UCHIME improves sensitivity and speed of chimera detection

Robert C. Edgar, Brian J. Haas, Jose C. Clemente, Christopher Quince and Rob Knight

Supplementary material

Contents

Single-region sequencing

UCHIME is designed for experiments that perform community sequencing of a single region such as the 16S rRNA gene or fungal ITS region, as illustrated in Figs. S1 and S2. While UCHIME may prove useful in other contexts, at the present time UCHIME has been validated only on 16S rRNA. Changes to the algorithm or parameters may give better results on other regions.

Classification and ROC analysis

Receiver-operator characteristic (ROC) curves (Mason and Graham, 2002) are used to summarize the performance of a binary classifier that computes a real-valued score and predicts a true/false result by testing whether the score exceeds a fixed threshold. With such a classifier, the sensitivity and error rate can be adjusted by changing the score threshold. To generate a ROC plot, the classifier is run on a test dataset where the correct classifications are known. For each unique value of the score obtained on the test set, the number of true positives and false positive results that would be obtained using that value as a threshold are recorded and displayed on a graph in which the X-axis is the percentage of true positives (sensitivity) and the Y-axis is the percentage of false positives (error rate). UCHIME, ChimeraSlayer and Perseus can all be interpreted as binary classifiers of this type. UCHIME uses the *h* score, ChimeraSlayer uses a bootstrap confidence percentage (*BS*), and Perseus uses a probability (*P*) that the query sequence is chimeric. The default thresholds are h=0.28, BS=90% and P=0.5 respectively. It should be noted that while UCHIME and ChimeraSlayer use a single score (*h* and *BS*, respectively) for classification, there are other parameters of these programs that also affect sensitivity and error rate. For example, both impose a threshold for the divergence between a chimeric alignment and the closest candidate parent (*--mindiv* option of UCHIME, *-R* option of ChimeraSlayer). Different ROC curves are obtained if these parameters are varied, and the maximum sensitivity obtained by varying the threshold while keeping all other parameters fixed therefore does not indicate the highest possible sensitivity achievable with these algorithms.

ROC curves for UCHIME reference mode and ChimeraSlayer

ROC plots for UCHIME and ChimeraSlayer on a representative set chosen from SIM2 are shown in Fig. S3. We observe that the UCHIME curve is above the ChimeraSlayer curve, indicating better accuracy for UCHIME. This means that for a given error rate, UCHIME has higher sensitivity, and for a given sensitivity, UCHIME has a lower error rate. On this particular set, the error rate of UCHIME is higher than ChimeraSlayer with default parameters, though the average UCHIME error rate over all SIM2 sets is lower than ChimeraSlayer (see Table S1).

ROC curve for Global-X vs. Local-X

ROC curves for UCHIME Global-X search (the default) and Local-X search on the same subset of SIM2 presented in Fig. S4. Since Global-X is a special case of Local-X, the maximum sensitivity of Local-X is necessarily greater than or equal to the maximum sensitivity of Global-X. However, this ROC analysis is typical in that we usually find Local-X to have error higher rates at a given sensitivity when error rates are in a practically useful range (say, <5%).

ROC curves for UCHIME *de novo* mode and Perseus

ROC plots for UCHIME, PerseusD and Perseus are shown in Fig. S5. PerseusD is a variant of the original Perseus algorithm that follows UCHIME by only testing parents that have been classified as non-chimeric and are at least twice as abundant as the query.

Fraction of chimeras in published datasets

We used UCHIME and ChimeraSlayer to screen selected published 16S-based sequencing surveys obtained from the RDP database (Cole et al, 2009) for the presence of chimeras; results are shown in Fig. S6. We chose to use the same datasets previously analyzed in (Haas et al, 2011). We observe an increase in the number of chimeras predicted by UCHIME, consistent with the increase in sensitivity over ChimeraSlayer found using simulated data (see main text).

UCHIME in practice

Reference database methods vs. de novo *methods*

Reference-based and *de novo* methods have different requirements and assumptions and in general are not directly comparable. *De novo* mode requires estimated amplicon sequences and abundances obtained by a single amplification stage, while reference database mode can operate on any type of sequence. It is not possible to evaluate a *de novo* method on the SIM2 benchmark because amplification is not simulated, so abundances are not available. While it is possible to use a reference-based method on the mock datasets, results will vary depending on which reference database is chosen. In real experiments, the reference database will probably be incomplete due to unknown species and unknown copies of the gene in known species, and tests of reference-based methods on mock datasets may therefore tend to report unrealistically high sensitivity.

Reference database mode

The reference database mode of UCHIME implicitly assumes that the database contains high-quality sequences close to the true biological sequences in the sample. The most common problems with a

reference database approach are: (i) the lack of a suitable reference database, (ii) inadequate phylogenetic coverage of the community being studied in available databases, and (iii) poor-quality sequences in available databases.

In practice, reference databases will usually be incomplete, and false negatives should be expected due to missing parents. Unknown species will of course be absent. Even if a given species has a high-quality reference sequence, it may have additional copies of the sequenced gene due to copies (paralogs, pseudo-genes or segmental duplications) that are absent from the database. Phylogenetic coverage should therefore be not understood not just in terms of species, but also considering of all sequences in the community that are homologous to the gene and match the chosen primers.

A false negative will occur if the query sequence is a chimera and the database contains a sufficiently similar chimera. Noisy reference sequences can cause both false negatives and false positives. Noise can reduce the score of a valid chimeric model below the *h* threshold, creating a false negative. To see how noisy sequences can produce false positives, let X be a correct biological sequences, X_L be a prefix of X, X_R be a suffix of X and X' be a "noisy" copy of X, i.e. a copy of X with spurious substitutions and/or indels. Suppose there are two noisy copies of X^1 and X^2 in the database with asymmetric noise, such that X^1 has more noise on the left and X^2 has more noise on the right, i.e. $X^1 = X'_L X_R$, $X^2 = X_L X'_R$. Then a good copy of X may appear to be a chimera $X = X^2_R X^1_L$ formed from parents X^1 and X^2. If X' and a chimera $C = X_L Y'_R$ are present in the reference database, but not Y, this can cause a false positive identification of Y, which may appear to be a chimera formed as $Y = X'_L C_R$.

