Inferring microbial co-occurrence networks from amplicon data: a systematic evaluation

Dileep Kishore[a,b], Gabriel Birzu[c,f], Zhenjun Hu[a], Charles DeLisi[a,c], Kirill S. Korolev[a,b,c]#, Daniel Segrè[a,b,d,e]#

[a]Bioinformatics Program, Boston University, Boston, Massachusetts, USA
[b]Biological Design Center, Boston University, Boston, Massachusetts, USA
[c]Department of Physics, Boston University, Boston, Massachusetts, USA
[d]Department of Biomedical Engineering, Boston University, Boston, Massachusetts, USA
[e]Department of Biology, Boston University, Boston, Massachusetts, USA
[f]Department of Applied Physics, Stanford University, Stanford, California, USA
#Correspondence should be sent to korolev@bu.edu or dsegre@bu.edu

Abstract

Microbes commonly organize into communities consisting of hundreds of species involved in complex interactions with each other. 16S ribosomal RNA (16S rRNA) amplicon profiling provides snapshots that reveal the phylogenies and abundance profiles of these microbial communities. These snapshots, when collected from multiple samples, can reveal the co-occurrence of microbes, providing a glimpse into the network of associations in these communities. However, the inference of networks from 16S data involves numerous steps, each requiring specific tools and parameter choices. Moreover, the extent to which these steps affect the final network is still unclear. In this study, we perform a meticulous analysis of each step of a pipeline that can convert 16S sequencing data into a network of microbial associations. Through this process, we map how different choices of algorithms and parameters affect the co-occurrence network and identify the steps that contribute substantially to the variance. We further determine the tools and parameters that generate robust co-occurrence networks and develop consensus network algorithms based on benchmarks with mock and synthetic datasets. The Microbial Co-occurrence Network Explorer or MiCoNE (available at https://github.com/segrelab/MiCoNE), follows these default tools and parameters and can help explore the outcome of these combinations of choices on the inferred networks. We envisage that this pipeline could be used for integrating multiple datasets, and for generating comparative analyses and consensus networks that can guide our understanding of microbial community assembly in different biomes.

***Keywords*—** Microbiome, 16S rRNA, Interaction, Denoising, Taxonomy, Network Inference, Correlations, QIIME2, Co-occurrence, Networks, Consensus algorithm, Pipeline, nextflow

Importance

Mapping the interrelationships between different species in a microbial community is important for understanding and controlling their structure and function. The surge in the high-throughput sequencing of microbial communities has led to the creation of thousands of datasets containing information about microbial abundances. These abundances can be transformed into co-occurrence networks, providing a glimpse into the associations within microbiomes. However, processing these datasets to obtain co-occurrence information relies on several complex steps, each of which involves numerous choices of tools and corresponding parameters. These multiple options pose questions about the robustness and uniqueness of the inferred networks. In this study, we address this workflow and provide a systematic analysis of how these choices of tools affect the final network, and guidelines on appropriate tool selection for a particular dataset. We also develop a consensus network algorithm that helps generate more robust co-occurrence networks based on benchmark synthetic datasets.

Introduction

Microbial communities are ubiquitous and play an important role in marine and terrestrial environments, urban ecosystems, and human health [1–7]. These microbial communities, or microbiomes, often comprise several hundreds of different microbial strains interacting with each other and their environment, often through complex metabolic and signaling relationships [8–11]. Understanding how these interconnections shape community structure and function is a fundamental challenge in microbial ecology, and has applications in the study of microbial ecosystems across different biomes. With the advancement in DNA sequencing technologies [12–14], more information can be extracted from these microbial community samples than ever before. In particular, high-throughput sequencing, including metagenomic sequencing and sequencing of 16S rRNA gene amplicons (hereafter referred to as 16S data) of microbial communities, can help detect, identify and quantify a large portion of the constitutive microorganisms of a microbiome [15–18]. These advances have led to large-scale data collection efforts involving terrestrial [2, 4, 19], marine [1, 3] and human-associated microbiota [7, 20, 21].

This wealth of information has the potential to help us understand how communities assemble and operate. In particular, a powerful tool for translating microbiome composition data into knowledge is the construction of association (co-occurrence) networks, in which microbial taxa are represented by nodes, and frequent co-occurrences (or negative co-occurrences) across datasets are encoded as edges between nodes. While the relationship between directly measured interactions [22–24] and statistically inferred co-occurrence is still poorly understood [25, 26], a significant amount of effort has gone into estimating co-occurrences from large microbiome sequence datasets [27–30].

