Interpreting 16S metagenomic data without clustering to achieve sub-OTU resolution

Supplementary information

Mikhail Tikhonov,[1,2] Robert W. Leach,[2] and Ned S. Wingreen[2,3]

[1] *Joseph Henry Laboratories of Physics*
[2] *Lewis-Sigler Institute for Integrative Genomics*
[3] *Department of Molecular Biology, Princeton University, Princeton NJ 08540.*

Contents

A Supplementary methods. Cluster-free filtering: details and applications

In this section, we illustrate the idea of error-model-based denoising (see also the introduction in Rosen et al., 2012) and give a detailed description of the simple denoiser we designed for this work. We then describe the workflow of an open source software package we created to implement this denoiser, and compare its performance on mock community data with DADA (Rosen et al., 2012). Finally, to illustrate that our cluster-free filtering approach is not restricted to longitudinal Illumina data, we provide an example of its application to a very different dataset, specifically 454 sequencing data from a cross-sectional environmental sampling performed by Preheim et al., 2013.

1 Motivation: sequencing noise is low

Clustering can be a useful strategy for filtering noise by coarse-graining data. However, such coarse-graining may not be a necessity: if the noise level is low, as suggested by known estimates of PCR and sequencing error rates (see, for example, Quince et al. 2011), then we can avoid clustering, since we expect each community member to be predominantly represented by the same 16S sequences.

We begin by illustrating this idea using the tongue microbiome data of Caporaso et al. Since the tongue community is relatively stable (Costello et al., 2009), the low-noise scenario would predict that certain specific sequences should consistently dominate in each sample. Alternatively, if the noise were high, then the high-abundance community members would be represented by clouds of similar reads, none of which would clearly dominate.

To show that the data of Caporaso et al. supports the first (low-noise) scenario, we identified the top 5 sequences by overall abundance. These sequences were strongly different (Fig. S1, inset), corresponding for the most part to bacteria from different phyla: in decreasing order of abundance, these were *Neisseria sp.* (phylum *Proteobacteria*, class *Betaproteobacteria*), *Haemophilus sp.* (phylum *Proteobacteria*, class *Gammaproteobacteria*), *Fusobacterium sp.* (phylum *Fusobacteria*), *Streptococcus sp.* (phylum *Firmicutes*), and *Prevotella sp.* (phylum *Bacteroidetes*). (Taxonomy assigned by a BLAST search (BLASTN 2.2.22, matrix $(1, -1)$, gap/extenstion penalty $(5, 2)$) against GreenGenes database; DeSantis et al., 2006. All 5 sequences had a match with 100% identity over 100% of sequence length.) The sample-by-sample rank of these overall top 5 sequences was consistently in the top 10. We stress that the goal of Fig. S1 is not to characterize the temporal stability of community composition (previously characterized, for example, in Costello et al., 2009); rather, it serves to show that the community members that correspond to these highest-abundance tags are consistently represented by the same 130nt sequence (at 100% identity) across all samples. In other words, despite the presence of noise in the data, 100% sequence identity is not an unreasonable criterion: the error rate is low enough that the error-free sequence dominates over the "error cloud" of its variants (Edgar, 2013). This key observation is the foundation of the approach described in this work.

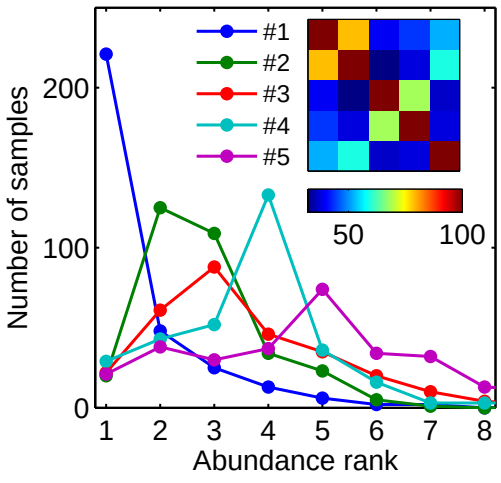

FIG. S1. The distribution of ranks for the top 5 sequences over all samples. Inset: pairwise sequence similarity (%). The top sequences are strongly distinct and their rank is consistent across samples.

2 Estimating rates of one-nucleotide substitutions

To estimate the rates of substitution errors observed in data after quality filtering, we used the "error clouds" around the high-abundance sequences in the dataset. Since all sequences were trimmed to a length of 130nt, each "mother" sequence has 390 direct neighbors in sequence space (Hamming distance = 1). For very high-abundance sequences such as Seq. #1, all 390 neighbors were observed in at least one sample of the time series. The time series of their abundances, normalized to the abundance of Seq. #1, is shown in Fig. S2. For this figure, the neighbors were ordered by the type of substitution that differentiates them from the mother sequence, and, within these categories, by the position of the differing nucleotide along the sequence. We see that, with a few exceptions (most notably the three neighbors also shown in Fig. 1B), the abundance of a given neighbor is a constant fraction of the abundance of the mother sequence. This is precisely what we expect for neighbors that arise as PCR or sequencing errors of the mother sequence, and the abundance ratio is then the probability of that particular error.

We see that the error rate is set primarily by the type of substitution, and does not exhibit significant dependence on the position along the sequence. For long reads, we would likely have seen an increase in error rates towards the end of the sequence, but our sequences are only 130nt long, well within the capabilities of accurate base-calling of the Illumina platform. We can therefore assign probabilities to substitution errors based solely on the substitution type (which nucleotide was replaced by which other), independent of the position along the read.

To determine these probabilities, we first identify the neighbors that are outliers in their substitution category; they likely correspond to true biological sequences physically present in the community, rather than sequencing