Correct sequences in the reference database may give rise to false positives if evolutionary rates in different regions of the gene vary in different lineages. Suppose the gene contains two regions r_1 and r_2, and there are three lineages A, B and C where r_1 evolves faster in A than in B or C, and r_2 evolves faster in B than in A or C. Now suppose the database contains A and B but not C, then C may appear to be a chimera formed from A and B.

These considerations present conflicting goals in the design of a reference database: high phylogenetic coverage and high-quality sequences. Increased phylogenetic coverage generally requires incorporating sequences from unfinished genomes and/or from environmental sequencing studies, both of which tend to have higher error rates than finished genomes. This can be mitigated by using the reference database mode of UCHIME to check a candidate reference database against itself using self mode, as described in the next section.

Self mode for database screening

The self mode (*--self* option) is used when the same file is used for both query and reference. This can be used to screen databases for chimeras. This option causes the query sequence to be excluded as a possible parent, otherwise all sequences would trivially be annotated as non-chimeric due to self-matches. Hits reported using *--self* are 3-way alignments in which either one or two of the sequences are putative chimeras. It should not be assumed that the query sequence is the chimera in this case. Further evidence is required to determine which, if any, of the sequences in the 3-way alignment are PCR artifacts. For example, if two of the sequences are derived from high-quality, finished genomes and the third is from an environmental sequencing study, then the third is most likely to be an artifact and should be discarded from the database. Any remaining sequences found in 3-way alignments can be annotated as unresolved. Hits to experimental data that have an unresolved parent can be treated differently. Whether they should be included or discarded depends on the goals of the study, which will determine the relative importance of sensitivity and specificity of chimera detection. Discarding questionable hits will tend to improve specificity at the expense of sensitivity; including them will tend to improve sensitivity at the expense of specificity.

It is often the case that a reference database contains full-length sequences while a shorter region is sequenced. Here it may be advantageous to trim the database to the shorter region. This can improve computational efficiency because the time required to make a dynamic programming alignment scales with the square of the sequence length (Durbin et al, 1998). This may also reduce the number of false negatives due to failures to identify the correct parent which may be caused by the word-counting heuristic filter (Edgar, 2010) that is used to increase search speed in both the public-domain and USEARCH implementations of UCHIME.

De novo *mode*

The *de novo* mode of UCHIME assumes (i) input sequences correspond to unique sequences in the amplified sample, (ii) the abundances of those sequences have been estimated with sufficient accuracy, (iii) errors due to amplification and sequencing can be neglected, i.e. are adequately suppressed preprocessing of the sequences and/or by the UCHIME scoring function, and (iv) chimeras have abundance less than their parents, as specified by the abundance skew parameter. At the present time, it is not known how well these assumptions hold in practice, except for the mock communities described in the main text. It is an open research problem to determine how predictive these mock communities are of experiments on natural communities.

An advantage of the *de novo* approach is that we expect most or all parent sequences to be present in the reads, which may enable higher sensitivity to be achieved compared with a pre-existing reference database, which will generally be incomplete. A disadvantage of *de novo* mode is that an estimate of unique amplicon abundances is required, which may not be readily available.

The process of estimating unique amplicon sequences and their abundances from a set of reads is called *denoising*. Denoising is a challenging algorithmic problem in itself, and is a rapidly moving target as sequencing technologies evolve. Currently available methods for denoising include PyroNoise (Quince et al., 2009) or AmpliconNoise (Quince et al., 2011) for 454 flowgrams, or clustering methods such as UCLUST (Edgar, 2010) which can be applied to any set of reads.

Consistency check

Where possible, we recommend that the reference database mode and *de novo* modes be used to check each other. We would consider hits found by both methods to be more reliable than hits reported only by one method, though this assumption has not yet been validated. While the mock communities considered in the main test could potentially have been used to test this idea, it turns out that they have too few false positives to give statistically informative results.

A hit found by the reference database mode but not by *de novo* mode can be investigated by searching the estimated amplicons for the putative parent sequences. If these are present in the reads, then this is probably a false negative by the *de novo* mode, which could be due to poor estimates of amplicon sequences or abundances, a preceding false positive that incorrectly identified a parent as a chimera, or a violation of the assumption that the parents have higher abundance. If the parents are not found in the reads, then this could be a false positive by the reference database mode (see previous discussion of causes of false positives in this mode).

A hit found by *de novo* mode but not by reference database mode may be explained by a missing parent sequence in the reference database, which can be verified by searching the reference database for the parents predicted by *de novo* mode.

Sensitivity vs. specificity

The user can trade sensitivity against specificity by adjusting the score threshold (*h* parameter, *--minh* command line option). It is difficult to predict the sensitivity or specificity that will be obtained for a given experiment with a given score threshold. When considering whether sensitivity or specificity is more important in a given experiment, it should be noted that while chimeric amplicons may be relatively

rare in the amplicon pool, they may represent a large fraction of unique amplicon sequences. For example, in the Uneven1 mock community described in the main text, chimeras accounted for 898/992 = 91% of the unique sequences after denoising. This can arise because there are many possible pairs of parent sequences, and each pair could potentially combine in many different ways to create distinct chimeras. While 91% may not be representative of results obtained with natural communities, we would still expect a substantial fraction of the unique sequences to be chimeric. This is supported by our observation that up to 32% of the sequences in selected RDP datasets are predicted to be chimeric by UCHIME (Fig. S6).

Computational efficiency

Community sequencing experiments often produce very large numbers of reads that can be computationally expensive to process. It is generally recommended that the number of sequences be reduced before running UCHIME in order to save computational resources. Preprocessing steps can include dereplication (removing identical sequences), denoising (attempting to correct sequencing error) and data reduction (clustering at, say, 98% identity to reduce experimentally irrelevant variation in the sequences), as illustrated in Fig. S2.

