to idiosyncratic genome fragments.

The core components of LSA, hyperplane hashing and streaming SVD in particular, could be applied to datasets other than those generated in metagenomics. For instance, our hash function has the desirable property that sequences of greater similarity are more likely to collide. This might be useful in sequence error correction, consensus building, graph building or in search/alignment. One possible application is *de novo* transcriptome assembly. In this context, sequences might be clustered and a graph built from an iterative hashing procedure. Alternatively, if multiple transcriptomic samples are available, individual transcripts might be recapitulated based on covariance information, just as genomes are partitioned here. By adopting the same optimizations that enabled us to run LSA on terabytes of metagenomic data, any of these alternative analyses might be made to scale to equally large data sets in fixed memory. Finally, we consider that advancing sequencing technologies that enable dramatically longer reads may necessitate adapted, or even custom, analyses. For example, although our hyperplane hash function admits arbitrarily long *k*-mers, we would likely want to modify this function to accommodate overlapping short and long read fragments. However, as the scale of data is only likely to increase, the principles and techniques that we have used to achieve fixed memory scaling should remain relevant irrespective of sequence read length.

METHODS
Methods and any associated references are available in the [online version of the paper](online).

Accession codes. SRA: PRJNA217052. Specific samples used for this analysis are listed in **Supplementary Table 11**.

Note: Any Supplementary Information and Source Data files are available in the online version of the paper.

AUTHOR CONTRIBUTIONS
B.C. conceived the algorithm. B.C., I.L.B., K.H., D.G. and E.J.A. designed the experiments. B.C., I.L.B., K.H., T.S. and S.Y. performed the experiments. B.C., I.L.B., K.H., D.G. and E.J.A. wrote the manuscript. All authors reviewed and approved the final manuscript.

