Accuracy and precision of 16S rRNA gene sequence assignments to OTUs is more important than the stability of OTUs

Patrick D. Schloss

October 02, 2015

- possible to include commas for large numbers?

Introduction

The ability to affordably generate millions of 16S rRNA gene sequences has allowed microbial ecologists to thoroughly characterize the microbial community composition of hundreds of samples. To simplify the complexity of these large datasets, it is helpful to cluster sequences into meaningful bins. These bins, known as operational taxonomic units (OTUs), are commonly used to compare the biodiversity contained within and between different samples. Such comparisons have enabled researchers to characterize the microbiota associated with the human body (REF), soil (REF), aquatic ecosystems (REF), and numerous other environments. Within the field of microbial ecology a convention has emerged where sequences are clustered into OTUs using a threshold of 3% dissimilarity or 97% similarity. One advantage of the OTU-based approach is that the definition of the bins is operational and can be changed to suit the needs of the particular project. However, with the dissemination of clustering algorithms within packages such as mothur and QIIME and stand-alone tools (REFS), it is important to understand how different clustering methods implement this conventional OTU threshold. Furthermore, it is necessary to understand how algorithm choice affects the precision and accuracy of assigning sequences to OTUs. Broadly speaking, three approaches have been developed to assign sequences to OTUs.

The first approach has been referred to as phylotyping or closed reference clustering. This approach involves comparing sequences to a curated database and then clustering sequences together that are similar to the same reference sequence. Reference-based clustering methods suffer when the reference sequences do not reflect the composition of the community. If a large fraction of sequences are novel, then they cannot be assigned to an OTU. In addition, the reference sequences used in this application are selected because they are less than 97% similar to each other over the full length of the gene; however, it is known that the commonly used variable regions within the 16S rRNA gene do not evolve at the same rate as the full-length gene. Thus, a sequence representing a fragment of the gene may be more than 97% similar to multiple reference sequences. Because of this, defining OTUs in the closed-reference approach is complicated because two sequences might be 97% similar to the same reference sequence; however, they may only be 94% similar to each other. The strength of the reference based approach is that the methods are generally fast, scaling linearly with the number of sequences being clustered. A subtle alternative to this approach is to use a classifier to assign a taxonomy to each sequence so that sequences can be clustered at a desired level in the Linnean taxonomic hierarchy. Ultimately, this approach is an example of supervised classification.

The second approach has been referred to as *de novo* clustering. In this method, the similarity between sequences is used to cluster sequence rather than the similarity to a reference database. In contrast to the efficiency of closed-reference clustering, the speed of hierarchical *de novo* clustering algorithms scale quadratically with the number of unique sequences. The expansion in sequencing throughput combined with sequencing errors inflates the number of unique sequences resulting in the need for large amounts of memory and time to cluster the sequences. If error rates can be reduced through stringent quality control measures then these problems can be overcome (REF). As an alternative, heuristics have been developed to approximate the clustering of hierarchical methods. Previous comparisons of hierarchical and heuristic algorithms have shown that the average neighbor algorithm, a hierarchical algorithm outperforms the other *de novo* clustering algorithm. An early analysis using 16S rRNA gene sequence data generated using the 454 sequencing platform found that if these methods were applied to the same dataset using a subset of the sequences then a different number of OTUs were observed compared to clustering the full dataset and rarefying to the same number of sequences. Because the initial subsampling of the dataset was not repeated in the earlier analysis, it is unclear whether the result was product of the algorithm or sampling. One explanation could be that the clusters are getting better with additional sampling. Another observed problem with *de novo* clustering is that because it is necessary to break ties when assigning sequences to OTUs, clustering the same data multiple times may result in different clustering assignments. Whether the differences in assignments is meaningful is unclear; however the variation in results could represent equally valid clustering of the data. The strength of *de novo* clustering is its independence of references for carrying out the clustering step. After clustering, the classification of each sequence can be used to obtain a consensus classification for the OTU (ref). For this reason, *de novo* clustering has been preferred across the field. In contrast to closed-reference clustering, *de novo* clustering is an example of unsupervised clustering.

