Binning metagenomic contigs by coverage and composition

Johannes Alneberg[1,8], Brynjar Smári Bjarnason[1,8], Ino de Bruijn[1,2], Melanie Schirmer[3], Joshua Quick[4,5], Umer Z Ijaz[3], Leo Lahti[6,7], Nicholas J Loman[4], Anders F Andersson[1,9] & Christopher Quince[3,9]

Shotgun sequencing enables the reconstruction of genomes from complex microbial communities, but because assembly does not reconstruct entire genomes, it is necessary to bin genome fragments. Here we present CONCOCT, a new algorithm that combines sequence composition and coverage across multiple samples, to automatically cluster contigs into genomes. We demonstrate high recall and precision on artificial as well as real human gut metagenome data sets.

Metagenomic shotgun sequencing is widely used to investigate the genetic potential and taxonomic composition of microbial communities. It has been used to reconstruct the genomes of individual species from communities[1,2], including samples from complex environments such as sediments and reactors[3]. Repeated sequences within and across genomes, limited coverage, sequencing errors and strain-level variation in gene content all result in fragmented genomes after assembly. A major challenge in metagenomics is therefore the binning of contigs into species-level groups. If closely related reference genomes are lacking, binning has to be conducted in an unsupervised fashion. Numerous automatic binning methods exist that utilize sequence composition alone (the frequency of 'k-mers', or short motifs of length k, characteristic to each genome)[4–7], but a fully automated method that also groups sequences based on correlated coverage across multiple samples has yet to be developed. Existing approaches that do consider coverage require human supervision to distinguish clusters[8,9] and therefore cannot be truly reproducible or scalable.

Here we present CONCOCT, a program that uses Gaussian mixture models to cluster contigs into genomes based on sequence composition and coverage across multiple samples. To determine the number of clusters, we use a variational Bayesian approach[10]. The CONCOCT software is available as **Supplementary Software** and at https://github.com/BinPro/CONCOCT.

To validate CONCOCT, we constructed two synthetic mock metagenome data sets. The species mock was designed to test the ability of CONCOCT to resolve species-level variation in a complex community. It consists of 96 samples, each comprising random paired-end reads from the same 101 species but with different relative frequencies (see Online Methods). The strain mock contains only 20 genomes across 64 samples, but 5 genomes are from different strains of *Escherichia coli*; it was constructed to investigate the impact of strain-level variation on clustering (see **Supplementary Tables 1** and **2** for lists of genome sequences used). For both data sets, frequencies were taken from true abundances of organisms in human fecal samples determined by 16S rRNA sequencing as part of the Human Microbiome Project[11] (**Supplementary Fig. 1**). Reads from all samples were coassembled into contigs (see **Supplementary Table 3** for assembly statistics), and large contigs (>10,000 base pairs (bp)) were fragmented to ensure that they were given more statistical weight; this also reduced the impact of chimeric assemblies. The reads were then mapped back onto the contigs to determine coverage in each sample (Online Methods). These coverages reflect the abundances of the underlying organisms across samples and can thus be used to disentangle which contig derives from which genome.

Sequence composition can also be used to bin each contig, as different organisms have different characteristic signatures of k-mers[4–7,12,13]. To incorporate both composition and coverage into our model, we generated a combined profile for each contig, joining the coverage and composition vectors (Online Methods). We then applied principal-component analysis (PCA) to reduce the high dimensionality of this matrix, keeping enough dimensions D to maintain 90% of the information. When contigs from the synthetic communities were projected along the first two PCA dimensions, species formed distinct ellipsoid clusters (**Supplementary Figs. 2** and **3**). We therefore decided to describe each cluster by a D-dimensional Gaussian distribution with a full covariance matrix. A Gaussian mixture model (GMM) describes the whole data set by the weighted sum of K such Gaussians. The optimal cluster number is determined by automatic relevance determination[10]. Starting from a large number of clusters, the program selects only those necessary to explain the data.

Applying CONCOCT to the species mock containing 101 genomes, we predicted 101 clusters (Online Methods, **Supplementary Figs. 4** and **5** and **Supplementary Table 4**). The precision (purity of the clusters) was 0.988, so that on average a

[1] KTH Royal Institute of Technology, Science for Life Laboratory, School of Biotechnology, Division of Gene Technology, Stockholm, Sweden. [2] Bioinformatics Infrastructure for Life Sciences (BILS), Stockholm, Sweden. [3] School of Engineering, University of Glasgow, Glasgow, UK. [4] Institute of Microbiology and Infection, University of Birmingham, Birmingham, UK. [5] National Institute for Health Research Surgical Reconstruction (NIHR) Surgical Reconstruction and Microbiology Research Centre, University of Birmingham, UK. [6] Department of Veterinary Biosciences, University of Helsinki, Helsinki, Finland. [7] Laboratory of Microbiology, Wageningen University, Wageningen, the Netherlands. [8] These authors contributed equally to this work. [9] These authors jointly directed this work. Correspondence should be addressed to A.F.A. (anders.andersson@scilifelab.se) or C.Q. (christopher.quince@glasgow.ac.uk).