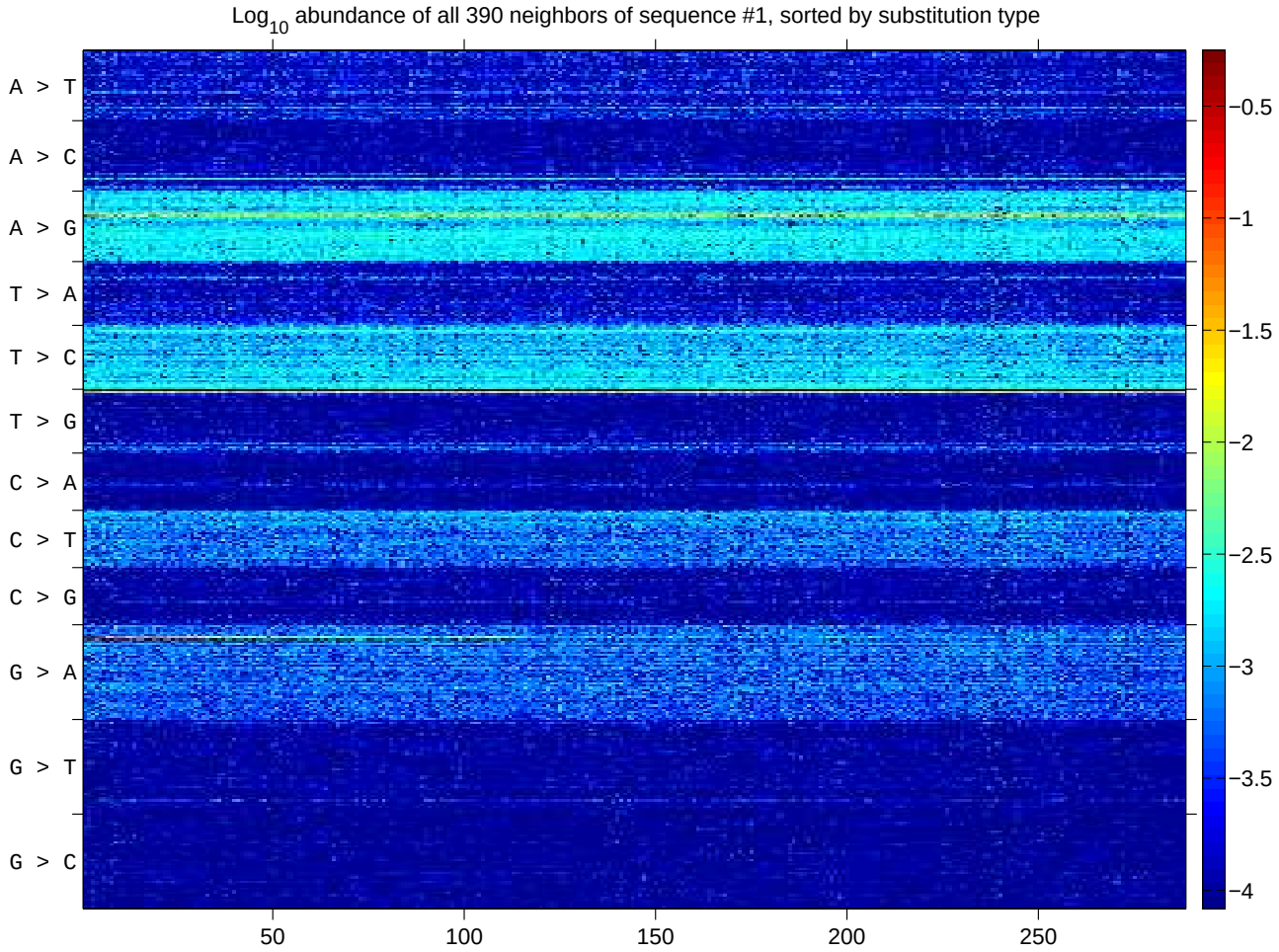

FIG. S2. This extended version of Fig. 1B shows all 390 first neighbors of Seq. #1, ordered by the type of substitution (and within these classes, by the position of the substitution along the sequence). Color indicates abundance on a log scale, normalized to the abundance of Seq. #1. Except for a few overrepresented neighbors (*cf.* Fig. 1B), the substitution type accounts for most of the variance in neighbor abundance.

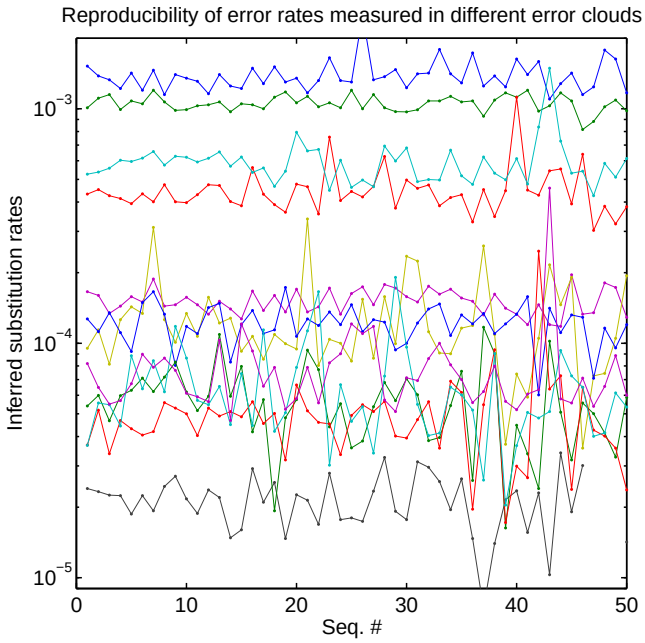

FIG. S3. The substitution error rates directly inferred from the error clouds of top 50 sequences by abundance are reproducible across error clouds. Each of 12 separate plots shows the inferred rate of a specific substitution (not labeled to reduce clutter; see also Fig. S4 and Sup. Table S1). Predictably, the variability increases when the error clouds of less abundant sequences are used.

errors. Outlier exclusion is done based on z-scores, i.e. for each sequence we compare its raw cumulative abundance over all samples to the mean in its substitution category, and normalize by the standard deviation in the category. A strong outlier differing from the mother sequence by a nucleotide substitution at location K will skew the error rate estimation at that location: some substitution type will appear to be unusually frequent. Therefore, we exclude nucleotide locations that correspond to the strongest outliers, those for which the z-score exceeds some threshold. The remaining locations are then used to estimate the error rates: for each of these locations, we count the number of times a particular substitution occurred, as well as the number of times the nucleotide was recorded correctly. After appropriate normalization, these counts give us the probability of each type of the error.

Fig. S3 shows the inferred substitution rates for error clouds around the top 50 sequences by abundance, using minimally quality-filtered data processed as described in the Methods (Phred score cutoff 2, z-score threshold 2). We find that the rates of different substitutions can differ dramatically (up to 50-fold), but our estimates are highly reproducible (note the log scale on the Y axis), with variability predictably increasing if lower-abundance error clouds are used. Table S1 lists the error rates estimated from the error clouds of the top 10 sequences (mean $\pm$ standard deviation). Note that, in principle, this effective error probability includes both the base-call errors of the Illumina sequencer and the single-nucleotide substitution errors occurring during PCR. However, the approximate symmetry between rates of a substitution and its reverse-complement partner (e.g. $p_{T\to A} \approx p_{C\to G}$), and a clear bias towards transitions as opposed to transversions, suggests that the observed substitutions are dominated by PCR errors (compare with Quince et al., 2011, Table 2).

Inferring error rates directly from the data offers multiple strong advantages. Specifying the error rate as an external parameter (e.g., Morgan et al., 2013) necessarily requires resorting to a conservative global upper bound. Different PCR conditions and different sequencing machines will have different error rates (for example, compare Fig. S4 and Fig. S7A). Further, substitutions vary strongly in probability: in our case, using a single upper bound on error rates would have over-estimated the probability of certain error types by up to 50-fold, reducing our ability to resolve close sequences. In other words, measuring substitution rates directly from the data both reduces the number of algorithm parameters and improves performance.

To investigate how the measured substitution probabilities depend on the quality filtering parameters, we applied the same analysis to data filtered using different Phred quality score thresholds ($Q_{\min} = 2, 10, 15, 20$) as well as different z-score thresholds (2.0, 3.5). The results are presented in Fig. S4. As expected, the average error rates increase as the Phred score threshold is lowered; however, the magnitude of this change is very small, comparable

	$\to A$	$\to C$	$\to T$	$\to G$	Total
$A \to$		0.14 ± 0.06	0.15 ± 0.02	1.34 ± 0.12	1.63 ± 0.14
$C \to$	0.07 ± 0.02		0.59 ± 0.04	0.06 ± 0.01	0.72 ± 0.05
$T \to$	0.12 ± 0.03	1.06 ± 0.07		0.07 ± 0.01	1.26 ± 0.08
$G \to$	0.42 ± 0.03	0.02 ± 0.01	0.05 ± 0.01		0.49 ± 0.03

TABLE S1. Substitution error rates per nucleotide, multiplied by 1000, as measured from the "error clouds" of the top 10 sequences by abundance. Error bars are standard deviations across the 10 estimates.

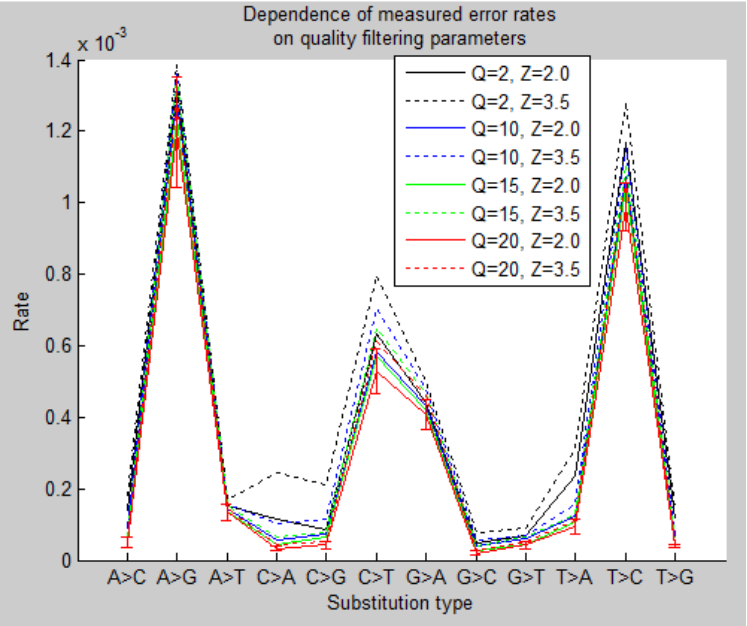

FIG. S4. The estimates of substitution error rates exhibit only a weak dependence on Phred quality score filtering parameters, as expected if the dominant source of substitution errors is PCR amplification rather than base call errors. Shown are average error rates as measured for top 10 sequences in the sample. Error bars on the (Q=20, Z=2) plot indicate standard deviation across the 10 estimates.

with the variability of error rate estimates across the top 10 sequences as indicated with the error bars on the plot corresponding to the most stringent filtering, $Q_{\min} = 20, Z = 2$. This provides further evidence that the majority of substitution errors occur during PCR amplification rather than during sequencing, and thus are not captured by Phred quality scores. We conclude that strict Phred quality filtering unnecessarily reduces data quantity while only marginally improving its quality; for our analysis, we therefore subjected the reads to minimal quality filtering as described in the Methods.