The third approach is a hybrid of the closed-reference and *de novo* approaches. This approach has been called open-reference clustering. This method involves performing closed-reference clustering followed by *de novo* clustering on those sequences that are not sufficiently similar to the reference. This method should exploit the strengths of both closed-referene and *de novo* clustering; however, the different OTU definitions generated by both approaches poses a possible problem when the methods are combined. An alternative to this approach has been to classify sequences to a bacterial family or genus and then assigned to OTUs within those levels (REF). For example, all sequences assigned to the *Porphyromonadaceae* would then be assigned to OTUs using the average neighbor algorithm. Those sequences that did not classify to a known family would also be clustered using the average neighbor algorithm. An advantage of this approach is that it lends itself nicely to parallelization since each taxonomic group (e.g. each family) is seen as being independent and can be processed separately. Such an approach would overcome the difficulty of mixing OTU definitions between the closed-reference and *de novo* approaches.

The growth in options for assigning sequences using each of these three broad approaches has been considerable. It has been difficult to objectively assess the quality of OTU assignments. Some have focused on the time and memory required to process a dataset (REFS). These are valid parameters to assess when judging a clustering algorithm, but have little to say about the quality of the clustering. Others have attempted to judge the quality of an algorithm by its ability to generate data that parallels classification data. This approach is problematic because bacterial taxonomy often reflects the biases amongst bacterial systematicists and the rates of evolution across lineages are not the same. We recently proposed an approach for evaluating OTU assignments using the similarity between sequences (REF). Those sequences that were similar to each other and found in the same OTU were called true positives while those that were similar and found in different OTUs were called false negatives. Meanwhile, those sequences that were different from each other and found in the same OTU were called false positives and those that were dissimilar and found in different OTUs were called true negatives. Counting the frequency of these different classes allowed us to judge how each method balanced the ratio of true positives and negatives to false positives and ratios using the Matthew's correlation coefficient (MCC; REF). That analysis focused on assessing *de novo* clustering algorithms and found that the average neighbor algorithm outperformed the other hierarchical and heuristic algorithms.

A recent analysis by He and colleagues (REF) attempted to characterize the three general clustering approaches by focusing on what they called stability. They defined stability as the ability of an algorithm to provide the same clustering on a subset of the data as was found in the full dataset. Related to this, the authors expressed concern that because some algorithms use a random number generator to break ties there may be further instability between executions of algorithm. Their concept of stability does not account for the accuracy of the clustering and instead focuses on the precision of the clustering. A method may be very precise, but low in accuracy. In the current analysis, we attempted to assess the accuracy and precision of the various clustering algorithms. Building on our previous analysis of clustering algorithms, our hypothesis was that the algorithms praised by the He study for their stability actually suffered a lack of accuracy. In addition, we assess these parameters in light of sequence quality using the original 454 dataset and a larger and more modern dataset generated using the MiSeq platform.

Results and Discussion

Summary and replication of He study. We sought to identify the more critical analyses performed in the He study. Similar to their study, we obtained the Canadian soil dataset from Roesch et al. (REF) and processed the sequences as described in their analysis. In our opinion there were three important tests.

First, we sought to quantify whether the clustering observed for a subset of the data represented the same clustering that was found with the full dataset. The He study found that when they used the open and closed-reference methods clusters formed using the subset data most closely resembled those of the full dataset. Among the *de novo* clustering methods they observed that the abundance-based greedy clustering (AGC) algorithm followed by single linkage (SL), distance-based greedy clustering (DGC), complete linkage (CL), and average linkage (AL) algorithms. They observed a broad range of MCC values among their AL replicates, which was not explained by the authors. We first sought to assess calculated the MCC for for each of the clustering algorithms using 20, 40, 60, and 80% relative to the clusters formed by the algorithms using the full dataset (Figure 1A). Because a random number generator is used in some of algorithms to break ties where pairs of sequences have the same distance between them, similar to the He study, we replicated each algorithm and subsample 30 times. Across these sequencing depths, we observed that the stability of the SL and CL algorithms were highly sensitive to sampling effort relative to the AL, AGC, and DGC algorithms (Figure 1A). Our results (Figure 1B) largely confirmed those of Figure 4C of He study with one notable exception. In the He study, when comparing the OTUs formed using the AL algorithm with 60% of the data, they observed a mean MCC value of approximately 0.63 (95% confidence interval between approximately 0.15 and 0.75). In contrast, we observed a mean value of 0.93 (95% confidence interval between 0.91 and 0.95). This result suggests that the AL algorithm was far more stabile than indicated in the He study. Regardless, it supports the assertion that the clustering observed for the subset of the data does not represent the same clustering that was found with the full dataset; however, the significance of these differences is unclear.