In the case of *de novo* mode, preprocessing of raw reads is always required in order to estimate amplicon sequences and abundances. The estimated number of unique amplicons is usually much smaller than the number of reads, reducing the computational cost of downstream stages in an analysis pipeline, such as UCHIME.

Computational cost can also be significantly reduced by using the USEARCH (Edgar, 2010) implementation of the UCHIME algorithm. The most expensive step in UCHIME is generally searching the reference database. The implementation of UCHIME in the usearch package (http://drive5/com/usearch) exploits the highly optimized USEARCH algorithm for the database search step, which often results in significantly improved execution times. As noted in the main text, UCHIME results are generally not sensitive to the details of the database search method, and the USEARCH implementation therefore gives very similar results to the public-domain version.

In reference database mode, execution time for UCHIME scales approximately linearly with the reference database size and number of query sequences, and like the square of the sequence length (due to the dynamic programming step required for alignment). In *de novo* mode, time scales linearly with the

number of query sequences, linearly with the number of non-chimeric sequences identified in the input, and with the square of the sequence length.

Paired-end reads

At the time of writing, UCHIME does not explicitly support paired-end reads. Work is in progress to add support for pair-end reads in a future version of the algorithm.

Providing that the gap between the ends is short, a reasonable strategy would be to concatenate the two ends. Longer gaps are likely to result in substantially increased false negative rates. It is not recommended to use the common practice of representing the gap between the ends using the corresponding number of Ns as UCHIME will consider these to be differences between the chimera and the parents which will usually result in a false negative. An alternative would be to fill the gap with a consensus sequence obtained by a multiple alignment of the top hits to the query sequence. Note that by default, gapped positions are not considered differences, and simple concatenation without using Ns may therefore be more effective.

Parameter tuning

As noted in the main text, the scoring function used in UCHIME is *ad hoc*. It was initially developed by seeking a simple, closed-form analytic function that approximated the bootstrap sampling method in ChimeraSlayer which had previously been shown to perform well on simulated chimeras. No explicit strategies are employed to suppress particular types of false positive. It is possible that improved performance might be achieved via explicit modeling of a particular experimental protocol, e.g. of site-specific rates in the sequenced gene, error characteristics of the chosen sequencing technology (e.g., read-position-specific base call and homopolymer error rates), etc. The default parameters of UCHIME were tuned to give lower error rates and higher sensitivity than ChimeraSlayer on the SIM2 benchmark. This strategy was chosen in order to demonstrate that UCHIME has better performance than ChimeraSlayer on a published benchmark (Haas et al., 2011) on which ChimeraSlayer was shown to be superior to previous methods and thereby establish that UCHIME is superior to all previously published methods. We believe that while these parameters probably represent reasonable default settings, different parameters may be optimal in some applications. It should be noted that the ChimeraSlayer validation emphasized sensitivity to closely related parents: the divergence measure used by Haas *et al.* is the distance D between the parents A and B (D = 100% – id(A,B)), while in this work we use the identity T between the chimera Q and the closest parent (T = 100% – max { id(Q,A), id(Q, B) } in the case of bimeras). Generally we expect that $T \leq D/2$ since at last half of the bimera will be identical to the closer parent. In many experiments, it is T rather than D that indicates whether the chimera is experimentally relevant. For

example, if the goal is to identify OTUs by clustering at 97%, and a parent is successfully identified as the representative sequence for a cluster, then a chimera with $T \leq 3\%$ should be assigned to the parent cluster and will not create a spurious OTU. Such a chimera could have $D \geq 6\%$, and conversely a chimera with D=6% could have arbitrarily small T and thus fall inside a 3% cluster radius. By default, the minimum T divergence, set by the *--mindiv* option of UCHIME, is set to 0.8% to allow detection of chimeras with small D, which is required to achieve good performance on SIM2. Chimeras with divergence $T \gtrsim 0.8\%$ may have very small numbers of diffs and hence be difficult to discriminate from false positives, requiring a higher h threshold to suppress errors. These considerations suggest that in a typical OTU clustering experiment, higher sensitivity to experimentally relevant chimeras could be achieved with acceptable false positive rates by increasing *--mindiv* and reducing h (*--minh* option) and/or β (*--xn* option). In addition, SIM2 has no multimeras and adds simulated noise that is designed to indicate the general impact of sequencing error and natural variation on performance rather than to accurately model errors due to a given sequencing technologies or to model natural biological variation that can cause a reference sequence to differ from the true parent sequence. Ideally, parameters would be re-tuned on a benchmark that is tailored to the details of a particular experiment, including simulated errors based on estimates of error rates of the chosen sequencing technology. Designing and implementing such a benchmark would be challenging. Further work is needed to determine whether and how parameters should be varied according to the details of a particular experiment.

References

Cole, J. R., Q. Wang, E. Cardenas, J. Fish, B. Chai, R. J. Farris, A. S. Kulam-Syed-Mohideen, D. M. McGarrell, T. Marsh, G. M. Garrity, and J. M. Tiedje. (2009), The Ribosomal Database Project: improved alignments and new tools for rRNA analysis. *Nucleic Acids Res.* **37**, D141-D145.

Durbin, R., Eddy, S., Krogh, A. and Mitchison, G. (1998) Biological Sequence Analysis. Cambridge University Press.

Edgar,R.C. (2010) Search and clustering orders of magnitude faster than BLAST, *Bioinformatics* **26**(19), 2460-1.

Haas, B.J., Gevers, D., Earl, A.M., Feldgarden, M., Ward, D.V., Giannoukos, G., Ciulla, D., Tabbaa, D., Highlander, S.K., Sodergren, E., Methe, B., Desantis, T.Z., Petrosino, J.F., Knight, R. and Birren, B.W. (2011) Chimeric 16S rRNA sequence formation and detection in Sanger and 454-pyrosequenced PCR amplicons, *Genome Res*, **21**, 494-504.