The dependence on z-score threshold is also consistent with our expectations: a high z-score threshold increases the error rate estimate. Predictably, including stronger outliers ($z = 3.5$) causes the measured error rate to vary significantly across filtering conditions; we used $z = 2$ which provided excellent reproducibility.

The reproducibility of error rates as observed on Fig. S3 justifies a posteriori our simplifying assumptions such as neglecting the probability of double substitutions in our calculation. Note that, according to the Table S1, the average total error rate per nucleotide is only 1.0 10^{-3}/nt. Therefore, within our error model, assuming that errors occur independently, we estimate find that a 130 nt-long sequence has 88% probability of being recorded with no errors. In practice, errors appear to correlate and the true zero-error probability is likely lower. As a different estimate, we calculate the total abundance of all sequences retained by our filtering (7 057 860 reads in 507 samples), and compare to the total number of reads before filtering (8 685 722 reads distributed across 1.4M unique sequences). We find that the filtering algorithm retained 81% of all reads. Since the algorithm intentionally disregards true sequences with low abundance, this estimate is conservative. Further, this estimate is largely insensitive to the error independence assumption: given our stringent filtering criteria, even an unexpectedly frequent double error will be discarded, provided it is less common that a single error. We conclude that $> 81\%$ of reads in the dataset had no errors, which justifies our decision to discard noisy reads rather than attempting to remap them to their most likely source. For longer reads or noisier data, our approach remains applicable without changes; however, the fraction of

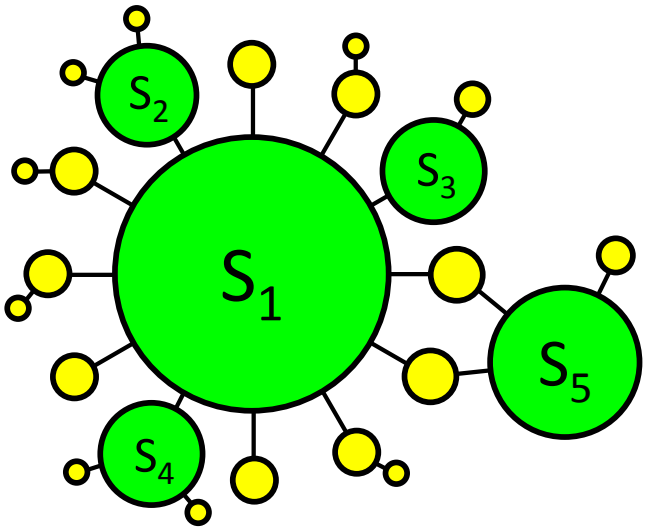

FIG. S5. A more detailed version of the error cloud cartoon in Fig. 1C. Each circle is a unique sequence, with size representing abundance in a sample. True biological sequences (S_1-S_5; green circles) generate "daughter" variants due to substitution errors (yellow circles). Black lines denote Hamming distance = 1 in sequence space. The error rates calculated from the error clouds (see Fig. S3) can be used to calculate, for every sequence, its expected abundance under the assumption that it arose through substitution errors from its more abundant neighbors. Sequences whose abundance is significantly above this expectation are labeled as real (green circles). Note that sequences may arise as substitution errors of multiple "mother" sequences: common neighbors of S_1 and S_5 in this cartoon will have a larger abundance than other substitution errors of either S_1 or S_5. However, if this increase in abundance is consistent with the null model, they will be correctly recognized as substitution errors.

error-free reads will be lower. In this case, to avoid significant loss of sequencing depth, we recommend replacing our simple denoiser by an algorithm such as DADA that performs read remapping.

3 The algorithm for filtering substitution errors

For the Illumina sequencing platform, substitution errors account for the bulk of the errors. As described above, these errors have a reproducible structure and their rates can be estimated directly from the data. Using these numbers, for any sequence S present in a given sample, we can estimate its null model abundance, denoted N^0 (abundance derived from sequencing errors of its more abundant neighbors), as follows (Fig. S5):

1. Order sequences by decreasing abundance: S_1, S_2, etc.
2. Set $N_i^0 = 0$ for all i
3. For each sequence S_i with abundance N_i:
   (a) Find all j such that S_j is a first neighbor of S_i and $N_j < N_i$.
   (b) For each j, use the substitution error table to determine the probability p_{ij} of S_i to be recorded as S_j
   (c) Set $N_j^0 = N_j^0 + p_{ij}N_i$ ("spillover" from S_i into S_j)

This zero-parameter algorithm assigns, for each sequence, its null-model abundance expected in that particular sample, using error rates estimated directly from the data. We next use this information to identify "candidate sequences": those whose presence cannot be explained by a substitution-only error model. Candidate sequences are selected through an abundance criterion, requiring their abundance to exceed the null model prediction (N^0) by at least ten-fold, and be no less than 10 counts. We then retain all sequences that independently passed this stringent filtering in at least 2 samples. The reasoning behind this strategy is explained in the Methods section of the main text.

4 Other error types, including chimeras and PCR indels

With Illumina sequencing, substitution errors account for most of the erroneous sequences in the data, and their occurrence appears to be adequately described by a simple quantitative model. This type of errors is therefore well-suited for error-model based denoising. The list of sequences retained after denoising includes true biological sequences, but also errors not described by our model. The latter category includes chimeras, PCR indels, and possibly other errors such as context-dependent PCR substitutions occurring much more frequently than expected within our model.

We are not aware of any quantitative model for PCR indel errors, which, in our experience, are strongly context-specific. Following Rosen et al., one could make the conservative decision that whenever two candidate sequences differ by pure indels, the lower-abundance should be treated as a possible error. A corresponding script is included in our cluster-free filtering pipeline. However, by definition, this makes it impossible to resolve true biological sequences differing by an indel. Retaining putative indel errors and comparing their abundance distribution across samples with their presumed "mother sequences" would allow identifying such cases. Since PCR indels are comparatively infrequent, for Illumina sequencing we consider indel filtering an optional step of the pipeline. In contrast, the 454 sequencing platform introduces frequent indel errors at homopolymer regions of the sequence. For 454 data, therefore, proper indel treatment becomes a necessity. The indel-filtering script we provide offers one solution; however, since errors we seek to eliminate occur during PCR, while indels occur during 454 sequencing, the best indel treatment strategy for the 454 platform is to merge sequences into "indel families" (Rosen et al., 2012) prior to denoising. Implementing this functionality within our software package will improve its support of 454 data; at the moment, the better approach is to apply our cross-sample analysis to the output of the DADA denoiser (Rosen et al., 2012).

As for chimeras, in our analysis pipeline, we filter chimeric sequences with UCHIME de novo (Edgar, 2011). Following Robert Edgar (UCHIME documentation), we recommend applying chimera filtering to pooled data across samples.

5 Cluster-free filtering software package

The implementation of the denoising algorithm described here is freely available at http://github.com/hepcat72/cff as a suite of open-source Perl scripts. Fig. S6 summarizes the workflow of the filtering process.