The importance of these networks is two-fold: first, they can serve as maps that help identify hubs of keystone species [26, 31], and the community response to environmental perturbations or

underlying host conditions [32]; second, they can serve as a bridge towards building mechanistic models of microbial communities, greatly enhancing our capacity to understand and control them. For example, multiple studies have shown the importance of specific microbial associations in the healthy microbiome [7, 21, 33] and their role in dysbiosis [32, 34, 35]. In the context of terrestrial biogeochemistry, co-occurrence networks were shown to help understand microbiome assembly [36], and the response of microbial communities to environmental perturbations [37].

One of the most frequently used avenues for inferring co-occurrence networks is the parsing and analysis of 16S sequencing data [26, 38]. Numerous software tools and pipelines have been developed to analyze 16S sequencing data, with a strong emphasis on the known limitations of this method, including resolution, sequencing depth, compositional nature, sequencing errors, and copy number variations [39, 40]. Popular methods for different phases of the analysis of 16S data include tools for: (i) quality checking and trimming the sequencing reads; (ii) denoising and clustering the trimmed reads [41–43]; (iii) assigning taxonomy to the denoised reads [44]; (iv) processing and transforming the taxonomy count matrices [45]; and (v) inferring the co-occurrence network [46–48]. Different specific algorithms are often aggregated into popular online platforms (like MG-RAST [49], Qiita [50]) and software packages (such as Quantitative Insights Into Microbial Ecology 2 (QIIME2) [51]). The different methods and tools can lead to vastly different inferences of community compositions and co-occurrence networks [52, 53], making it difficult to reliably compare networks across different publications and studies. This difference is partially due to the focus of existing platforms on Operational Taxanomic Unit (OTU) or Exact Sequence Variant (ESV) generation and not on the effects of upstream statistical methods on the inferred co-occurrence networks. Furthermore, no organized framework currently exists that can systematically analyze and compare each step in the pipeline for processing amplicons into co-occurrence networks.

In this study, we present a standardized 16S data analysis pipeline called Microbial Co-occurrence

Network Explorer (MiCoNE) that produces robust and reproducible co-occurrence networks from 16S sequence data of microbial communities, and enable users to interactively explore how the network would change upon using different alternative tools and parameters at each step. Our pipeline is coupled to an online integrative tool for the organization, visualization, and analysis of inter-microbial networks called Microbial Interaction Network Database (MIND) [54], which is available at http://microbialnet.org/. Through a systematic comparative analysis, we determine which steps of the MiCoNE pipeline have the largest influence on the final network, and which choice seems to have the optimal agreement with the tested mock and synthetic datasets. These steps together with our default settings ensure better reproducibility and easier comparison of co-occurrence networks across datasets. We expect that our tool will also be useful for benchmarking future alternative methods, and for ensuring a transparent evaluation of the possible biases introduced by the use of specific tools.

Results

Microbial Co-occurrence Network Explorer (MiCoNE)

We developed MiCoNE, a flexible and modular pipeline for the inference of co-occurrence networks from 16S data. MiCoNE incorporates various popular, publicly available tools as well as custom Python modules for 16S data analysis and network inference (Methods). The different steps that are a part of the MiCoNE co-occurrence network inference workflow (Figure 1) can be grouped into five major modules; (i) Sequence Processing (SP); (ii) Denoising and Clustering (DC); (iii) Taxonomy Assignment (TA); (iv) OTU Processing (OP); and (v) Network Inference (NI). Each process in the pipeline is implemented through multiple tools (see Methods and Figure 1). The effects of changing any intermediate step of the pipeline can be evaluated in terms of the final

network outcome, as well as on any of the intermediate metrics and data outputs. The choice of tools and parameters is encoded in a configuration file (with parameters as shown in Tables S2-S6 at https://github.com/segrelab/MiCoNE-pipeline-paper). Through a systematic analysis of tool combinations at each step of the pipeline, we estimated how much the final co-occurrence network depends on the possible choices at each step.

Our analysis involved two types of data: The first type consisted of 16S sequencing data from samples of human stool microbiomes from a fecal microbiome transplant (FMT) study of autism [55]. The second type was a collection of datasets synthetically or artificially created for the specific goal of evaluating computational analysis tools. In particular, in order to benchmark each step in MiCoNE, we used both mock data (labeled mock4, mock12, and mock16) from mockrobiota [56] and synthetic networks generated using the NorTA [47] and seqtime [26] approaches (See Methods).