Second, rarefaction curves calculated using clusters obtained using a portion of the dataset did not overlap with rarefaction curves generated using clusters generated from the full dataset. This result was originally observed in the Roesch study using the complete linkage algorithm and was reproduced in the He study where this result was more pronounced problem when using the CL, SL, and DGC algorithms relative to the other algorithms. The He and Roesch studies both found that the CL algorithm produced fewer OTUs in the subset than in the rarefied data. Expanding the analysis to other algorithms, the He study found that the SL algorithm produced more OTUs, the

AGC produced fewer, and the other algorithms produced similar numbers of OTUs than expected when comparing the subsetted data to the rarefied data. Our results support those of these previous studies (Figure 2). It was clear that inter-method differences were generally more pronounced than the differences observed between rarefying from the full dataset and from clustering the subsetted data. The number of OTUs observed was largest using the CL algorithm, followed by the open-reference algorithm. The AL, AGC, and DGC algorithms all provided comparable numbers of OTUs. Finally, the closed-referene and SL algorithms generated the fewest number of OTUs.

Third, the authors attempted to describe the effects of the clustering instability on comparisons of communities. They used Adonis to test whether the community structure represented in subsetted communities resembled that of the full dataset. Inspection of these results indicates that although they were able to detect significant p-values, they were were of marginal biological significance. Adonis R statistics close to zero indicate the community structures from the full and subsetted datasets overlapped while values of one indicate the communities are completely different. The He study observed adonis R statistics of 0.02 (closed-reference), 0.03 (open-reference), 0.07 (CL, AGC, DGC), and 0.16 (SL and AL). Regardless of the statistical or biological significance of these results, the analysis does not make sense since the *de novo* and open-reference approaches do not consistently label the OTUs that sequences belong to when the clustering algorithms are run multiple times with different random number seeds. To overcome this, the authors selected representative sequences from each OTU and used those representative sequences to link OTU assignments between the different sized sequence sets. The justification for this analysis is specious as the OTU assignments are based on the data available in the dataset when the sequences are clustered and comparing assignments in this manner are irreconcilable. It is not surprising that the only analysis that did not provide a significant p-value was for the closed-reference analysis, which is the only analysis that provides consistent OTU labels. Because this analysis was so poorly designed, we did not seek to reproduce it.

It is worth noting that the entire design of the He study is artificial. First, their analysis was based on a single sample. Researchers generally have dozens or hundreds of samples that are pooled and clustered together to enable comparison across samples. Second, all of the sequence data from these datasets is pooled for a single analysis. No one would ever perform an analysis based on a subset of their data. Because of these points, the value of identifying stable OTUs is unclear.

Methods vary in their accuracy. More important than the stability of OTUs is whether sequences are assigned to the correct OTUs. A method can generate highly stable OTUs, but the OTUs may be meaningless by poorly representing a specified cutoff in the assignment of sequences to those OTUs. To assess the accuracy and precision of the various methods, we made use of the pairwise distance between the unique sequences to count the number of true positives and negatives and the number of false positives and negatives for each method and sampling depth so that we could calculate the average MCC value as a measure of a method's accuracy and its variation as a measure of its precision. We made three important observations. First, each of the *de novo* methods varied in how sensitive their MCC values were to additional sequences (Figure 3A). The SL and CL algorithms were the most sensitive; however, the MCC values using 80 and 100% of the data did not vary meaningfully when using *de novo* algorithms. Second, the AL algorithm out-performed the other methods followed by DGC, AGC, CL, open-reference, and closed-reference, and SL. Third, with the possible exception of the CL algorithm, the MCC values for each of the only demonstrated a small amount of variation between runs of the algorithm with a different ordering of the input sequences. This indicates that although there may be variation between executions of the same algorithm, they produce OTU assignments that are equally good. Revisiting the concept of stability, we question the value of obtaining stable OTUs when the the full dataset is not optimally assigned to OTUs. Our analysis indicates that the best algorithm for assigning the Canadian soils sequences to OTUs using a 97% threshold is the AL algorithm.