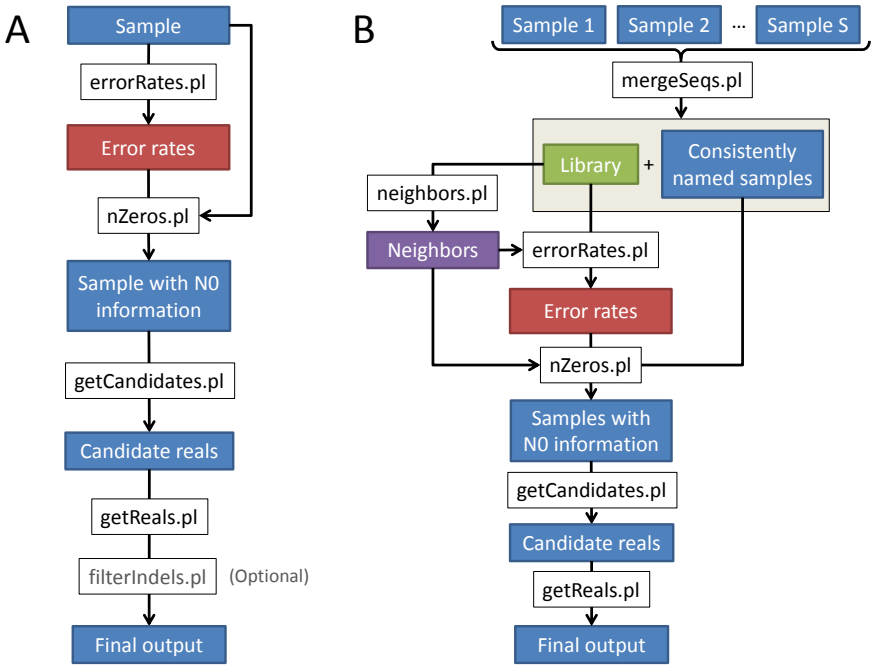

FIG. S6. The workflow of cluster-free filtering software package. **A:** The simplest way of running cluster-free filtering denoiser on a single sample. **B:** Extended workflow diagram appropriate for large multi-sample datasets. The optional indel filtering step is omitted for simplicity. In both cases, the blue rectangles represent dereplicated FASTA files with sequences of identical length. **getReals.pl** includes chimera filtering (performed with UCHIME).

Applying the denoiser on a per-sample basis is a straightforward four-step process, optionally supplemented by indel filtering (Fig. S6A). However, our denoiser is specifically designed to be run on large multi-sample datasets. The extended workflow appropriate for large datasets (Fig. S6B) has three key differences:

1. The original samples are pooled to construct a library of all unique sequences ever observed, and sequences in the original samples are renamed so that the same sequence has the same identifier in all samples (**mergeSeqs.pl**).

2. The error rates are estimated using the pooled data from all samples (i.e. the library) for better accuracy.

3. The neighbor structure is constructed once, for all sequences in the library (**neighbors.pl**). In the per-sample workflow (Fig. S6A), **neighbors.pl** is automatically invoked for every sample in a manner transparent to the user, which simplifies the workflow; however, many sequences are shared across samples, so for sufficiently large datasets, explicitly calculating the neighbor structure only once results in better performance.

The optional indel-filtering step uses MUSCLE aligner (Edgar, 2004) with modified gap penalty parameters, appropriate for detecting 454 homopolymer indels (`-gapopen -400 -gapextend -399`; see documentation).

The software package is provided with built-in documentation, a test dataset and two shell scripts that allow running the entire workflow presented above with a single command: `run_CFF_on_FastA.tcsh` and `run_CFF_on_FastQ.tcsh` (the latter script uses USEARCH to perform the minimal quality filtering as described in the Methods). The flexible and thoroughly documented command-line interface makes it easy to incorporate cluster-free filtering into any existing pipeline. To reproduce our analysis of the data from Caporaso et al. (2011), download the quality-filtered data published with that study (available at MG-RAST:4457768.3-4459735.3), place it in a folder `CaporasoData` and run:

```
tcsh run_CFF_on_FastA.tcsh 130 analysisResults "CaporasoData/*.fna".
```

6 Mock community validation and comparison with DADA

To validate the performance of our simplified denoiser, we compared it with a state-of-the art denoiser DADA (Rosen et al., 2012) using two mock community datasets (*Divergent* and *Artificial*; Quince et al., 2011) that Rosen et al. used to demonstrate DADA's superior accuracy to AmpliconNoise. Quoting from the original publication, these datasets were constructed by amplifying the V5 region of the 16S rRNA gene from 23 and 90 clones, respectively, isolated from lake water. The *Divergent* clones were mixed in equal proportions and are separated from each other by a minimum nucleotide divergence of 7%, while the *Artificial* clones were mixed in abundances that span several orders of magnitude, with some of the clones differing by a single-nucleotide substitution. For purposes of comparison, we used the exact same sets of filtered reads (35 190 reads in *Divergent* set; 31 867 in *Artifiical*), kindly provided to us by Michael Rosen.

The comparison of denoiser output and the reference set of Sanger clones was complicated by the imperfections of the reference set. A number of "reference" Sanger clones differed from their closest high-abundant matches in the 454 data at the same locations towards the beginning of the read, which is suggestive of errors in the reference sequences. Further, some reference sequences of the *Artificial* set had no close matches in the data; some Sanger clones differed at locations that were not part of the 454 sequenced fragments; and 454 sequences included 6 extra bases at the beginning of the sequence that were absent from the Sanger clones.

We therefore began by constructing "cleaned" reference sets as follows: for each reference Sanger clone, we found its closest match in the dataset that had at least 98% similarity and an abundance of at least 10 counts. This matching 454 read was used as the new reference sequence, and the differences, if any, were ascribed to Sanger clone errors. For the *Divergent* dataset, each reference sequence had exactly one clear match in the 454 data. For the *Artificial* set, of the 90 reference Sanger clones, we found that one was 29 nts away from the closest 454 read; for 3 other clones, no 454 read within $\geq 98\%$ sequence similarity radius reached an abundance of 10 counts. Our algorithm intentionally disregards any sequences below this abundance threshold; therefore, for the purposes of this comparison these reference sequences were considered absent and we did not count them as false negative for any of the algorithms. Several groups of clones were not distinguishable by the 454 sequenced fragment. Altogether, the new reference set of sequences that were both present and distinct contained 49 reference sequences.

We then ran DADA and cluster-free-filtering on both datasets. DADA was run with the same parameters as used for this data in the original publication, namely $\Omega_a = 10^{-40}$ and $\Omega_r = 10^{-3}$. Cluster-free-filtering included indel filtering step, since this data was obtained using the 454 platform and indels appear frequently.

The results are presented in Table. S2. Both algorithms identified correctly all 23 reference sequences of the *Divergent* dataset. For the *Artificial* set, and due to the conservative parameters recommended by Rosen et al., one of the reference sequences was missed by DADA but was correctly identified by our algorithm. Sequence #35 (in order of decreasing abundance), absent from the reference set, was retained by both algorithms and is likely a true

Category	DADA	CFF	Abundance
Divergent: 23 reference sequences	23 true positives 0 false negatives	23 true positives 0 false negatives	231-1426 counts
Other detections	0	0	
Artificial: 49 reference sequences	48 true positives 1 false negative	49 true positives 0 false negatives	18-3587 counts
Other detections	Seq. #35	Seq. #35 Seq. #95 Seq. #103 Seq. #119	163 counts 13 counts 12 counts 12 counts

TABLE S2. Comparison of DADA and cluster-free filtering (CFF) denoiser on mock community data. Sequences numbered by decreasing abundance in the dataset.

biological sequence. Cluster-free filtering generated 3 additional detections just above its threshold of 10 counts. It is instructive to trace the origin of these calls. For example, Seq. #95 was discarded by DADA as possibly an erroneous read generated by its closest reference sequence (Seq. #1) two substitutions away. Specifically, Seq. #95 differs from Seq. #1 by a T at location 23 and a G at location 118, a relation that we denote "Seq.#95 = Seq.#1 23T 118G". If it were true that Seq. #95 is a substitution error of Seq. #1, we would generally expect single-error variants to be more abundant than double errors. In reality, Seq.#1 23T (=Seq. #587) and Seq.#1 118G (=Seq. #121) have abundances of just 4 and 12 counts, respectively, which is why our algorithm identified Seq. #95 as likely real. However, its unexplainably high abundance could also have arisen through amplification of a double substitution that occurred early in the PCR cycle, and the default parameters of DADA were chosen conservatively so as to eliminate such cases (Rosen et al., 2012). Whether or not these detections are false positives or true biological contaminants can be determined only by a cross-sample analysis as presented in the main text.