RECEIVED 2 DECEMBER 2013; ACCEPTED 18 AUGUST 2014; PUBLISHED ONLINE 14 SEPTEMBER 2014; DOI:10.1038/NMETH.3103

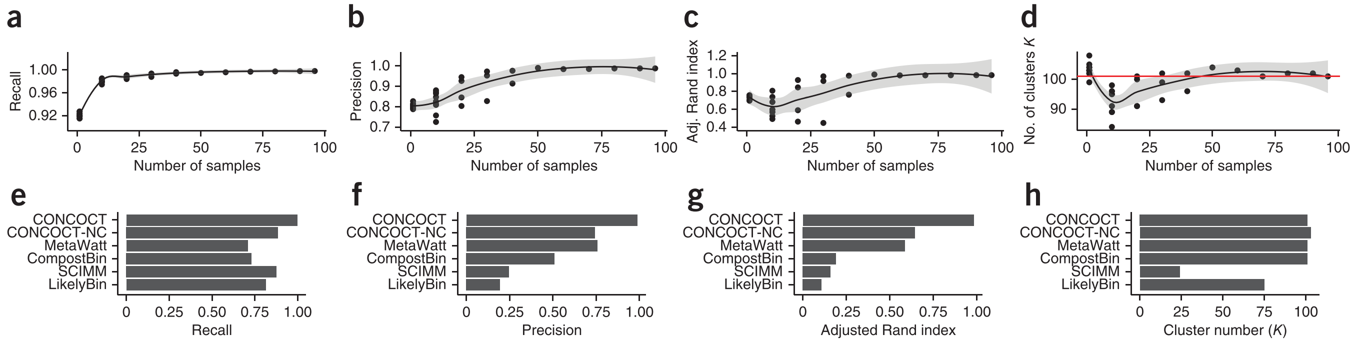

Figure 1 | Performance of CONCOCT and other unsupervised binning methods on a synthetic community. (**a**–**d**) CONCOCT was run on the synthetic species mock community with variable sample numbers to determine recall (**a**), precision (**b**), adjusted Rand index (**c**) and optimal number of clusters, *K* (**d**) (see Online Methods). Solid lines represent Loess smoothing, and gray shaded portions represent standard errors in the local polynomial regression predictions. (**e**–**h**) Performance of clustering algorithms CONCOCT, CONCOCT-NC (without coverage information), MetaWatt[5], CompostBin[6], SCIMM[7] and LikelyBin[4]. Results are given for recall (**e**), precision (**f**), adjusted Rand index (**g**) and number of clusters predicted (**h**). The algorithms are ordered based on adjusted Rand index.

cluster almost entirely contains contigs from the same species. Similarly, the recall (proportion of each species binned to the same cluster) was nearly perfect at 0.998. For the adjusted Rand index, which summarizes both precision and recall, we obtained 0.983 for the species mock. For individual contigs, the error probability decreased significantly with length (logistic regression – *P* value $< 2.0 \times 10^{-16}$; see **Supplementary Fig. 6**).

To illustrate how much information coverage adds and to determine how many samples are necessary to resolve a data set of the complexity of the species mock, we ran CONCOCT with different sample numbers (**Fig. 1**). The overall accuracy started to decrease below 50 samples (**Fig. 1c**), mostly due to a loss of precision (**Fig. 1b**), which in turn was due to the species number being underestimated (**Fig. 1d**). At small sample number the situation was reversed and the number of genomes was overestimated, which then affected the recall (**Fig. 1a**). We also compared CONCOCT to four binning methods that only use composition information on the species mock (**Fig.1e–h** and **Supplementary Table 5**). The best was MetaWatt, with an adjusted Rand index of 0.588, lower than those for CONCOCT (0.983) and even CONCOCT without coverage (CONCOCT-NC: 0.646). For real data sets, we cannot know the true assignment of contigs to genomes, so we also used a method of evaluating clusters based on 36 single-copy core genes (SCGs) that are found once in almost all known bacterial genomes (**Supplementary Table 6** and Online Methods). Of the clusters generated by CONCOCT from the species mock, the vast majority were complete in pure clusters; 93 of 101 clusters had >90% of SCGs present in single copies, 1 (cluster 53) had five SCGs in multiple copies and the rest were incomplete (**Supplementary Fig. 7**).

On the strain mock containing 20 genomes, CONCOCT predicted 21 clusters (**Supplementary Figs. 8** and **9**), for an adjusted Rand index of 0.945 and precision of 0.942. The lower precision with respect to the species mock is because the five *E. coli* strains were separated into only two clusters, one comprising the most abundant strain, K12, and the other containing contigs from the other strains. As a final validation, we applied CONCOCT to the relatively simple data set of Sharon *et al.*[8], comprising a time series of 11 fecal samples from the first month of a preterm infant's life (**Supplementary Figs. 10** and **11**). We extracted six pure and complete genomes, results essentially identical to those of the original study, which used an extended self-organizing map (ESOM) for visualization followed by manual definition of clusters. In contrast, the four composition-based clustering methods performed poorly on the strain mock data set (see **Supplementary Table 5** for validation statistics) and either fragmented or collapsed some of the genomes in the Sharon *et al.* data set (**Supplementary Fig. 12**). We conclude that CONCOCT performs well at reconstructing species from complex communities and can resolve some but not all strain-level variation. The limitation probably arises from the assembly itself; if there are insufficient contigs unique to a strain, then the software will not distinguish it as a distinct cluster.