Deep sampling of 16S rRNA genes. Three factors make the Canadian soil dataset less than desirable to evaluate clustering algorithms. First, the Canadian soil dataset that the He study used to analyze the stability of OTUs is one of the earliest 16S rRNA gene sequence datasets published using the 454 FLX platform. Developments in sequencing technology now permits the sequencing of millions of sequences for a study. In addition, because the original phred quality scores and flowgram data are not available, it was not possible to adequately remove sequencing errors (REFS). The large number of sequences that one would expect to remain in the dataset are likely to negatively affect the performance of all of the clustering algorithms. Second, the dataset used in the He study covered the V9 region of the 16S rRNA gene. For a variety of reasons, this region is not well represented in databases, including the reference database used by the closed and open-reference algorithms. Of the 99,322 sequences in the default QIIME database, only 99,310 fully cover the V9 region. In contrast, 99,310 of the sequences fully covered the V4 region. Inadequate coverage of the V9 region would adversely affect the ability of the reference-based methods to assign sequences to OTUs. Third, our previous analysis has shown that the V9 region evolves at a rate much slower than the rest of the gene. With these points in mind, we compared the clustering assignment for each of these algorithms using a time series experiment that was obtained using mouse stool (REFS). The MiSeq platform was used to generate 2,825,001 sequences from the V4 region of the 16S rRNA gene of 360 samples. Parallel sequencing of a mock community indicated that the sequencing error rate was approximately 0.02% (REFS). Although no dataset is perfect for exhaustively testing these clustering algorithms, this dataset was useful for demonstrating several points.

[Something, something, something.]

- Compare MCC and its variation when using different algorithms
- Address stability
- Quality of algorithms are data-dependent

Evalution of an open-source alternative to USEARCH. The AGC and DGC algorithms appear to perform as well and possibly than the hierarchical clustering algorithms for some datasets. These algorithms utilize the USEARCH program (REF), which is not available as open source code and is only available for free to academic users as a 32-bit program. Access for non-academic users and those needing the 64-bit version is available commercially from the developer. An alternative to USEARCH is VSEARCH, which is being developed in parallel to USEARCH as a open-source alternative. One subtle difference between the two programs is that USEARCH employs a heuristic to determine when to stop searching a database because it think it has found the best hit or does not believe that the database contains a hit. This means that the best possible match is not necessarily returned once a highly similar hit is obtained. In contrast, VSEARCH does not utilize the heuristic, which the VSEARCH developers claim enhances the sensitivity relative to USEARCH. Using the two datasets, we determined whether the AGC and DGC algorithms as implemented by the two methods yielded sequencing assignments of similar quality.

- Able to parallelize

[Something, something, something.]

- MCC for VAGC/VDGC > AGC/DGC
- MCC for VAGC and VDGC are basically the same
- Stability of VAGC/VDGC goes up

This also explains the variation observed in the AGC and DGC algorithms, which rely upon the USEARCH algorithm.

Problems with reference-based clustering in general and as implemented in QIIME. The He study indicated that the closed-reference algorithm generated perfectly stable OTUs. This was unsurprising since, by definition, the algorithm represents a one-to-one mapping of reads to a reference and treats the input sequences independently. An important test that was not performed in the He study was to determine whether the clustering was sensitive to the order of the sequences in the database. The default QIIME database, which was used in the He study and by others, contains full-length sequences that are at most 97% similar to each other. We randomized the order of the reference sequences 30 times and used them to carry out the closed-reference algorithm with the full MiSeq dataset, which contained 32,107 unique sequences. Surprisingly, we observed that the number of OTUs generated was not the same in each of the randomizations. On average there were 28,058.97 sequences that mapped to a reference OTU per randomization (range from 28,007 to 28,111). The original ordering of the reference resulted in 27,876 sequences being mapped, less than the minimum observed number of mapped sequences when the references were randomized. This surprising result was likely due to the performance of the USEARCH heuristic. As with the AGC and DGC algorithms, if the order of the reference sequences differed between runs, then the heuristic criteria would be triggered at different points. To test this further, we substituted VSEARCH for USEARCH in the closed-reference algorithm. When we used VSEARCH the original ordering of the reference sequences and all randomizations were able to map 28,440 sequences to reference

OTUs. That VSEARCH mapped more sequences than even the maximum number of sequences found across the USEARCH randomizations supports the VSEARCH developers' assertion that it has greater sensitivity than USEARCH.