7 Runtime comparison with DADA

The methodology presented in this work was designed to perform cross-sample comparisons of sequence abundance in individually denoised samples. As explained in the Methods, the simplified denoiser we developed is meant to maximize performance on large datasets specifically for this application, taking advantage of our focus on moderate-to-high abundance sequences. Other denoisers can be used. To estimate the runtime of DADA on the tongue dataset considered here, we used a representative subset of 20 samples, 10 from lane 5 and 10 from lane 6 of Caporaso et al. Following the instructions in Rosen et al., 2012, we used ESPRIT to precluster sequences in each sample prior to processing them with DADA. The measured runtime is presented in Table S3. Extrapolation to the full set of 507 samples yields the estimate of $2.3\,10^5$ sec total runtime quoted in the text, compared to 626 sec actual runtime for cluster-free-filtering. As explained in the main text, one of the reasons for this speedup is that our multi-sample detection strategy allows us, in any given sample, to look for candidate sequences only among those with abundance $\geq$10 counts. This speedup can be applied to DADA as well; to this end, we removed all clusters that contained no sequences with abundance $\geq$10 counts, and measured DADA runtime after this filtering; this decreased the estimated runtime on the full dataset to $5.5\,10^4$ sec. Using DADA in this way is the strategy we recommend for applying our cross-sample comparison methodology to 454 data with long reads where erroneous read remapping and indel family merging become advisable. Eliminating low-abundance sequences leads to a considerable improvement of DADA runtime; nevertheless, the total runtime remained two orders of magnitude slower than our cluster-free filtering approach, due primarily to the computational cost of preclustering.

8 Example of other applications: environmental cross-sectional 454 data

The approach described in this work does not explicitly rely on the longitudinal nature of the sampling. Most of our analysis can be readily applied to any multi-sample datasets, e.g. a cross-sectional sampling or a location series, provided samples were collected and processed in a similar way so that the error structure can be assumed to be similar. Further, and despite the caveats we described, our method can be applied even to data collected using the 454 sequencing platform. To illustrate the broad applicability of our approach, we used data from a cross-

	ESPRIT+DADA	ESPRIT+DADA, abundant clusters only	CFF denoiser
Lane 5, 10 samples	969 + 1297 sec	969 + 49 sec	12 sec
Lane 6, 10 samples	924 + 5133 sec	924 + 186 sec	16 sec
Whole dataset	$2.3\,10^5$ sec (est.)	$5.5\,10^4$ sec (est.)	626 sec (actual)

TABLE S3. Runtime comparison of DADA and the simplified cluster-free filtering (CFF) denoiser on two representative sets of 10 tongue samples (lane 5 and lane 6). Lanes were considered separately since samples in the two groups tended to differ significantly in the number of reads retained by quality filtering. The whole dataset consisted of 189 samples on lane 5 and 320 samples on lane 6. Comparisons were performed on an Intel Xeon CPU 2.83GHz.

sectional environmental sampling conducted by Preheim et al. (SRA accession number from SRP029470). Lake water microbiota were sampled at depths ranging from 0 m (surface) to 22 m with 1-meter depth intervals. The authors used this data to illustrate their sequence clustering algorithm (DBC) that also relies on cross-sample comparisons to distinguish between closely related OTUs; for details, see the original reference (Preheim et al., 2013). They report their algorithm worked best with stringent quality filtering whereupon sequences were trimmed to just 76 nt, and any reads containing bases with Phred quality scores at or below 16 were discarded. This filtering retained 7.78M total sequences (120K unique). Since our approach includes data denoising, we could use much more liberal quality score filtering and retain more data (USEARCH maxEE of 1 and truncating at Phred quality score 2). To compare runtime of our algorithm and DBC, we increased the read truncation length so as to keep the same total number of sequences. This set the quality-filtered sequence length to $L = 91$ nt, 20% longer than used by the authors (7.98M sequences, 300K unique; `tcsh run_CFF_on_FastQ.tcsh 91 analysisResults "PreheimData/*.fastq"`.).

Fig. S7A shows the substitution error rates inferred from the data at both sets of quality filtering parameters. Note that these rates are significantly lower than those of Fig. S4 (the scales of the two plots are identical), exhibit a very weak transition/transversion bias, and are more sensitive to Phred score quality filtering than what we have seen with Caporaso et al. data (Fig. S4). This seems to indicate that the protocol used by Preheim et al. generates significantly fewer PCR substitution errors. This dependence of error rates on the experimental protocol highlights the advantage of being able to estimate error rates for a given dataset directly from the data, without the need for a separate calibration.

Fig. S7BC provide examples of sequences differing by a single nucleotide exhibiting ecologically significant distinctions, as identified by our method in this environmental dataset; compare with Fig. 2A, Fig. S10 and Fig. S11AB. Sequence abundance is shown as a function of depth (each sample was independently normalized to $3.2\,10^5$ total quality-filtered reads per sample, to correct for varying sample size). The DBC method of Preheim et al. is also capable of identifying OTUs differing by a single nucleotide (compare Fig. S7BC to Fig. 5b in the original reference); however, our analysis achieved higher resolution by retaining longer reads and took only 13 min single-core processor

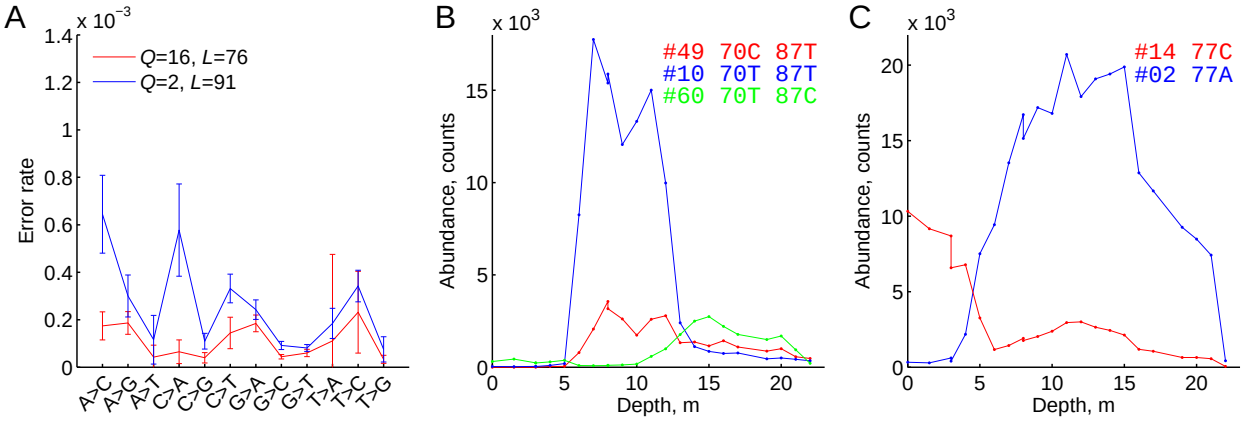

FIG. S7. Cross-sectional environmental 454 data: lake water microbiota (Lake Mystic) sampled at depths 0–22m. **A:** Substitution error rates inferred from the data for two sets of quality filtering parameters indicated in the legend; Q is the Phred quality score truncation threshold; L is read truncation length. **B:** Three sequences resolved by our analysis; Seq. #49 and Seq. #60 both differ from Seq. #10 by a single nucleotide at locations 70 and 87, respectively. Sequence abundance is shown as a function of depth; sequences are labeled by cumulative abundance rank. **C:** Same, for sequences Seq. #2 and Seq. #14 differing at nucleotide 77.

time (Intel Xeon CPU 2.83GHz), compared to 8 hours analysis time the authors report for parallelized DBC running on a cluster with 60-100 processes executing simultaneously. In fact, the true runtime difference is even greater, since the complexity of both algorithms scales with the number of *unique* sequences rather than total reads. For $Q_{\min} = 17, L = 76$ as used by the authors of DBC, our algorithm completes in only 210 seconds.

We stress, however, that DBC and cluster-free filtering seek to achieve different goals and are not directly comparable. DBC is an OTU clustering algorithm, whereas the goal of cluster-free filtering is to identify sub-OTU structure of moderate-to-high-abundance community members. However, to our knowledge DBC is the only existing tool that exploits cross-sample comparisons to inform the interpretation of sequencing data, and the performance comparison above serves to illustrate the drastically different computational cost of the two approaches.

9 How many samples is enough?

We have described a method that employs cross-sample comparisons to achieve sub-OTU resolution. The analysis presented in the main text uses data from a study with an uncommonly large number of samples; in contrast, the previous section demonstrates that our method can be usefully applied to a dataset with only 22 datapoints. What is the minimum number of samples required by our method?