We next applied CONCOCT to a real data set of over 3 billion reads from 53 fecal samples taken from individuals during the 2011 outbreak of Shiga toxin–producing *E. coli* (STEC) O104:H4 (ref. 14). 43 samples are from individuals who tested positive for STEC by PCR, and 10 samples were diagnosed as containing unrelated pathogens (Online Methods). Following coassembly of these reads, we obtained 142,723 contigs longer than 1,000 bp. We could classify 24,465 (17%) of these to the species level by matching to the NCBI whole-genome database using TAXAassign (Online Methods), from which we observed 134 species as compared to the 297 clusters generated by CONCOCT (Online Methods; **Supplementary Table 4**; **Supplementary Fig. 13**). The classified contigs gave a high precision of 0.94 and a somewhat lower recall of 0.82, but without complete and accurate assignments these statistics should be treated with caution. We also calculated the frequencies of the SCGs in each cluster. There are some chimeric clusters, but CONCOCT succeeded in generating 15 clusters with at least 75% of the SCGs in a single copy (**Supplementary Fig. 14**), representing pure and mostly complete genomes.

An important use of metagenomic data in pathogen discovery is the reconstruction of entire pathogen genomes, without requiring a reference sequence. Expecting that the *E. coli* outbreak genome will be more abundant in recently infected individuals, we correlated the time since onset of diarrhea, ddays, with the mean log-coverage profiles for clusters in the 43 STEC samples (**Supplementary Table 7**). Eight correlated clusters were found with a false discovery rate of less than 10% (**Fig. 2**). Clusters 83 and 122 were strongly negatively correlated with ddays and are predominately made up of *E. coli* contigs, making them candidates for the outbreak genome. To test this, we computed the number of contigs from each cluster that map either to strain-specific

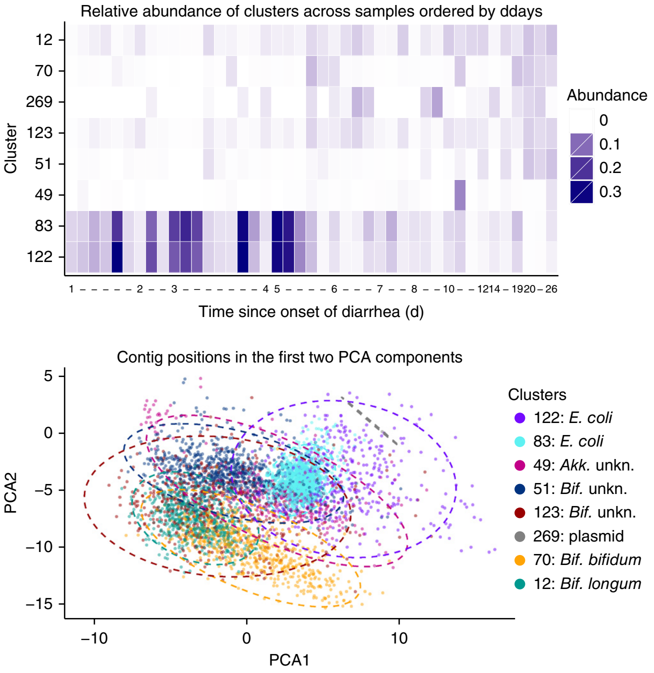

Figure 2 | Analysis of the 2011 STEC *Escherichia coli* 0104:H4 outbreak. Top, heat map showing the relative abundance of the eight clusters that correlated significantly with days since onset of diarrhea (ddays) in each of the 43 STEC-positive samples. The STEC samples have been ordered by ddays; '–' indicates samples with a ddays value to the left of the dash. The clusters are ordered bottom to top from negative to increasingly positive correlation coefficients (see **Supplementary Table 7**). Relative frequencies have been square-root transformed to reveal more information at low abundance. Bottom, the same seven clusters plotted in the first two PCA components of the transformed combined log coverage composition profiles. *Akk.*, *Akkermansia*; *Bif.*, *Bifidobacterium*.

portions of the known *E. coli* O104:H4 genome or to core *E. coli* genes (**Supplementary Table 8**). In total, 91.1% of the mapped contigs derive from clusters 83 and 122, which together represent the vast majority of the outbreak genome. Many more contigs mapping to the *E. coli* core genome derive from cluster 83, and a higher proportion of outbreak-specific contigs derive from cluster 122 (**Supplementary Fig. 15**), probably reflecting differences in coverage due to the presence of non-outbreak *E. coli* strains. The single-outbreak genome was thus best described by two components, which also have different variances (**Fig. 2**, lower panel).