1. Fierer, N. *et al.* Metagenomic and small-subunit rRNA analyses reveal the genetic diversity of bacteria, archaea, fungi, and viruses in soil. *Appl. Environ. Microbiol.* **73**, 7059–7066 (2007).
2. Koren, O. *et al.* A guide to enterotypes across the human body: meta-analysis of microbial community structures in human microbiome datasets. *PLoS Comput. Biol.* **9**, e1002863 (2013).
3. Gans, J., Wolinsky, M. & Dunbar, J. Computational improvements reveal great bacterial diversity and high metal toxicity in soil. *Science* **309**, 1387–1390 (2005).
4. Tringe, S.G. *et al.* Comparative metagenomics of microbial communities. *Science* **308**, 554–557 (2005).
5. Daniel, R. The metagenomics of soil. *Nat. Rev. Microbiol.* **3**, 470–478 (2005).
6. Bates, S.T. *et al.* Global biogeography of highly diverse protistan communities in soil. *ISME J.* **7**, 652–659 (2013).
7. Arumugam, M. *et al.* Enterotypes of the human gut microbiome. *Nature* **473**, 174–180 (2011).
8. Thomas, T., Gilbert, J. & Meyer, F. Metagenomics—a guide from sampling to data analysis. *Microb. Inform. Exp.* **2**, 3 (2012).
9. Pop, M. Genome assembly reborn: recent computational challenges. *Brief. Bioinform.* **10**, 354–366 (2009).
10. Treangen, T. *et al.* MetAMOS: a metagenomics assembly and analysis pipeline for AMOS. *Genome Biol.* **12** (suppl. 1), 25 (2011).
11. Namiki, T., Hachiya, T., Tanaka, H. & Sakakibara, Y. MetaVelvet: an extension of Velvet assembler to de novo metagenome assembly from short sequence reads. *Nucleic Acids Res.* **40**, e155 (2012).
12. Peng, Y., Leung, H.C., Yiu, S.M. & Chin, F.Y. Meta-IDBA: a de Novo assembler for metagenomic data. *Bioinformatics* **27**, i94–i101 (2011).
13. Boisvert, S., Raymond, F., Godzaridis, E., Laviolette, F. & Corbeil, J. Ray Meta: scalable *de novo* metagenome assembly and profiling. *Genome Biol.* **13**, R122 (2012).
14. Howe, A.C. *et al.* Tackling soil diversity with the assembly of large, complex metagenomes. *Proc. Natl. Acad. Sci. USA* **111**, 4904–4909 (2014).
15. Pell, J. *et al.* Scaling metagenome sequence assembly with probabilistic de Bruijn graphs. *Proc. Natl. Acad. Sci. USA* **109**, 13272–13277 (2012).
16. Zerbino, D.R. & Birney, E. Velvet: algorithms for *de novo* short read assembly using de Bruijn graphs. *Genome Res.* **18**, 821–829 (2008).
17. Li, D. *et al.* MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics* doi.10.1093/bioinformatics/btv033 (20 January 2015).
18. Sharon, I. *et al.* Time series community genomics analysis reveals rapid shifts in bacterial species, strains, and phage during infant gut colonization. *Genome Res.* **23**, 111–120 (2013).
19. Albertsen, M. *et al.* Genome sequences of rare, uncultured bacteria obtained by differential coverage binning of multiple metagenomes. *Nat. Biotechnol.* **31**, 533–538 (2013).
20. Alneberg, J. *et al.* Binning metagenomic contigs by coverage and composition. *Nat. Methods* **11**, 1144–1146 (2014).
21. Nielsen, H.B. *et al.* Identification and assembly of genomes and genetic elements in complex metagenomic samples without using reference genomes. *Nat. Biotechnol.* **32**, 822–828 (2014).
22. Imelfort, M. *et al.* GroopM: an automated tool for the recovery of population genomes from related metagenomes. *PeerJ* **2**, e603 (2014).
23. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K. & Harshman, R. Indexing by latent semantic analysis. *J. Am. Soc. Inform. Sci.* **41**, 391–407 (1990).
24. Řehůřek, R & Sojka, P. Software framework for topic modelling with large corpora. *Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks* 46–50 (University of Malta, 2010).
25. NIH HMP Working Group. *et al.* The NIH Human Microbiome Project. *Genome Res.* **19**, 2317–2323 (2009).
26. Deloger, M., El Karoui, M. & Petit, M.-A. A genomic distance based on MUM indicates discontinuity between most bacterial species and genera. *J. Bacteriol.* **191**, 91–99 (2009).
27. Morowitz, M.J., Poroyko, V., Caplan, M., Alverdy, J. & Liu, D.C. Redefining the role of intestinal microbes in the pathogenesis of necrotizing enterocolitis. *Pediatrics* **125**, 777–785 (2010).
28. Wu, M. & Eisen, J.A. A simple, fast, and accurate method of phylogenomic inference. *Genome Biol.* **9**, R151 (2008).

ONLINE METHODS

***Salmonella*-spiked HMP samples.** We designed a small pilot data set with 50 samples to demonstrate that LSA could effectively partition short metagenomic reads into biologically relevant clusters. In this data set, synthetic reads from ten finished *Salmonella* genomes were spiked into bona fide metagenomic data, which were subsamples from healthy human gut microbiomes from the Human Microbiome Project[25]. Each mock sample had a total of 40 million Illumina paired-end reads (~100 bp long), of which 2 million (5%) were synthetic reads. Although the percent of synthetic reads were precise in all samples, the presence and abundance of reads from individual *Salmonella* genomes varied randomly across samples. However, the maximum number of genomes that could be present in a sample was limited to three to guarantee a reasonable level of sequencing depth. On average, each genome, while present in a sample, was sequenced at a coverage of 25~50×. See **Supplementary Table 2** for more details.