We also observed that regardless of whether we used USEARCH or VSEARCH, the reference OTU labels that were assigned to each OTU differed between randomizations. When we used USEARCH to perform closed-reference clustering, an average of 57.4% of the labels were shared between pairs of the 30 randomizations (range=56.1 to 59.5). If we instead used VSEARCH an average of 55.3% of the labels were shared between pairs of the 30 randomizations (range=52.5 to 58.4). To better understand this result, we further analyzed the reference database that is used in QIIME. We hypothesized that within a given region there would be sequences that were more than 97% similar and possibly identical to each other. When a sequence was used to search the randomized databases, it would encounter a different reference sequence as the first match with each randomization. Among those reference sequences that fully overlap the V4 region, there were 7,785 pairs of sequences that were more than 97% similar to each other over the full length of the 16S rRNA gene. When the extracted V4 sequences were dereplicated, we identified 88,743 unique sequences. Among these dereplicated V4 sequences there were 317,176 pairs of sequences that were more than 97% similar to each other. The presence of duplicate V4 reference sequences explains the lack of labeling stability when using either USEARCH or VSEARCH to carry out the closed-reference algorithm. We suspect that the reference database was designed to only include sequences that were at most 97% similar to each other way to overcome the limitations of the USEARCH search heuristic.

Beyond comparing the abundance of specific OTUs across samples, the reference database is used in the open and closed-reference algorithms to generate OTU labels that are used in several downstream applications. to extract information from a reference phylogenetic tree to carrying out UniFrac-based analyses (REFS) and to identify reference genomes for performing analyses such as PICRUSt (REFS). Because these downstream applications depend on the correct and unique labeling of the OTUs, the lack of stability of the labeling is problematic. As one illustration of the effects that incorrect labels would have on an analysis, we asked whether the duplicate sequences had the same taxonomies. Among the 3,060 reference sequences that had one duplicate, 425 had discordant taxonomies. Furthermore, among those 1,628 sequences with two or more duplicates, 655 had discordant taxonomies. Two sequences mapped to 30 and 10 duplicate sequences and both contained 7 different taxonomies. There was also a sequence had 129 duplicates and contained 5 different taxonomies. Together, these results demonstrate some of the considerable problems with the reference-based clustering of sequences.

Conclusions

- Stability is meaningless in context of accuracy and precision of clustering assignments
- *De novo* clustering algorithms outperform reference-based approaches.
- The best algorithm is data-specific and researchers should evaluate the clustering obtained by multiple methods
- Don’t use USEARCH to cluster sequences
- VSEARCH is an excellent alternative
- Reference-based clustering has many problems
- Many difficulties in clustering have to do with underlying data quality.

Methods

454 FLX-generated Roesch Canadian soil dataset After obtaining the 16S rRNA gene fragments from GenBank (accessions EF308591-EF361836), we followed the methods outlined by the He study by removing any sequence that contained an ambiguous base, was identified as being a chimera, and fell outside a defined sequence length. Although they reported observing a total of 50,542 sequences that were represented by 13,293 unique sequences, we obtained a total of 50,946 sequences that were represented by 13,393 unique sequences. Similar to the He study, we randomly sampled, without replacement, 20, 40, 60, and 80% of the sequences from the full data set. The random sampling was repeated 30 times. The order of the sequences in the full dataset was randomly permuted without replacement to generate an additional 30 datasets. For the hierarchical clustering algorithms and to generate a distance matrix, the pairwise distances between sequences were calculated in mothur using the pairwise.seqs command with the default Needleman-Wunsch alignment algorithm and parameters. Execution of the clustering algorithms was performed as described in the original He study using mothur (v.1.37) and QIIME (v.1.9.1) as appropriate.

MiSeq-generated Murine gut microbiota dataset

Analysis of reference database * used gg_13_8 with the 97_otus.fasta as the reference

Calculation of Matthew's Correlation Coefficient

-> Available within mothur as the sens.spec command

Software availability QIIME v.1.9.1 mothur v.1.37.0 R v.3.2 Reproducible workflow including this manuscript as a literate programming document is available at https://github.com/SchlossLab/Schloss_Cluster_PeerJ_2015