We have focused here on the outbreak *E. coli* clusters, but the clustering approach of CONCOCT facilitates whole-community analysis. The PCA will reflect key patterns in the data; in this case, the first two PCA components defined the outbreak. As clusters move from negative to positive correlation with ddays, their mean values for both components tend to become more negative, although the small cluster 269 was an exception (**Fig. 2**). All the significant clusters other than those representing the two *E. coli* strains were positively correlated with ddays ($P < 5.0 \times 10^{-3}$). These represent organisms that may be important in the recovery from STEC infection, with a majority assigned to *Bifidobacterium* species. The strongest associations were with *Bifidobacterium longum* (cluster 12) and *Bifidobacterium bifidum* (cluster 70). Two clusters may be new species from the genus *Bifidobacterium*. In fact, of the positively associated clusters, only cluster 49, which has no species assignments but is classified in genus *Akkermansia*, and cluster 269, which is very small, most likely a plasmid, are not classified as *Bifidobacterium*. The ability of bifidobacteria to protect against STEC has been directly demonstrated in mice[15]. Our findings indicate that they may also be important for recovery from infection in humans and illustrates the power of CONCOCT to extract biologically relevant clusters directly from metagenomics data. Currently this approach may be limited to environments of low to medium diversity where the necessary coassembly can be performed. But for highly complex communities such as soil, new bioinformatics approaches may make assembly tractable[16].

METHODS

Methods and any associated references are available in the online version of the paper.

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

ACKNOWLEDGMENTS

This research arose out of a workshop funded through the COST project ES1103 and hosted by P. Fernandes at the Instituto Gulbenkian de Ciência. This work was funded by grants (to A.F.A.) from the Swedish Research Councils VR (grant 2011-5689), FORMAS (grant 2009-1174) and EC BONUS project BLUEPRINT. C.Q. is funded by an EPSRC Career Acceleration Fellowship—EP/H003851/1. M.S. is supported by Unilever R&D Port Sunlight, Bebington, UK. L.L. is supported by the Academy of Finland (grant 256950), N.L. by a UK Medical Research Council Special Training Fellowship in Biomedical Informatics and J.Q. by the UK National Institute for Health Research (NIHR) Centre for Surgical Reconstruction and Microbiology. This paper presents independent research funded by the NIHR Surgical Reconstruction and Microbiology Research Centre (partnership between University Hospitals Birmingham National Health Service (NHS) Foundation Trust, the University of Birmingham and the Royal Centre for Defence Medicine). The views expressed are those of the author(s) and not necessarily those of the NHS, the NIHR or the Department of Health.

AUTHOR CONTRIBUTIONS

C.Q. developed the core algorithm and cluster validation metrics and performed analyses. A.F.A. assisted with the analyses, developed the SCG validation and contributed to algorithm development. J.A. and B.S.B. developed the CONCOCT software pipeline and contributed to algorithm development. I.d.B. performed assemblies and mappings. M.S. generated simulation data. J.Q. performed *E. coli* mappings. U.Z.I. assisted with SCG validation and production of graphics. L.L. helped with graphics and algorithm design. N.J.L. performed *E. coli* analysis and contributed to algorithm development. All authors contributed to the writing of the manuscript.

COMPETING FINANCIAL INTERESTS

The authors declare no competing financial interests.

1. Tyson, G.W. *et al. Nature* **428**, 37–43 (2004).
2. Herlemann, D.P. *et al. MBio* **4**, e00569–e00512 (2013).
3. Sharon, I. & Banfield, J.F. *Science* **342**, 1057–1058 (2013).
4. Kislyuk, A., Bhatnagar, S., Dushoff, J. & Weitz, J.S. *BMC Bioinformatics* **10**, 316 (2009).
5. Strous, M., Kraft, B., Bisdorf, R. & Tegetmeyer, H. *Front. Microbiol.* **3**, 410 (2012).
6. Chatterji, S., Yamazaki, I., Bai, Z. & Eisen, J.A. in *Res. Comput. Mol. Biol.* (eds. Vingron, M. & Wong, L.) 17–28 (Springer, 2008).
7. Kelley, D.R. & Salzberg, S.L. *BMC Bioinformatics* **11**, 544 (2010).
8. Sharon, I. *et al. Genome Res.* **23**, 111–120 (2013).
9. Albertsen, M. *et al. Nat. Biotechnol.* **31**, 533–538 (2013).
10. Corduneanu, A. & Bishop, C.M. in *Artif. Intell. Stat. 2001* (eds. Jaakkola, T. & Richardson, T.) 27–34 (Morgan Kaufmann, 2001).
11. Human Microbiome Project Consortium. *Nature* **486**, 207–214 (2012.).
12. Sandberg, R. *et al. Genome Res.* **11**, 1404–1409 (2001).
13. Dick, G.J. *et al. Genome Biol.* **10**, R85 (2009).
14. Loman, N.J. *et al. J. Am. Med. Assoc.* **309**, 1502–1510 (2013).
15. Asahara, T. *et al. Infect. Immun.* **72**, 2240–2247 (2004).
16. Pell, J. *et al. Proc. Natl. Acad. Sci. USA* **109**, 13272–13277 (2012).