Salmonella was chosen as a probe for our pilot data set for several reasons: (i) the clade was rarely present in normal human gut microbiome; (ii) the GC content and genome size of this group of organisms were similar to those commonly found in normal gut microbiome; (iii) the strain diversity was better sampled with many finished genomes available, which enabled us to validate the taxonomy resolution of LSA. The ten selected reference genomes included eight strains of *Salmonella enterica* subsp. *enterica servovar* and two *Salmonella bongori*. All of them were obtained from GenBank (*Salmonella enterica* subsp. *arizonae* serovar 62:z4,z23 (*S. ent.* Arizonae; NC_010067). *Salmonella enterica* subsp. *enterica* serovar Schwarzengrund str. CVM19633 (*S. ent.* Schwarzen; NC_011094). *Salmonella enterica* subsp. *enterica* serovar Newport str. USMARC-S3124.1 (*S. ent.* Newport; NC_021902). *Salmonella enterica* subsp. *enterica* serovar Gallinarum/pullorum str. CDC1983-67 (*S. ent.* Gallinarum; NC_022221). *Salmonella enterica* subsp. *enterica* serovar Heidelberg str. CFSAN002069 (*S. ent.* Heidelberg; NC_021812). *Salmonella enterica* subsp. *enterica* serovar Typhimurium str. DT2 (*S. ent.* Typh. (DT2); NC_022544). *Salmonella enterica* subsp. *enterica* serovar Typhimurium SL1344 (*S. ent.* Typh. (1344); NC_016810). *Salmonella enterica* subsp. *enterica* serovar Typhimurium str. LT2 (*S. ent.* Typh. (LT2); NC_003197). *Salmonella bongori* N268-08 (*S. bong.* (N268-08); NC_021870). *Salmonella bongori* NCTC 12419 (*S. bong.* (NCTC 12419); NC_015761). The strain diversity of these genomes was determined based on MUMi scores[26].

Synthetic paired-end reads were generated using maq[29]. The read length and quality distribution were determined by training the program with a bona fide HMP gut microbiome sample.

Data from the infant gut time series were downloaded with the accession SRA052203.

Pipeline steps. The major steps of the partitioning pipeline are shown in **Figure 3**. Most tasks are highly distributable, and all were carried out on an LSF compute cluster. Code for the entire pipeline is publicly available as **Supplementary Code** and at https://github.com/brian-cleary/LatentStrainAnalysis (analysis primarily done with version 57c0121). A table of parameters used to generate the various results is included in the github repository. These include specific parameters for a test data set, our simulations, each of the metagenomic datasets analyzed here, as well as parameters used for iterative partitioning.

Hyperplane hashing. In order to populate a *k*-mer abundance matrix, *k*-length nucleotide sequences were mapped onto a set of columns. Many hash functions could be used here, though most arbitrary functions will produce random collisions in which two or more unrelated *k*-mers map onto the same column. We used a novel locality-sensitive hash function that is both probabilistic, and sequence-sensitive.

To begin, each base from a *k*-mer is mapped onto a complex simplex that incorporates quality scores, and facilitates efficient reverse-complement operations (e.g., A = 1, T = −1, C = *i*, G = −*i*). When each letter is mapped, the *k*-mer is represented in *k*-dimensional complex space. Random projections, or hyperplanes, are drawn through the *k*-dimensional space, creating boundaries that define 'bins' within the space[30,31]. By drawing *n* such hyperplanes, 2^n bins are created, and by numbering each of these bins, a mapping from *k*-mer to

column is established: *k*-mers that fall in the *k*-dimensional space defined by bin b are mapped to column b.

The sequence-sensitive nature of this hash function derives from the fact that nearly identical *k*-mers will be neighbors in *k*-dimensional space, and therefore more likely to fall within the same bin and, thus, column. The probability that two points fall in the same bin is

$$1 - \mathrm{acos(phi)}/\mathrm{pi}$$

where phi is the cosine distance between the length *k* vectors, and is roughly proportional to sequence identity.

k-mers of length 33 were used in each experiment, and the number of random projections was 29 in the small-scale experiment, and 31 in all others. The hashed value of *k*-mers and their reverse-complements are each calculated, though only one is stored to conserve space.

***k*-mer abundance matrix.** Once each *k*-mer in a given sample has been hashed, the number of occurrences of each hashed value is tabulated, and a proxy for the *k*-mer abundance within each sample is established. The full matrix then undergoes a two-step conditioning. First, a local weight is calculated from the total number of *k*-mers in each sample:

$$\mathrm{l}(i) = \mathrm{norm}(i)/\mathrm{sqrt}(h)$$

where norm(*i*) is the Euclidean norm of the abundance vector of sample *i*, and *h* is the hash size. This mitigates the effects of variance in sampling depth.