Mason, S.J. and Graham, N.E. (2002) Areas beneath the relative operating characteristics (ROC) and relative operating levels (ROL) curves: Statistical significance and interpretation, *Q. J. Meteorol. Soc.*, **128**, 2145-2166.

Quince, C., Lanzen, A., Curtis,T.P., Davenport,R.J., Hall,N., Head,I.M., Read.,L.F. and Sloan,W. (2009) Accurate determination of microbial diversity from 454 pyrosequencing data, *Nature Methods*, **6**(9), 639-641.

Quince, C., Lanzen, A., Davenport, R.J. and Turnbaugh, P.J. (2011) Removing noise from pyrosequenced amplicons, *BMC Bioinformatics*, **12**, 38.

Supplemental Tables

Len	Evo	CS Sens	UC Sens	Diff	CS Err	UC Err	Diff
200	-	70.7%	72.7%	+2.8%	1.6%	1.1%	+0.5%
200	mm1	38.6%	69.6%	+80.2%	0.3%	0.7%	-0.4%
200	mm2	24.6%	65.9%	+168.4%	0.1%	0.6%	-0.5%
200	mm3	16.6%	61.7%	+271.6%	0.0%	0.6%	-0.6%
200	mm4	9.6%	57.9%	+500.8%	0.0%	0.4%	-0.4%
200	mm5	5.9%	53.1%	+797.3%	0.0%	0.3%	-0.3%
200	ind1	60.4%	66.6%	+10.4%	1.4%	0.6%	+0.8%
200	ind2	52.2%	59.9%	+14.8%	0.8%	0.6%	+0.3%
200	ind3	40.8%	51.3%	+25.7%	0.8%	0.4%	+0.5%
200	ind4	30.3%	41.1%	+35.6%	0.4%	0.3%	+0.1%
200	ind5	20.7%	29.6%	+43.0%	0.4%	0.3%	+0.2%
300	-	77.5%	81.3%	+4.9%	1.9%	1.9%	+0.0%
300	mm1	55.5%	78.5%	+41.4%	0.4%	1.4%	-1.0%
300	mm2	45.0%	74.8%	+66.1%	0.2%	1.0%	-0.9%
300	mm3	37.1%	71.8%	+93.5%	0.1%	0.8%	-0.7%
300	mm4	28.1%	67.8%	+141.1%	0.0%	0.5%	-0.5%
300	mm5	20.5%	64.4%	+213.8%	0.0%	0.4%	-0.4%
300	ind1	66.6%	76.4%	+14.8%	1.9%	1.3%	+0.6%
300	ind2	62.1%	70.3%	+13.2%	1.4%	0.9%	+0.5%
300	ind3	57.2%	64.0%	+11.9%	1.2%	0.6%	+0.7%
300	ind4	49.4%	56.1%	+13.7%	0.9%	0.3%	+0.6%
300	ind5	38.8%	46.6%	+20.4%	0.7%	0.4%	+0.3%
FL	-	90.3%	90.8%	+0.5%	1.0%	0.3%	+0.7%
FL	mm1	87.4%	90.4%	+3.3%	0.4%	0.2%	+0.3%
FL	mm2	83.9%	89.9%	+7.2%	0.4%	0.1%	+0.3%
FL	mm3	82.2%	89.3%	+8.7%	0.2%	0.0%	+0.2%
FL	mm4	79.8%	87.2%	+9.2%	0.2%	0.0%	+0.1%
FL	mm5	77.9%	83.8%	+7.6%	0.1%	0.0%	+0.1%
FL	ind1	83.6%	94.3%	+12.7%	0.9%	0.3%	+0.6%
FL	ind2	81.4%	90.3%	+10.9%	0.8%	0.1%	+0.7%
FL	ind3	79.5%	84.6%	+6.4%	0.7%	0.0%	+0.7%
FL	ind4	75.4%	77.2%	+2.3%	0.5%	0.0%	+0.5%
FL	ind5	72.5%	70.1%	-3.3%	0.4%	0.0%	+0.4%
Total		54.6%	70.6%	+129%	0.62%	0.49%	+0.13%

Table S1. UCHIME and ChimeraSlayer results on the SIM2 benchmark

The SIM2 benchmark was used to train both UCHIME and ChimeraSlayer. Columns are: Len=sequence length, FL=full-length 16S genes; Evo=added mutations, sub*n* means *n*% substitutions, ind*n* means *n*% indels were added; CSSens=ChimeraSlayer sensitivity; UCSens=UCHIME sensitivity; UC/CS = 100% × UCSens/CSSens; CSErr=ChimeraSlayer error rate; UCErr=UCHIME error rate; CS-UC= CSErr – UCErr. The total average error rate of UCHIME is slightly lower than ChimeraSlayer (0.49% vs. 0.62%), while the average sensitivity is 16% higher (70.6% vs. 54.6%), with higher sensitivity on all individual sets except one: full-length sequences 5% added indels.