ONLINE METHODS

Synthetic data sets. We based the simulations of the mock data sets on 16S rRNA samples from the Human Microbiome Project (HMP)[11]. The samples were de-noised with the AmpliconNoise pipeline and operational taxonomic units (OTUs) constructed at 3% sequence difference to approximate species. This generated a total of 6,839 OTUs with known relative abundance profiles across the 187 samples.

The first species mock community reflects a complex population containing 101 species but without strain-level variation. We based this simulation on the first 96 data sets from the HMP project. After filtering out OTUs with a total of less than 20 counts summed across samples, we used BLAST to match the remaining 16S rRNA OTU sequences against the NCBI whole-genome database. We were able to identify distinct organisms for 106 OTUs with a greater than 80% nucleotide identity. We then removed multiple strains from the same species, and for each of the resulting 101 organisms in our mock database we compiled all chromosomes and plasmids available on the NCBI whole-genome database. Unknown nucleotides (N) and ambiguous nucleotides were deleted from the sequences for the purpose of the simulation. See **Supplementary Table 1** for details of the genomes used. For the frequency distribution of each genome, we used the OTU that it mapped to if that was one of the 101 most abundant OTUs. This occurred for 21 genomes. The others were assigned the distribution of one of the remaining 101 most abundant OTUs that did not have a matching genome.

The second strain mock data set was designed to test the ability of CONCOCT to resolve different levels of taxonomic resolution, including strains of the same species. The mock community comprises five different *E. coli* strains, five Bacteroides species, five species from different Clostridium genera plus five other typical gut bacteria. See **Supplementary Table 2** for details. The frequency distributions for the simulation were adapted from the first 64 HMP samples restricted to the 20 most abundant organisms across all HMP samples.

The number of simulated reads corresponds to the yield of approximately 1/2 flow cell (4 lanes) of an Illumina HiSeq 2500 high output run for each mock community. We assumed that each lane yields 188 million read pairs, so for each of our 96 samples in the species mock we simulated 7.75 million paired-end reads and for the strain mock we simulated 11.75 million paired-end reads for each of the 64 samples. Reads were generated by sampling randomly across the genomes present in a sample according to their relative abundance and then sampling the starting position uniformly at random within the selected genome. Our read simulation program utilizes position- and nucleotide-specific substitution, insertion and deletion patterns. We inferred these error profiles from a real data set, where a diverse *in vitro* mock community of known genomes was prepared and sequenced on a HiSeq 2500 following TruSeq library preparation. Since the genomes were known, error profiles could be inferred by mapping reads back to the genomes, and calculating position and base specific error rates. We also inferred the fragment size distribution from this data set and used it for the simulation of the paired-end 2 × 100 bp reads. The read simulation program outputs reads in fasta format, which we converted into a pseudo fastq format for the downstream analysis assuming uniform quality scores. The read names contain information on the genome from which the respective read originated for the subsequent validation of the CONCOCT algorithm.

Sharon *et al.*[8] data. To provide a simple real metagenome data set to test CONCOCT, we applied it to a time-series of 11 fecal samples from the first month of a preterm infant's life[8]. The reads were downloaded from the NCBI short-read archive (SRA052203). We used the reads for all 18 libraries available for the coassembly and CONCOCT, but these represent only 11 samples, since 7 were resequenced as for these samples the first run did not provide enough data (I. Sharon, University of California, Berkeley, pers. comm.). This data set comprised a total of 1.372×10^8 100-bp reads.

***E. coli* O104:H4 outbreak.** This data consisted of over 3 billion 150-bp reads filtered for human DNA from 53 metagenomic data sets from individuals presenting during the 2011 outbreak of Shiga toxin–producing *Escherichia coli* (STEC) O104:H4 (ref. 14) (ERP001956). 43 of these samples were from STEC infected individuals and the remainder from patients with clinical diagnoses of *Clostridium difficile*, *Salmonella enterica* or *Campylobacter jejuni*.

Coassembly and mapping. For both the simulated and real data, a coassembly of reads from all samples was performed. Ray version 2.3.1 (ref. 17) was used to generate the coassemblies because it is able to handle large data sets by distributing the computation over multiple nodes and is specifically designed to handle metagenomics data. For all assemblies a *k*-mer length of 41 was used. The strain and species mocks, both with around 750 million read pairs, were assembled on a Cray XE6 system using 2,048 cores for 22.5 h and 4,096 cores for 12.5 h, respectively. The *E. coli* data was assembled on the same machine in 2 h on 2,048 cores. The smaller Sharon data set was able to complete within 1 d and 3 h using 16 cores on a CPU with a dual Intel Xeon E5-2660 and 512 GB of RAM. Contigs were cut up into non-overlapping fragments of 10 kilobases in order to (1) give more weight to large contigs and (2) mitigate the effect of local assembly errors by having the potential to separate erroneous parts during clustering. The final fragment is appended to the one before if it is shorter than 10 kilobases to avoid generating any fragments less than 10 kilobases. The fragment size was chosen as to have the aforementioned advantages without overly increasing the computational costs. For convenience, in all the analysis that follows, we generally refer to these contig fragments simply as contigs. The coassembly statistics are given in **Supplementary Table 3**.