The answer is that the number of samples determines the resolution that can be achieved; more samples will always allow higher resolution, but coarser differences can be resolved with just a few. For example, just 2 samples (say, 0m and 10m) would have been enough to resolve the two subpopulations presented on Fig. S7C. By contrast, the difference between depth traces of Seq. #10 and Seq. #49 (Fig. S7B) is less pronounced and more samples are required. Finally, resolving the sequences in Fig. 2B would not have been possible with fewer than ≈100 samples. For high-abundance sequences where the complex struture of noise in the counts can be neglected, this tradeoff can be formally quantified using the Jensen-Shannon divergence as a measure of distance between abundance distributions of two sequences across samples; for details, see Preheim et al., 2013.

B Supplementary information for Figure 2

1 A pair of sequences representing strongly anticorrelated subpopulations

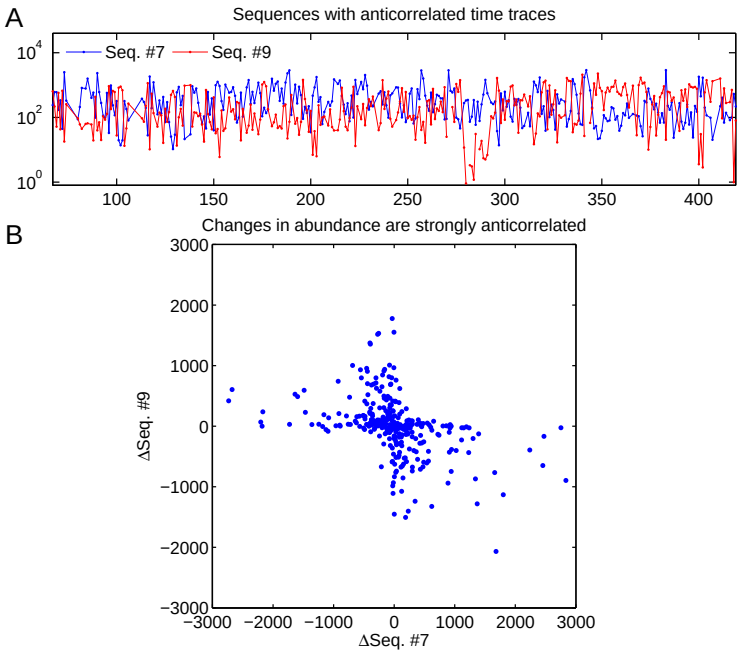

FIG. S8. **A.** Abundance time traces of Seq. #7 and Seq. #9. **B.** Scatter plot for the same sequences of their discrete derivatives of abundance (i.e. abundance changes from each day to the next). A BLAST search against the GreenGenes database identifies the likely taxonomy of Seq. #7 as *Streptococcus thermophilus*. Seq. #9 does not have a good match; the closest hit is an unclassified *Prevotella sp.* at only 88% identity.

2 Best expected correlation of two time traces

The maximum degree to which time traces of two sequences can be correlated is a function of their abundance: for low-abundance sequences the Poisson sampling noise becomes non-negligible and sets an upper bound for the best achievable correlation coefficient. Consequently, to define a correlation as strong or weak, any measured correlation coefficient should be compared to this abundance-dependent quantity rather than to 1.

Let $N(t)$ be the true abundance time trace of some bacterial strain (in units of cells, rather than sequence counts). Imagine that two sequences in the dataset were measuring the abundance of this exact same strain, but with different amplification efficiencies λ_1 and λ_2 (let $\lambda_1 > \lambda_2$). Neglecting all sources of noise other than the Poisson counting noise, the abundance traces of these two sequences can be modeled by

$$n_{1,2}(t) = \text{Poiss}[\lambda_{1,2} N(t)],$$

where Poiss[·] denotes adding Poisson noise. Since Poisson noise is unavoidable, the correlation coefficient between these two traces sets an upper bound for the correlation between $n_1(t)$ and any other trace $n^*(t)$ with the same mean abundance as $n_2(t)$. This maximum correlation depends on the shape of the trace $N(t)$ and amplification efficiencies λ_1, λ_2, and can be expressed as follows:

$$c_{\max}[N(t), \lambda_1, \lambda_2] = \text{corr}\left(\text{Poiss}[\lambda_1 N(t)], \text{Poiss}[\lambda_2/\lambda_1 * \lambda_1 N(t)]\right).$$

And therefore, in terms of measurable quantities only:

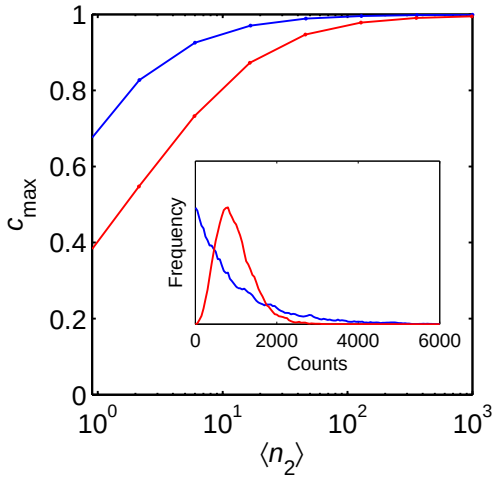

FIG. S9. Best expected correlation $c_{\max}$ for a pair of abundance time traces $n_{1,2}(t)$ is sensitive to the shape of the distribution of the daily counts, not just the average abundance $\langle n_{1,2}(t)\rangle$. The figure shows the best expected correlation $c_{\max}[n_1(t), \langle n_2\rangle]$ as defined in the Supplementary Information, for two different mock traces $n_1(t)$ with the same mean ($\langle n_1\rangle = 1000$ counts/day) but with different distributions, modeled here by Gamma distributions with shape parameter 1 (blue) and 5 (red). The best expected correlation increases with the mean abundance $\langle n_2(t)\rangle$, but for the same mean it is higher for the blue trace whose distribution covers a wider dynamic range. Because of this nontrivial dependence on the distribution shape, in our definition of dynamical similarity we compute the best expected correlation individually for every pair of sequences.

$$c_{\max}[n_1(t), \langle n_2\rangle] \approx \mathrm{corr}\left(\mathrm{Poiss}[n_1(t)], \mathrm{Poiss}[\langle n_2\rangle / \langle n_1\rangle * n_1(t)]\right).$$

Here $\langle\cdot\rangle$ denotes the average abundance, and we use the higher-abundance trace of the pair as the best estimate of the shape of the true abundance $N(t)$. The maximum correlation coefficient depends on the shape of the trace $n_1(t)$ and on the mean abundance of the trace we compare it to; the lower the mean abundance is, the stronger the effect of Poisson noise and the lower the $c_{\max}$.

In practice, for a pair of traces $n_1(t)$, $n_2(t)$, we compute their best expected correlation as follows:

1. Take the more abundant trace $n_1(t)$
2. Construct a renormalized trace $n_2^{\mathrm{mock}}(t) = \frac{\langle n_2\rangle}{\langle n_1\rangle} n_1(t)$
3. Poisson-resample both of these 10 times: denote these $n_1^{(i)}$, $n_2^{\mathrm{mock}(i)}$, $i = 1 \ldots 10$.
4. Compute the 100 correlation coefficients between all pairs $c_{ij} = \mathrm{corr}\left(n_1^{(i)}, n_2^{\mathrm{mock}(j)}\right)$.
5. Set $c_{\max}[n_1(t), \langle n_2\rangle] = \langle c_{ij}\rangle$.

The shape of the function $c_{\max}[n_1(t), \langle n_2\rangle]$ is illustrated on Fig. S9.

3 Distance metric for sequence pairs

We use the BLAST definition, i.e. the ratio of the number of mismatches to the total number of columns after pairwise realignment, and multiply this ratio by the length of the sequence (130 nt). For close sequences that differ by a few substitution errors the alignment is trivial, and this normalization corresponds to the Hamming distance between sequences, in nt.