		M2		M3		M4		All_M	
Div.	Evo	CS	UC	CS	UC	CS	UC	CS	UC
97-99	-	64.0	89.0	26.0	55.0	12.0	34.0	34.0	59.3
97-99	i1	59.0	77.0	21.0	47.0	11.0	29.0	30.3	51.0
97-99	i2	51.0	57.0	24.0	30.0	10.0	24.0	28.3	37.0
97-99	i3	37.0	45.0	21.0	32.0	9.0	14.0	22.3	30.3
97-99	i4	35.0	23.0	15.0	16.0	8.0	7.0	19.3	15.3
97-99	i5	25.0	21.0	12.0	15.0	7.0	6.0	14.7	14.0
97-99	m1	27.0	83.0	10.0	52.0	8.0	31.0	15.0	55.3
97-99	m2	13.0	73.0	9.0	48.0	3.0	23.0	8.3	48.0
97-99	m3	6.0	66.0	5.0	38.0	1.0	21.0	4.0	41.7
97-99	m4	7.0	53.0	2.0	27.0	3.0	18.0	4.0	32.7
97-99	m5	1.0	42.0	1.0	20.0	0.0	15.0	0.7	25.7
95-97	-	91.0	100.0	58.0	79.0	29.0	62.0	59.3	80.3
95-97	i1	88.0	100.0	53.0	71.0	23.0	50.0	54.7	73.7
95-97	i2	78.0	94.0	46.0	57.0	23.0	44.0	49.0	65.0
95-97	i3	77.0	79.0	38.0	50.0	21.0	36.0	45.3	55.0
95-97	i4	56.0	64.0	35.0	40.0	18.0	26.0	36.3	43.3
95-97	i5	46.0	54.0	26.0	35.0	16.0	19.0	29.3	36.0
95-97	m1	59.0	99.0	28.0	74.0	17.0	58.0	34.7	77.0
95-97	m2	37.0	98.0	14.0	70.0	5.0	49.0	18.7	72.3
95-97	m3	19.0	92.0	10.0	63.0	5.0	40.0	11.3	65.0
95-97	m4	17.0	90.0	3.0	56.0	1.0	40.0	7.0	62.0
95-97	m5	8.0	84.0	0.0	51.0	0.0	35.0	2.7	56.7
90-95	-	98.0	100.0	88.0	93.0	67.0	88.0	84.3	93.7
90-95	i1	93.0	97.0	85.0	87.0	60.0	86.0	79.3	90.0
90-95	i2	90.0	97.0	78.0	82.0	56.0	73.0	74.7	84.0
90-95	i3	87.0	96.0	67.0	74.0	47.0	69.0	67.0	79.7
90-95	i4	74.0	92.0	52.0	70.0	43.0	58.0	56.3	73.3
90-95	i5	58.0	81.0	37.0	62.0	26.0	49.0	40.3	64.0
90-95	m1	93.0	100.0	72.0	89.0	49.0	86.0	71.3	91.7
90-95	m2	74.0	100.0	60.0	87.0	45.0	77.0	59.7	88.0
90-95	m3	72.0	100.0	38.0	86.0	27.0	74.0	45.7	86.7
90-95	m4	45.0	98.0	32.0	80.0	23.0	73.0	33.3	83.7
90-95	m5	36.0	96.0	31.0	79.0	11.0	73.0	26.0	82.7
90-99	-	84.3	96.3	57.3	75.7	36.0	61.3	59.2	77.8
90-99	i1	80.0	91.3	53.0	68.3	31.3	55.0	54.8	71.6
90-99	i2	73.0	82.7	49.3	56.3	29.7	47.0	50.7	62.0
90-99	i3	67.0	73.3	42.0	52.0	25.7	39.7	44.9	55.0
90-99	i4	55.0	59.7	34.0	42.0	23.0	30.3	37.3	44.0
90-99	i5	43.0	52.0	25.0	37.3	16.3	24.7	28.1	38.0
90-99	m1	59.7	94.0	36.7	71.7	24.7	58.3	40.3	74.7
90-99	m2	41.3	90.3	27.7	68.3	17.7	49.7	28.9	69.4
90-99	m3	32.3	86.0	17.7	62.3	11.0	45.0	20.3	64.4
90-99	m4	23.0	80.3	12.3	54.3	9.0	43.7	14.8	59.4
90-99	m5	15.0	74.0	10.7	50.0	3.7	41.0	9.8	55.0

Table S2. UCHIME and ChimeraSlayer results on the SIMM dataset.

The SIMM dataset contains 900 simulated *m*-meras, divided into three divergence bins (97-99%, 95-97% and 90-95%) and by *m* (*m*=2, 3 and 4) for a total of nine bins, each with 100 simulated *m*-meras. Ten

modified versions of this dataset were created with from 1% to 5% substitutions and indels, respectively. The above table shows the sensitivity of UCHIME (UC) and ChimeraSlayer (CS) on this data. Columns are: Div=divergence range, Evo=i*n* means *n*% indels were added, m*n* means *n*% substitutions; CS*m* is the number of *m*-meras found by ChimeraSlayer,; UC*m* is the number of *m*-meras found by UCHIME; CSAll and UCAll are the total numbers found, and CSPct and UCPct are the totals found as a percentage. We observe that UCHIME has higher sensitivity in all cases, with increasing improvements in more challenging sets having larger *m*, smaller divergence and higher levels of noise.