To determine coverage of the coassembly per sample, the reads were mapped back with Bowtie2 version 2.1.017 using default parameters. Bowtie2 outputs the overall alignment ratio, i.e., the number of reads aligning to the assembly divided by the total number of reads, which gives an indication of how much of the sequencing data is kept. These ratios are given in **Supplementary Table 3**. MarkDuplicates from Picard Tools v1.77 was subsequently used to remove duplicate reads. The duplication rate determined was below 1% for all data sets. After removing the duplicates, coverages were computed from the bam files with BEDTools v2.17.0. For the strain mock we observed a large number of duplicate contigs from *Faecalibacterium prausnitzii*, an error in the assembly process that was not observed in any of the other data sets. We therefore identified identical contigs by aligning the assembly against itself using MUMmer 3.1 and removed those.

doi:10.1038/nmeth.3103

For the synthetic communities we know which genome generated any given read, and hence for each contig we could determine the number of reads mapping to it from each genome. This script can also be found in CONCOCT: contig_read_count_per_genome.py. To label a contig we insisted that at least 10 reads had to map unambiguously to that contig; we then assigned the contig to the genome that the majority of those reads mapped from, provided at least 50% of the reads derived from that genome. This left a small proportion of ambiguous contigs unlabeled. The *E. coli* and Sharon contigs were taxonomically classified by searching against the NCBI nr/nt database using TAXAassign v0.4 (https://github.com/umerijaz/taxaassign). This program uses BLAST to search for matches from the NCBI nucleotide database that are within a given identity and query coverage. Taxonomy is then assigned based on the top *n* hits that match these criteria using a consensus approach, i.e., we assign at a given level if at least 90% of the hits at that level have the same taxa. We only included hits with 95% identity and 90% query coverage, and used the top 100 sequences. These values were chosen to ensure a sufficient stringency that we could be confident of species level assignments.

We recommend that a coassembly be performed before running CONCOCT. This maximizes the number of genomes that can potentially be resolved by the clustering process. However, if coassembly proves computationally prohibitive, then it is possible to use single sample assemblies. Mapping then proceeds as above from all samples and the per sample coverage information can still be used to cluster contigs within the focal sample.

Preprocessing. Prior to preprocessing, contigs are filtered by length, and only contigs greater than a minimum size are used. This is an adjustable parameter but was fixed at 1,000 bp throughout this study. Each filtered contig indexed $i = 1,\ldots,N$ is represented by a coverage vector and a composition vector. The coverage vector is simply the average number of reads per base from each of M samples, indexed $j = 1,.,M$, mapping to that contig. We will denote the coverage vector for each contig by $\mathbf{Y_i} = (Y_{i,1},\ldots,Y_{i,M})$. The composition vector contains the frequency for each k-mer and its reverse complement in that contig. In all the results presented here a k-mer frequency of 4 was used although CONCOCT can accept any k-mer length as a parameter. The frequencies are combined with complements since sequencing is bidirectional. The dimension V of the composition is 136 for tetramers due to palindromic k-mers. We denote the composition vector for each contig by $\mathbf{Z_i} = (Z_{i,1},\ldots,Z_{i,V})$. Prior to normalization we added a small pseudo-count to both coverage and composition vectors. This removes nonzero entries, necessary to allow the log-transform below. It is equivalent to assuming a uniform Dirichlet distribution prior on the relative frequencies. For the composition, we simply add a single count to each k-mer $Z'_{i,j} = Z_{i,j} + 1$, but for the coverage we imagine mapping an extra read of length 100 bp to each contig in each sample $Y'_{i,j} = Y_{i,j} + 100/L_i$, where L_i is the contig length.

Coverage vectors are normalized, first over samples, so that different read numbers from a sample are accounted for

$$Y''_{i,j} = \frac{Y'_{i,j}}{\sum\limits_{k=1}^{N} Y'_{k,j}}$$

and over contigs to give coverage profiles **p**. This normalizes for coverage variation within a genome, ensuring that this does not mask co-occurrence of contigs

$$p_{i,j} = \frac{Y''_{i,j}}{\sum\limits_{k=1}^{M} Y''_{i,k}}$$

However, the total coverage does contain further information that may potentially discriminate organisms, so we keep this as an additional variable

$$Y''_{i,.} = \sum_{k=1}^{N} Y''_{i,k}$$

We also normalize composition to give composition profiles **q**. This accounts for different contig lengths