Second, a global weighting is defined from the distribution of each *k*-mer across the samples. In this case, weighting was based on inverse-document frequency:

$$g(j) = \log_2(N/\mathrm{df}(j))$$

where *N* is the total number of samples, and df(*j*) represent the number of samples in which at least one *k*-mer hashed to column j. Related global measures, such as entropy, could also be used.

The final values in the conditioned *k*-mer abundance matrix are therefore a function of the frequency within each sample, the number of samples in which a hashed *k*-mer occurred, and the total size of each of the initial abundance vectors:

$$c(i,j) = \mathrm{x}(i,j) \star g(j)/\mathrm{l}(i)$$

Streaming SVD. Gensim[24], an open source library for Python, was used to calculate the SVD. Columns from the abundance matrix were streamed off disk in blocks of 200,000 *k*-mers times the number of samples. Because the decomposition is updated iteratively, Gensim allows for parallel and distributed processing of each block. Although this theoretically allows a distribution of the streaming computation across dozens of cores and many machines, in practice we typically used only eight cores on a single host because the process was quickly I/O bound.

***k*-mer clustering in latent space.** Cluster 'seeds' were discovered by an iterative sampling and merging process that avoided loading the entire eigengenome matrix into memory. Small blocks of ~10,000 eigen *k*-mers were randomly selected, then processed in a serial fashion. Each was merged with the most related cluster if they were sufficiently similar, or we created a new cluster if there was no sufficient similarity. The mean of all eigen *k*-mers in each cluster was then calculated, and two clusters merged if they were sufficiently similar. This process was then iterated, sampling 10K eigen *k*-mers and updating the cluster centers each time, until ~0.40% of the eigengenome matrix had been sampled.

The threshold used to merge *k*-mers and *k*-mer clusters has the most direct effect on clustering resolution. Higher thresholds correspond to more granular settings, which result in more *k*-mer clusters, and thus, a higher resolution analysis. More sophisticated methods are needed to computationally discover a 'natural' clustering. Resolution, therefore, is left as a tunable parameter for the user. Our github repository for this project contains a table

("misc/parameters.xlsx") of clustering and other parameters used in various aspects of this study.

After k-mer clustering, the entire set of reads, along with associated hashed k-mers, was then reexamined. Reads were assigned to a partition based on a log-likelihood calculation including the size of each of the k-mer clusters, the intersection of the k-mers in the read with each of the clusters and the global weight of each of the intersecting k-mers. Reads pairs were explicitly assigned to the same partition by considering the union of k-mers from both mates.

Relative abundance and enrichment. Relative bacterial abundances within partitions were estimated using MetaPhyler[32]. This was compared with surveys of the 16S rRNA marker for each sample. These were identified using an open reference-based approach, by aligning to the GreenGenes database[33] and then *de novo* clustering 16S sequences that were not contained within the GreenGenes database.

The *P*-value of the relative abundance within a partition was calculated by assuming a hypergeometric probability of observing enrichment greater than the given value in one partition and a binomial probability of observing such enrichment in any of the partitions.

29. Li, H., Ruan, J. & Durbin, R. Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Res.* **18**, 1851–1858 (2008).
30. Kulis, B. & Grauman, K. Kernelized locality-sensitive hashing for scalable image search. *Proceedings of the IEEE 12th International Conference on Computer Vision* 2130–2137 (October 2009).
31. Gionis, A., Indyk, P. & Motwani, R. Similarity search in high dimensions via hashing. *Proceedings of the 25th International Conference on Very Large Data Bases* (1999).
32. Liu, B., Gibbons, T., Ghodsi, M., Treangen, T. & Pop, M. Accurate and fast estimation of taxonomic profiles from metagenomic shotgun sequences. *BMC Genomics* **12** (suppl. 2), S4 (2011).
33. DeSantis, T.Z. *et al.* Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Appl. Environ. Microbiol.* **72**, 5069–5072 (2006).