Supplemental Figures

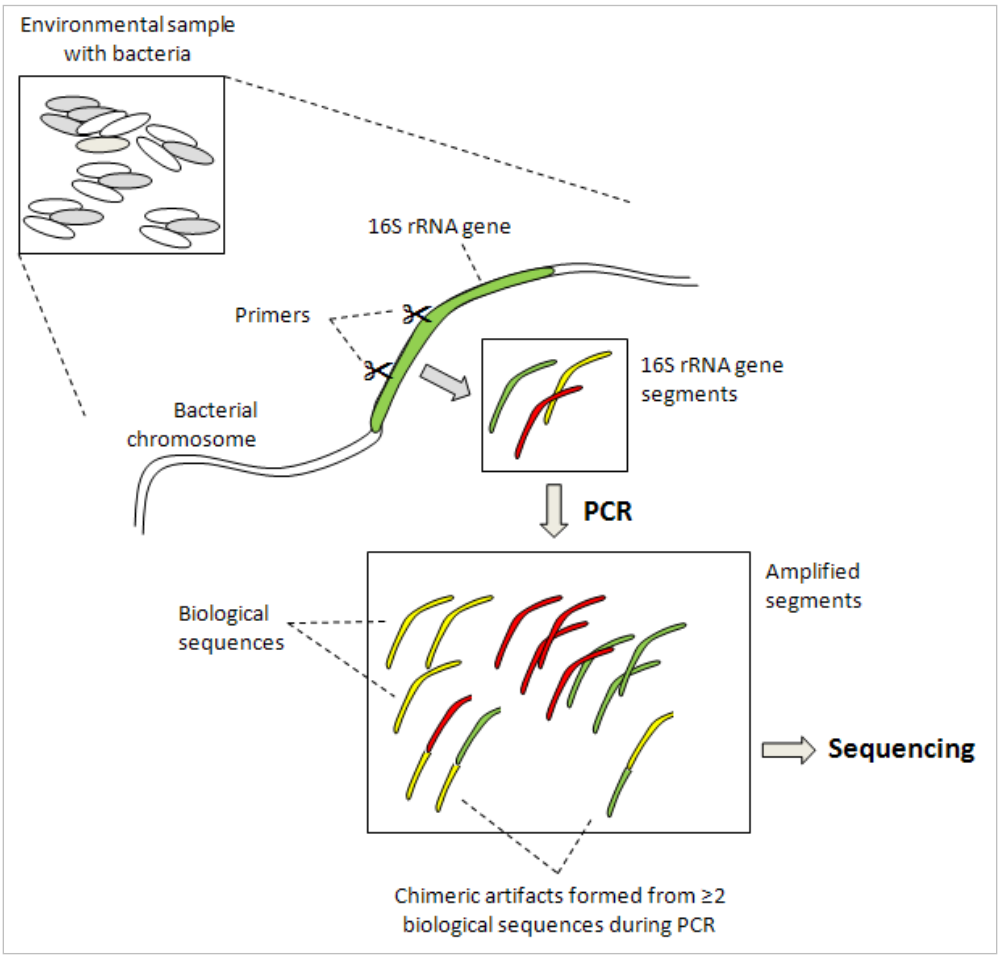

Figure S1. Schematic summary of a typical 16S environmental sequencing experiment.

Primers are used to extract a short segment of the 16S gene in an environmental sample. Primers are chosen to match highly conserved regions in the gene. The segment between the primers is short enough that the segment can be sequenced with a single read in a current "next-generation" sequencer, and includes variable regions that enable taxonomic identification. Chimeras form in the PCR stage used to amplify segments prior to sequencing.

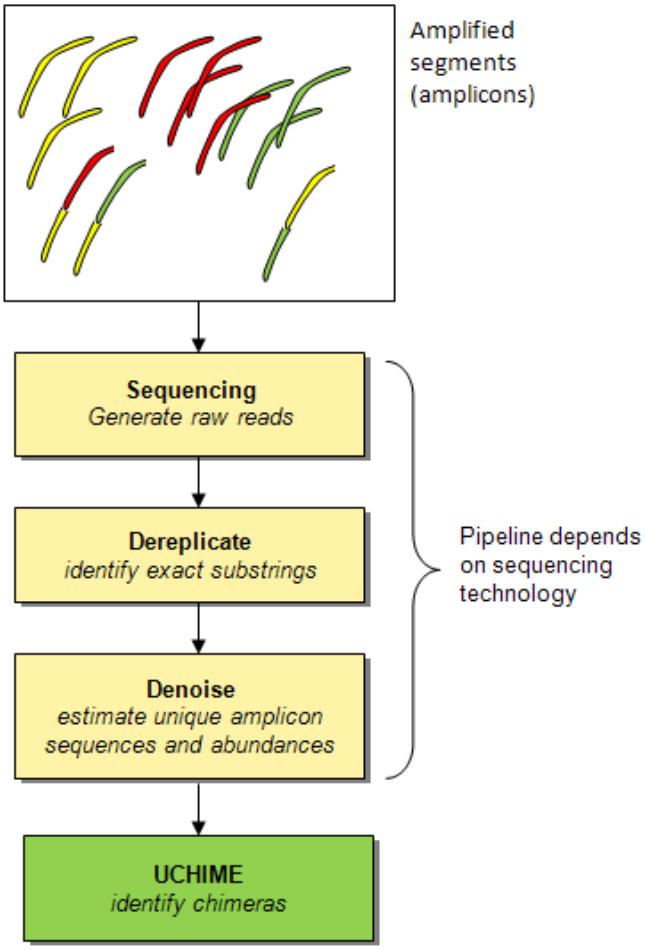

Figure S2. Schematic summary of a typical 16S sequence analysis pipeline with chimera filtering.

Prior to chimera identification, raw reads are usually quality-filtered, dereplicated and denoised. Input to UCHIME *de novo* mode is a set of estimated amplicon sequences and abundances generated by the denoising stage. In reference database mode, input sequences could be chimera-filtered at any stage in the pipeline and abundances are not needed or used, though in practice chimera filtering would usually be done after denoising as the number of sequences is usually greatly reduced and the computational resources required to run UCHIME are correspondingly less.