$$q_{i,j} = \frac{Z'_{i,j}}{\sum\limits_{k=1}^{V} Z'_{i,k}}$$

These two vectors and the total coverage are joined together and log-transformed to give a combined log profile

$$x_i = [\log(q_{i,1}),\ldots,\log(q_{i,V}),\log(p_{i,1}),\ldots,\log(p_{i,M}),\log(Y''_{i,.})]$$

of dimension $E = M + V + 1$. This vector then represents both coverage and composition; the transform expands the domain of the normalized variables to the negative half-space, and gives improved results over not transforming or alternatives such as the square root. Finally, a dimensionality reduction using principal-component analysis, implemented as a singular value decomposition, was performed on the $N \times E$ matrix of log-profiles $\mathbf{X}$ with rows corresponding to the vectors $\mathbf{x_i}$ and thus elements $x_{i,j}$. The number of components, D, necessary to explain 90% of the variance in the data were kept. This reduces the dimensionality of the clustering problem while keeping the majority of the coverage and composition information. We will denote the transformed data $N \times D$ matrix by $\mathbf{V}$ with row vectors $\mathbf{v_i}$ for each contig i.

To cluster contigs into bins we use a Gaussian mixture model fit with a variational Bayesian approximation (see **Supplementary Note**).

CONCOCT implementation. This algorithm was implemented in C but as a Python module. To run the main CONCOCT program, the transformation and clustering took a real-time of 37 m 00.88 s for the synthetic species mock community, 3 m 43.558 s for the strain mock and 2,132 m for the *E. coli* data set using 10 cores of a 64-core, 512-GB-RAM Xeon E5 compute server. A high-memory server was used here, but the memory usage of the algorithm is quite modest: even for the largest *E. coli* data set, less than 10% of the system memory was required.

Evaluating clusterings by comparison to known genome assignments. As discussed above, for the synthetic communities we know the genome that the vast majority of contigs derive from; we can view these as class labels. These class assignments represent the idealized grouping; a perfect clustering would predict a K equal to the number of classes or genomes, i.e., 101 or 20 for the species and strain mocks respectively, and with each cluster purely composed of contigs from one of the genomes. For the Sharon

(2013) and *E. coli* O104:H4 outbreak data we do not know the true species assignments for the majority of contigs but we do for those that we could unambiguously assign with the TAXAAssign script—3,167 out of 5,571 contigs with length >1,000 bp for the Sharon (2013) data and 24,465 out of 142,723 for the *E. coli* outbreak; we can use these labeled contigs for evaluation there, too. In reality, we will never obtain a perfect clustering, so we therefore need a statistic to determine how far from that perfect grouping a clustering is. For this the Rand index is an intuitive solution. It considers pairs of elements: if a pair of elements deriving from the same class are placed in the same cluster then this is considered a true positive; denote the number of such pairs as *TP*. Similarly, if a pair of elements deriving from different classes are placed in different clusters, then this is considered a true negative, *TN*. The Rand index, lying between 0 and 1, is simply the number of correct pairs, *TN* + *TP*, divided by the total number of pairs

$$\binom{N}{2} = \frac{N.(N-1)}{2}$$

where we have used the binomial coefficient notation. If we define a $K \times S$ matrix $\mathbf{n}$ with elements $n_{i,j}$ that are the number of assignments to the ith cluster and jth class then:

$$TP = \sum_{i,j} n_{i,j}(n_{i,j}-1)/2$$

The quantity *TN* is harder to calculate but it can be shown to be

$$TN = \binom{N}{2} + \sum_{i,j}\binom{n_{i,j}}{2} - \sum_{i}\binom{n_{i,.}}{2} - \sum_{j}\binom{n_{.,j}}{2}$$

where

$$n_{.,j} = \sum_i n_{i,j} \quad \text{and} \quad n_{i,.} = \sum_j n_{i,j}$$

Therefore the Rand index is

$$RI = \frac{\binom{N}{2} + 2\sum_{i,j}\binom{n_{i,j}}{2} - \sum_{i}\binom{n_{i,.}}{2} - \sum_{j}\binom{n_{.,j}}{2}}{\binom{N}{2}}$$

However, given a random classification and a random clustering, we would expect a nonzero Rand index just by chance. The adjusted Rand index accounts for this by subtracting the expected value given fixed class and cluster sizes and normalizing so that values are still smaller than equal to 1, which indicates a perfect clustering. This can be simplified to

$$ARI = \frac{\sum_{i,j}\binom{n_{i,j}}{2} - \frac{\sum_i \binom{n_{i,.}}{2}\sum_j\binom{n_{.,j}}{2}}{\binom{N}{2}}}{\frac{1}{2}\left[\sum_i\binom{n_{i,.}}{2} + \sum_j\binom{n_{.,j}}{2}\right] - \frac{\sum_i\binom{n_{i,.}}{2}\sum_j\binom{n_{.,j}}{2}}{\binom{N}{2}}}$$

In addition to the adjusted Rand index, we used two further measures that help us to understand in what way a clustering is deviating from the classification. The first is the recall: here we calculate the number of contigs from each species or class that are placed in the same cluster, sum over all classes and divide by *N*. This indicates how complete the genome bins are. The second is the precision, which inversely is calculated by summing the contigs in each cluster that derive from the same species and dividing by *N*; this indicates how pure the clusters are. These are standard statistics, but we adapted them to give more weight to correctly clustered long contigs rather than short ones; this was done by weighting each contig by a factor proportional to its length. This is simple to do: each contig is effectively treated as a set of replicate data points with the number of replicates equal to its length in base pairs. These statistics are generated by the script Validate.pl, distributed in CONCOCT, from the cluster and class assignments and optionally the contig sequences if the length weighting is to be used.