C Supplementary information for Figure 3

1 Estimating correlation time from autocorrelation function

We define the autocorrelation time τ of a sequence as the time shift Δt at which the autocorrelation function $c_{\Delta t}$ falls below the threshold of statistical significance. For reasons discussed above, the notions of strong (significant) or weak (insignificant) correlation of sequence time traces are abundance-dependent. Therefore, instead of using a fixed threshold value for all sequences, we proceed in the following way. For a given sequence, we first compute its root-mean-square autocorrelation coefficient for time shifts between 70 and 100 samples:

$$c_{\text{null}} = \sqrt{\langle (c_{\Delta t})^2 \rangle_{\Delta t = 70 \ldots 100}}.$$

If we assume that all autocorrelation observed at such large time shifts is entirely due to noise, then c_{null} provides a natural scale for statistical significance. We conservatively define the significance threshold at twice the magnitude of c_{null}.

Note that c_{null} provides an upper bound on a statistically significant correlation value. If some dynamical processes in the population are slow enough that they contribute to the autocorrelation function even at such large time shifts (*cf.* Fig. S10), this will increase c_{null} and cause us to underestimate the true autocorrelation time. This means that assuming c_{null} was entirely due to noise is a safe approximation to make: if it does not hold, it can only strengthen our conclusion that the sequence abundance time traces exhibit multi-day autocorrelations.

2 Examples of sequences exhibiting consistent dynamics on very long time scales

Fig. S10AB shows examples of sequences exhibiting steady change in abundance for more than a month. In both cases, the slow-changing sequence is 99.2% similar to a very high-abundance community member and could not have been resolved by traditional OTU-based methods. Note the sharp jump in panel B at day 182 of the sequence representing the invading subpopulation (red) to an abundance value close to the equilibrium established after day 210. It is intriguing to speculate that this trace may document spatial invasion of a subpopulation already established elsewhere on the tongue, a region accidentally sampled on day 182.

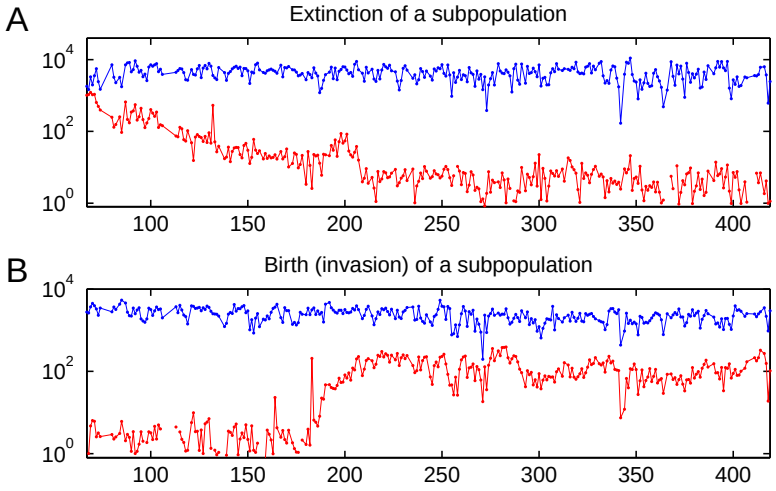

FIG. S10. **A.** Slow extinction of a subpopulation (red; cf. Fig. 1B, neighbor 3). From day 210 onwards the abundance of the red sequence is consistent with it being a substitution error of Seq. #1 (blue), which is a direct neighbor in sequence space. **B.** Slow birth/invasion of a subpopulation (red). The new sequence differs by 1nt from well-established Seq. #2 (blue), and prior to day 160 its abundance is consistent with being its substitution error. Note the high similarity of fluctuations from day 210 onwards.

3 Persistence of difference: the null model

To distinguish between 16S tags coming from distinct subpopulations or from physically the same bacterial cells, we introduced a quantity we called the *persistence of difference* P_D. For this, we first defined the fractional difference $\Delta(t)$ between two time traces renormalized to the same mean $n_{A,B}(t)$:

$$\Delta(t) = \frac{n_A - n_B}{(n_A + n_B)/2}.$$

We then defined the persistence of difference P_D as the 1-day autocorrelation coefficient of $\Delta(t)$. If A and B are two genomic variants contained within the same bacterium, then any difference between $n_A(t)$ and $n_B(t)$ must be due to measurement noise, and P_D must vanish. If, however, $n_{A,B}(t)$ reflect abundances of two distinct subpopulations, then $\Delta(t)$ can be expected to exhibit some degree of autocorrelation due to the slow dynamics observed for most individual sequences. We gave an intuitive argument for this in the main text. Here, to gain some extra intuition about the null model for P_D, we calculate it explicitly in the simplest case when the two traces $n_{A,B}(t)$ are independent and can be approximated by a stationary, weakly fluctuating process:

$$n_A(t) = \mu\,(1 + \sigma_A \xi_A(t)) \tag{C1}$$
$$n_B(t) = \mu\,(1 + \sigma_B \xi_B(t)) \tag{C2}$$

Here $\xi_{A,B}$ have zero mean, unit variance and are uncorrelated. Assuming $\sigma_{A,B} \ll 1$, we can write:

$$\Delta(t) \approx \sigma_A \xi_A(t) - \sigma_B \xi_B(t)$$

And therefore, making use of the independence assumption,

$$P_D = \frac{\langle \Delta(t)\Delta(t+1)\rangle}{\langle \Delta(t)^2\rangle} \approx \frac{\langle \sigma_A^2 \xi_A(t)\xi_A(t+1) + \sigma_B^2 \xi_B(t)\xi_B(t+1)\rangle}{\sigma_A^2 + \sigma_B^2} = \frac{\sigma_A^2 c_{1A} + \sigma_B^2 c_{1B}}{\sigma_A^2 + \sigma_B^2}$$

Here $c_{1A,B}$ are the one-day autocorrelation coefficients of the fluctuations of the two individual sequences.

The independence approximation made above is clearly not valid for the dynamics of most community members. For this reason, for the purposes of Fig. 3C, the null-model prediction was constructed directly from the data, by reversing in all pairs the time order for one of the sequences prior to the calculation of P_D. This removes any real correlations of the traces while preserving autocorrelation and other properties of the traces such as their fluctuation spectrum. Nevertheless, the calculation above is useful as it explains why the null-model expectation for P_D is non-zero when both sequences have slow internal dynamics.

Note that a sequence with an exceptionally long intrinsic time scale (as shown in Fig. S10AB) will have a large P_D score when paired with any other sequence. These two sequences were therefore excluded from Fig. 3C.

4 Persistence of difference for non-longitudinal data

None of the cross-sample comparison methodology described in this work is limited to time series data. The "persistence of difference" argument accompanying Fig. 4 is no exception; however, it does rely on two additional assumptions, namely that the composition of samples varies smoothly with some parameter labeling the samples, and that the sampling frequency is sufficiently high to allow correlations of fluctuations to be observed between consecutive samples. For the longitudinal data series of Caporaso et al. this parameter was time; for a location series one can expect community composition to vary smoothly in space, and the same argument can be applied. In other words, the use of "persistence of difference" P_D need not be limited strictly to longitudinal datasets. However, autocorrelation-based analysis is particularly sensitive to the number of samples (see section A 9). Determining whether P_D can be a useful concept for studying the spatial heterogeneity of populations requires further investigation.

D Supplementary information for Figure 4

1 Over-estimation of OTU quality scores

As described in the main text, for the purposes of Fig. 4, when calculating OTU quality scores, we restricted our attention only to high-abundance members of the OTU, considering only sequences from the top 200 by overall abundance. Since most of the diversity is contributed by low-abundance species (Huttenhower et al., 2012), Fig. 4 underestimates the true diversity of an OTU. Including lower-abundance OTU members makes OTU quality scores drop continuously as new OTU members are added; however, it also becomes increasingly hard to separate dynamical diversity from the effects of noise. Consequently, in Fig. 4 we report our most conservative estimate of within-OTU diversity, where we use only the highest-abundance members out of all those resolved by cluster-free filtering (there was an average of 18 ± 4 resolved sequences within a 97% OTU, and only 9 ± 2 per OTU were used for Fig. 4).

In addition, OTU quality scores were calculated under the assumption that each sequence represents a separate subpopulation. Sequences that in fact derive from the same bacteria (16S paralogs or errors not in our model) appear in the defining equation as independent, dynamically identical subpopulations, increasing the apparent OTU quality score. This is another reason why the true quality scores of OTUs are likely even lower than reported in Fig. 4.