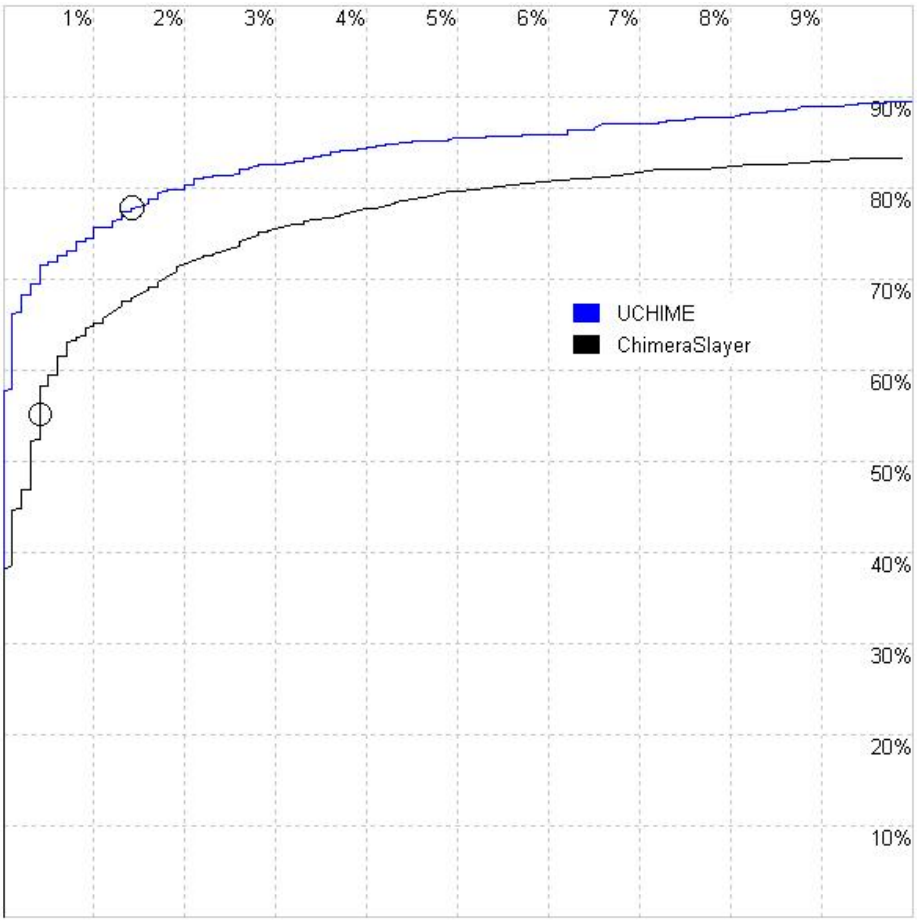

Figure S3. ROC curves for UCHIME and ChimeraSlayer on length 300 sets in SIM2 with 1% substitutions.

ROC curves obtained on a representative SIM2 set. Horizontal axis is error rate (% false positives), vertical axis is sensitivity (% true positives). Open circles indicate the default score threshold (h=0.5 for UCHIME, 90% bootstrap confidence for ChimeraSlayer). We observe that the UCHIME curve is consistently above the ChimeraSlayer curve, indicating better accuracy for UCHIME. In this case, the error rate of UCHIME is higher than ChimeraSlayer with default parameters, though the average error rate over all of SIM2 is lower for UCHIME.

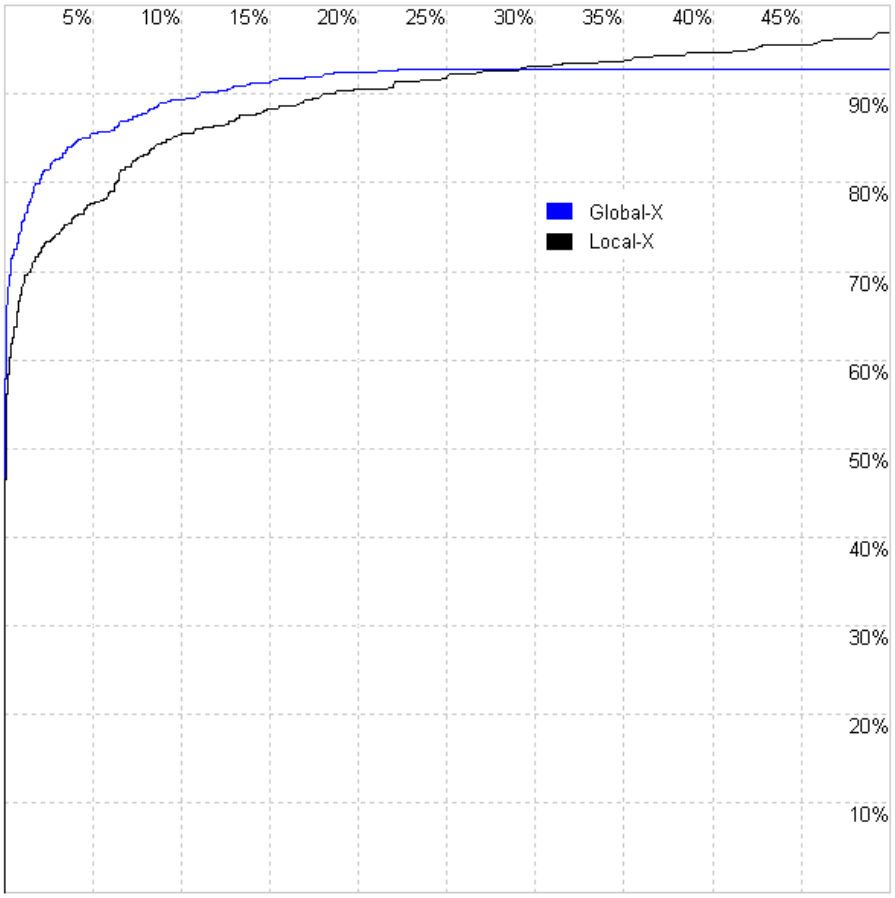

Figure S4. ROC curves for UCHIME Global-X and Local-X on length 300 sets in SIM2 with 1% substitutions.

ROC curves for UCHIME Global-X search (the default) and Local-X search on the same subset of SIM2 presented in Fig. S2. The upper blue curve for UCHIME is thus the same in both figures, with the error range expanded here to show the intersection between the Global-X and Local-X curves. Since Global-X is a special case of Local-X, the maximum sensitivity of Local-X is necessarily greater than or equal to the maximum sensitivity of Global-X. However, this ROC curve is typical in that we usually find Local-X to have error higher rates at a given sensitivity when error rates are in a practically useful range (say, <5%).