Impact of sample numbers. To explore the impact of sample number on the performance of CONCOCT for the 96 sample species mock, we reran the algorithm after first sub-setting and only using a fraction of the per sample coverages. This was done for 1, 10, 20, 30, 40, 50, 60, 70, 80 and 90 samples. Multiple replicates of up to 10 contiguous sample subsets were run but only where overlapping could be avoided; this allowed 10, 10, 4, 3, 2, 1, 1, 1 and 1 replicates, respectively. The results are shown in **Figure 1**.

Evaluating clusters with single-copy core genes (SCGs). Evaluating clusters with single-copy core genes is an alternative way of evaluating cluster completeness and purity. We utilize housekeeping genes that typically occur in single copies in microbial genomes. To select appropriate genes we downloaded all complete microbial genomes from NCBI (25 August 2013) and selected one random genome for each genus. The list of genomes can be downloaded from https://github.com/BinPro/CONCOCT/blob/master/scgs/gen_scg.txt. We counted the frequency of each Cluster of Orthologous Groups (COGs)[18] of genes in these 525 genomes. Since there are very few COGs that occur once in every genome we instead applied the more relaxed criteria of being present in greater than 97.0% of the genomes and having an average frequency of less than 1.03. This resulted in 36 COGs. Twenty-seven of these are shared with the list of 40 COGs that was selected in a similar way in an earlier study[19]. We then used Prodigal for gene recognition and translation followed by RPS-BLAST of these sequences against the NCBI COG database using an e-value cut-off of 1.0e-3. Each query was assigned to the top RPS-BLAST and only if it covered at least 50% of the target sequence. The script COG_table.py was then used to generate a table of counts for these COGs (or another list of genes provided by the user) within the different clusters output by CONCOCT. The 36 selected COGs are given in **Supplementary Table 6**.

Comparison to existing contig clustering algorithms. In order to place these results into context, we compared CONCOCT to four published programs that conduct clustering of metagenomic contigs in an entirely unsupervised fashion: LikelyBin[4], MetaWatt[5], CompostBin[6] and SCIMM[7]. All four programs purely utilize sequence composition: LikelyBin uses second- to fourth-order Markov models, MetaWatt and SCIMM use interpolated (variable-order) Markov models to model sequences, and CompostBin uses a PCA to reduce dimensionality followed by the normalized cut clustering algorithm. We ran these programs on the species and strain mocks and the Sharon *et al.*[8] data set. Not all algorithms predict cluster number, so in order to generate the best possible results for these programs, we set the cluster number to the number of genomes in the synthetic communities, i.e., 101 or 20 for the species and strain mock, respectively, or the number of clusters predicted by CONCOCT for the Sharon (2013) data, although not all clusters then ended up with nonzero memberships. LikelyBin was run using default parameters (using third-order Markov models), and SCIMM was run initializing the IMMs with LikelyBin. For MetaWatt we only employed the first three unsupervised steps. We chose the minimum bin size such that the number of resulting bins was equal to the number of genomes for the synthetic mocks and equal to the number of predicted clusters by CONCOCT for the Sharon data. We also included CONCOCT and CONCOCT with only sequence composition in this comparison (CONCOCT-NC). The generated clusters were then compared to known species assignments and precision, recall and adjusted Rand indices calculated. MetaWatt does not attempt to cluster all contigs leaving some unbinned, these were removed from the validation. The results are given in **Supplementary Table 5** and **Figure 1**.

Linking cluster abundances to environmental variables. Comparing the abundances of clusters across samples to measured environmental variables is a powerful method for determining more about the potential importance or function of the different clusters. The first step is to calculate the average coverages for each cluster across the samples. We provide the script ClusterMeanCov.pl in the CONCOCT package, which does this accounting for contig length in the average. In the *E. coli* STEC outbreak example, we then normalized these coverages by total coverage per sample, as in the preprocessing step, and log-transformed before performing the Pearson's correlations given in **Supplementary Table 7**.

17. Boisvert, S., Raymond, F., Godzaridis, E., Laviolette, F. & Corbeil, J. *Genome Biol.* **13**, R122 (2012).
18. Tatusov, R.L., Koonin, E.V. & Lipman, D.J. *Science* **278**, 631–637 (1997).
19. Ciccarelli, F.D. *et al. Science* **311**, 1283–1287 (2006).

doi:10.1038/nmeth.3103