E Supplementary information for Figure 5

1 Cross-individual analysis of fecal samples

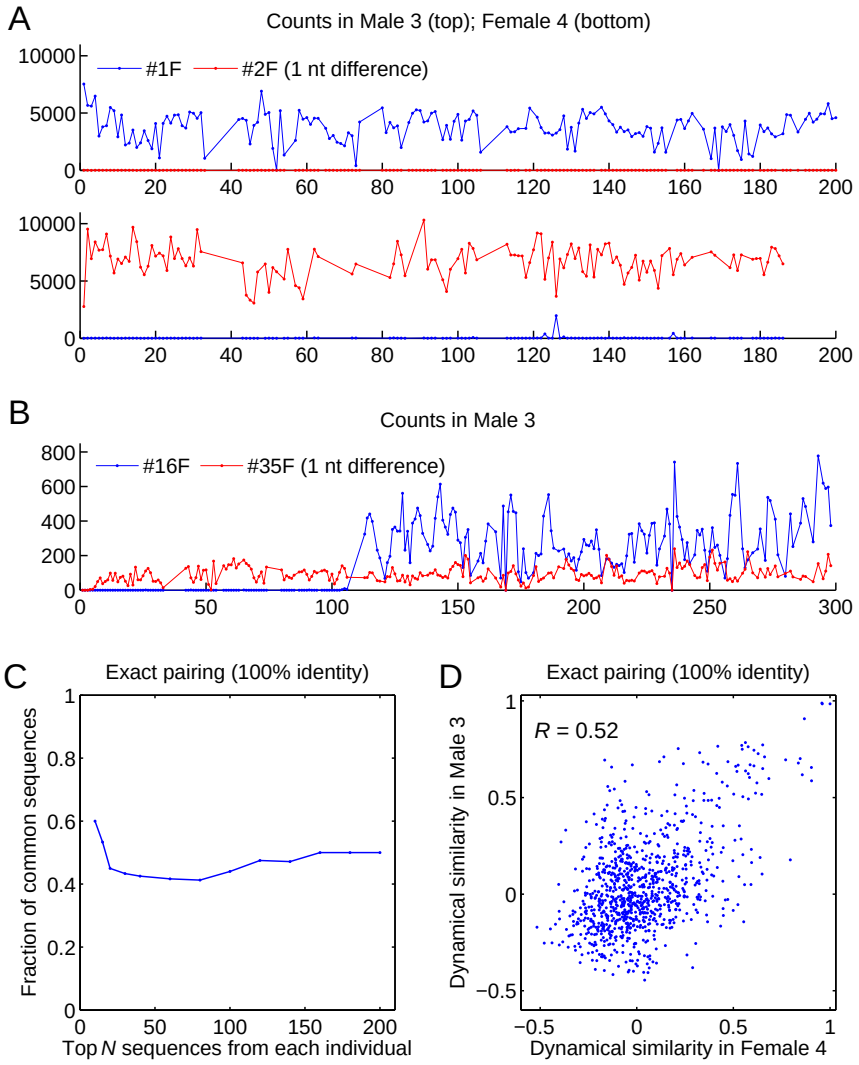

FIG. S11. **A.** Abundance time traces (sequence counts *vs.* observation day) for Seq. #1F and Seq. #2F, which differ by a single nucleotide and dominate in individuals Male 3 and Female 4, respectively. **B.** Another example of abundance time traces of two sequences that differ by a single nucleotide (99.2% similarity), yet exhibit strongly distinct dynamics and so derive from distinct bacteria. **C.** Fraction of shared 16S sequences, defined as the fraction of common tags (at 100% sequence identity) among the most abundant N sequences in the fecal samples of each of the two individuals, plotted as a function of N (compare with Fig. 5A.) **D.** Scatter plot of the dynamical similarity of pairs of common fecal sequences, as measured independently in the two individuals, for all possible pairs among the 44 common sequences shared within the top $N = 100$ (compare with Fig. 5B).

To confirm our conclusions from the analysis of tongue microbiome data presented in the main text, we repeated our analysis using fecal samples of the two individuals, collected in the same study (Caporaso et al., 2011). There were 374 samples, 243 from the male subject and 131 from the female, with $2.5 \pm 0.5\,10^4$ reads per sample. We normalized the observed abundances to $2.5\,10^4$ total reads in each sample to correct for varying sample size. As before, we labeled sequences in order of decreasing overall abundance (pooling samples from both individuals): Seq #1F, Seq #2F, etc., where "F" reflects that we are now dealing with fecal samples rather than the tongue.

Again, we find that sequences differing by as little as a single nucleotide can exhibit ecologically significant differences in their dynamics. The most striking example is that the dominating sequence in individual "Male 3" differs from

the dominating sequence in "Female 4" by a single nucleotide, and virtually no cross-contamination is observed (Fig/ S12A). Both these sequences map to *Bacteroides sp.* in GreenGenes (DeSantis et al., 2006). Another example is presented in panel B. Finally, we repeat the cross-individual analysis presented, for tongue samples, in Fig. 5AB. We find that the two gut communities as probed by the fecal samples also share a large fraction of sequences at 100% identity. This once again supports the scenario whereby the communities exchange members with non-negligible frequency, although less so than the tongue samples. The observation of panel A is therefore unlikely to represent the effect of dispersal limitation, suggesting instead a functional difference between the representatives of *Bacteroides sp.* established in the two individuals or a resistance to invasion. Finally, we find that the dynamical similarity of shared sequences, when measured independently in the two individuals, is clearly correlated, just as it was for the tongue communities (Fig. 5B). With the number of shared sequences being lower for fecal samples than for tongue samples, the statistics were insufficient to compare dynamical similarity of "intentionally mismatched" sequences as in Fig. 5C.

2 Cross-individual analysis at 97% OTU level

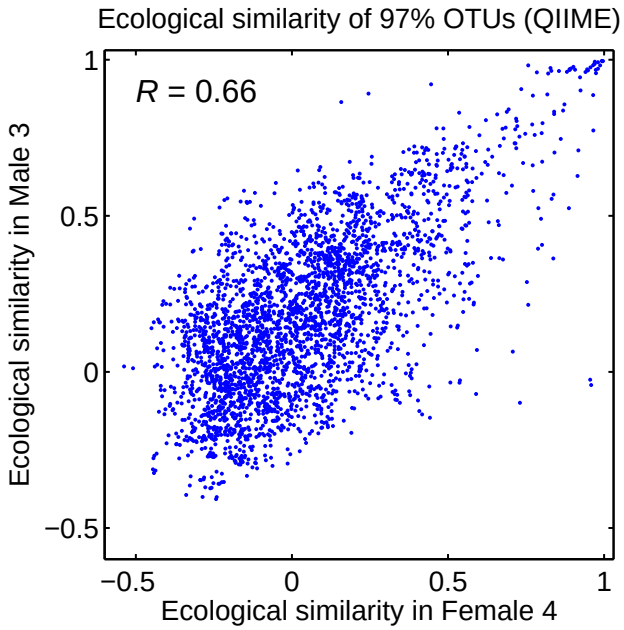

FIG. S12. Dynamical similarity between pairs of common 97% OTUs, as measured independently in the two individuals, for 78 common OTUs within the top 100, constructed using closed-reference OTU picking as implemented in QIIME.

The same analysis as in Fig. 5B can be performed for shared 97% OTUs rather than shared sequences (at 100% identity). We constructed OTUs using closed-reference OTU picking as implemented in QIIME, matching sequences at 97% sequence similarity against the GreenGenes database. Fig. S12 shows the scatter plot of the dynamical similarity between pairs of common OTUs, as measured independently in the two individuals, for 78 common OTUs (those shared within the top 100). Note, however, that most OTUs are dominated by a single high-abundance sequence (as evidenced by the high weighted quality score on Fig. 4), and most of these dominating sequences are shared across the two communities (Fig. 5A). For these reasons, the plot shown here is very similar to Fig. 5B, but only because the within-OTU diversity is masked by dominating subpopulations.

Supplementary references

DeSantis TZ, Hugenholtz P, Larsen N, Rojas M, Brodie EL, Keller K et al (2006). Greengenes, a chimera-checked 16S rRNA gene database and workbench compatible with ARB. *Appl Environ Microbiol* **72**: 5069–5072.

Edgar RC (2004). MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.* **32**:1792–1797.