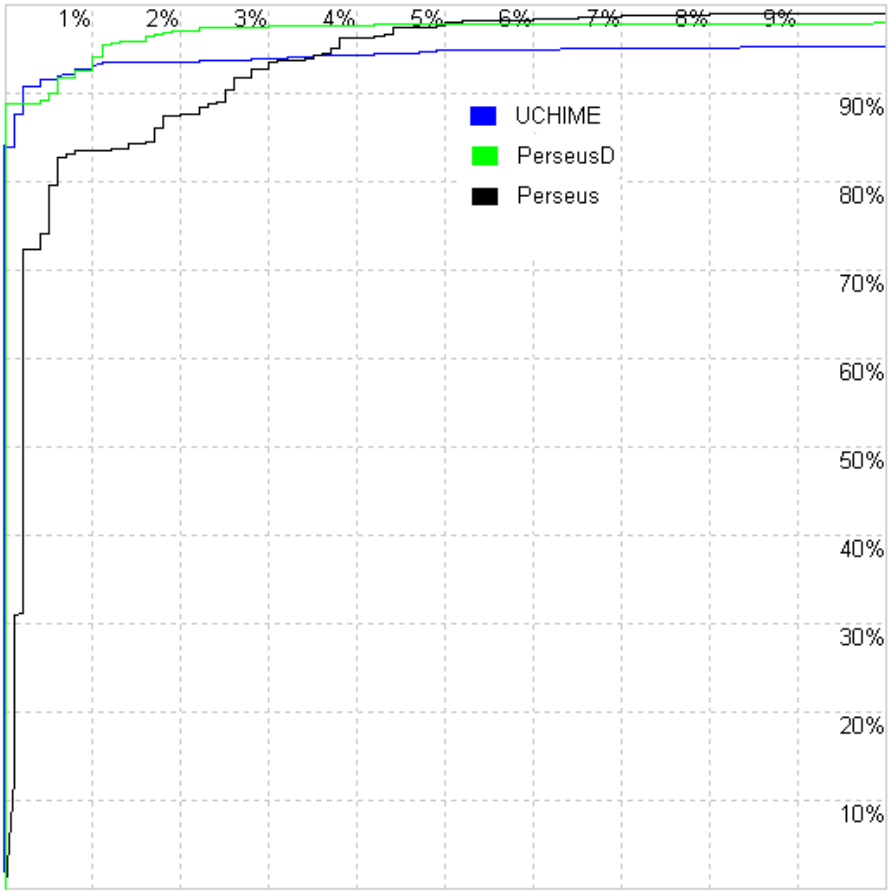

Figure S5. ROC curves for UCHIME, PerseusD and Perseus on all MOCK datasets.

These curves were obtained using UCHIME, Perseus and PerseusD on all MOCK datasets, some of which were used for training Perseus (UCHIME was trained on SIM2). Default parameters give (TP%, FP%): UCHIME 90.7%, 0.2%; PerseusD 90.5%, 0.6% and Perseus 86.7%, 1.8%. These results show that in the region with ≤ 1% errors, UCHIME and PerseusD have similar performance, while PerseusD is clearly better than Perseus.

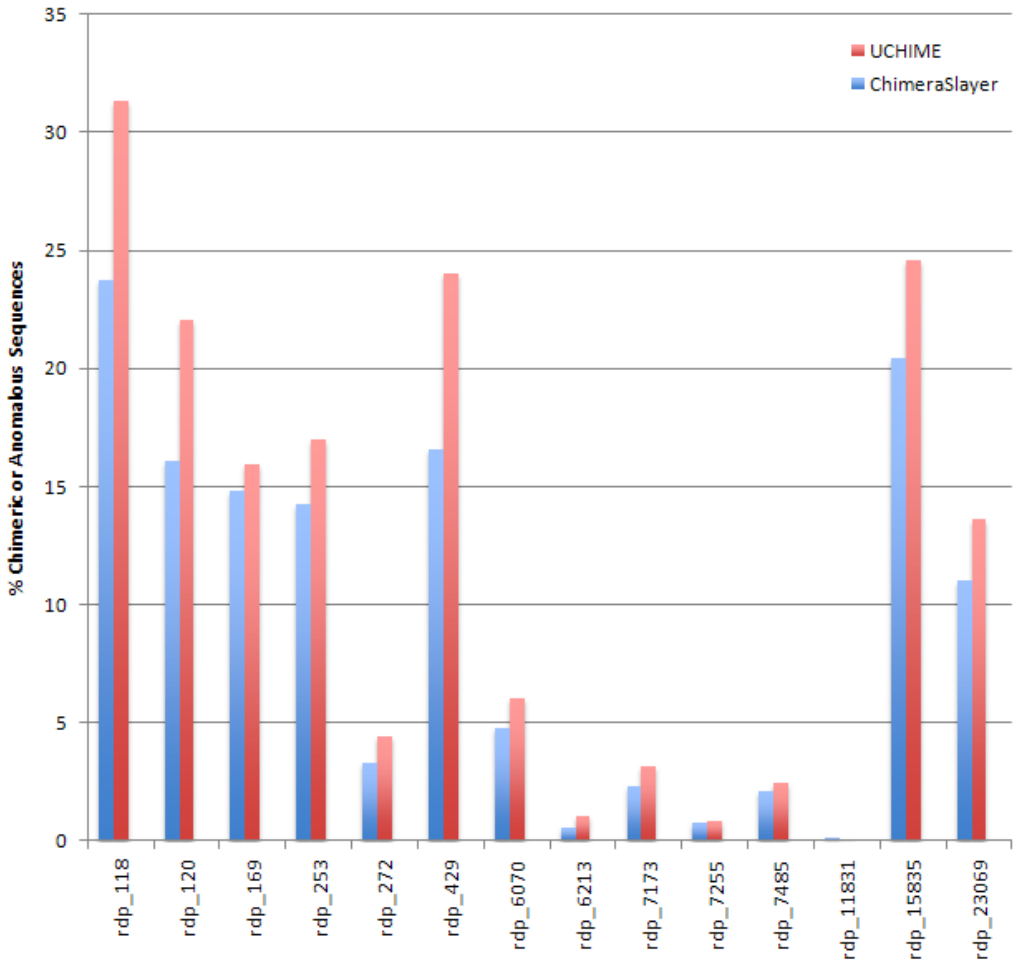

Figure S6. Fraction of chimeric sequences reported by UCHIME and ChimeraSlayer in selected datasets from the RDP database.

Previously published nearly full-length 16S sequence data sets were obtained from the Ribosomal Database Project website (Cole et al, 2009) and examined for chimeras using UCHIME and ChimeraSlayer, using the same data previously analyzed by ChimeraSlayer and WigeoN as reported in (Haas et al. 2009), supplemental figure SS1. An increase in the number of predicted chimeras is observed, consistent with the improved sensitivity of UCHIME on simulated data (see main text).