DC: Denoising and clustering methods differ in their identification of sequences that are low in abundance

The DC step is commonly carried out to generate representative sequences (in the form of the OTU/ESV tables) from the demultiplexed and trimmed 16S sequencing data. In order to compare the count tables generated by different tools, we processed the 16S sequencing reads (from the FMT study [55]) using 5 different methods: open-reference clustering, closed-reference clustering, de novo clustering, Divisive Amplicon Denoising Algorithm 2 (DADA2) [42] and Deblur [43]. The first three methods are from the vsearch plugin from QIIME2 [51]. The closed and open reference methods in this analysis use the Greengenes [57] database for reference sequence alignment.

A comparison of the different methods was carried out by calculating the mean UniFrac distances across all samples (Figure 2). The analysis was performed using both the weighted UniFrac [58] (Figure 2A) distance metric, which takes into account the counts of the representative sequences,

and the unweighted UniFrac [59] (Figure 2B) distance metric, which gives equal weights to each sequence.

The first main message emerging from this analysis is that the representative sequences generated by the different methods, with the exception of Deblur, are similar to each other when weighted by their abundance (Figure 2A). A second message is that the different methods differ mainly in the assignment of sequences of lower abundance. This can be inferred from the unweighted comparison (Figure 2B) which shows an increase in dissimilarity between each pair of methods (see additional details in Supplementary and Figure S2).

These comparisons only elucidate the similarity between a pair of methods. To determine which tool most accurately recapitulates the reference sequences in the samples, we applied the same pipeline step to process the mock datasets (mock4, mock12, and mock16) and compared the predicted representative sequences with the true sequences and their distribution. The results (Figure 2C and 2D) show that the predicted sequence distributions are overall different from the expected ones. The variation across datasets indicates that the datasets themselves play a big role in method performance. We note that there is no method that outperforms the rest in all datasets (see Supplementary for an extended discussion). Based on being among the top performers on the mock datasets, their de novo error correcting nature and previous independent evaluation [60], DADA2 and Deblur appear to be the most reliable. This is because the open-reference and de novo clustering methods return a much larger number of OTUs compared to the other pipelines and would affect the accuracy of the network inference step if stringent filtering is not performed. Overall, since DADA2 as compared to Deblur, displays better performance on all the mock datasets on the weighted UniFrac metric, we set this tool as the default for the DC step of the pipeline. However, if comparison across studies that have sequenced different 16S regions is required, closed-reference and open-reference might be a better option.

After the denoising, the sequences are subject to Chimera Checking (CC). The MiCoNE pipeline supports two different chimera checking methods, "uchime-denovo" [51], and "remove bimera" [42]. We did not notice any notable difference between the two methods (Figure S3), implying that they identify and remove mostly the same set of sequences as chimeras. Since the remove bimera method was originally developed in conjunction with dada2 we use this method as the default. The DC step thus results in a reduced set of unique sequences, which will be referred to as representative sequences in the subsequent steps.

TA: Taxonomy databases vary widely in taxonomy assignments beyond Order level

Taxonomy databases are used to assign taxonomic identities to the representative sequences obtained after the DC step. The three 16S taxonomic reference databases used in this study are SILVA [61], Greengenes (GG) [57] and National Center for Biotechnology Information (NCBI) RefSeq [62] (Methods). These databases vary substantially in terms of taxonomy hierarchies, including species names and phylogenetic relationships [63]. Assignment using a particular database also requires a query tool. We used the "Naive Bayes" classifier from QIIME2 for the GG and SILVA databases and the "BLAST" tool (included as a QIIME2 plugin) for the NCBI database. These tools have been well quantified and optimized [44], hence, we made use of the default parameters in our analyses.

The representative sequences obtained using the default settings of the DC step were used for taxonomic assignment using the three reference databases. Figure 3A depicts a flow diagram that shows how the top 50 representative sequences (sorted by abundance) are assigned a Genus according to the three databases. The different databases lead to assignments that qualitatively display similar distributions. However, the assigned Genus compositions also display clear differences, as does the percentage of unassigned representative sequences (pink). Some of the differences in

Genus composition have a clear explanation, for example, abundant Genera like Bacteroides and Escherichia are assigned to different representative sequences. The large percentage of unassigned sequences is due to the large fraction of the representative sequences assigned to an "unknown" Genus during the assignment process (Methods).

After the assignment, we performed a pairwise comparison of the similarity between the top 100 assignments (by abundance) from different databases at every taxonomic level (Figure 3B). The comparisons of the assignments below the Order level (Family, Genus, and Species) show less than 45% similarity between any pair of databases. This implies that the taxonomy assignments from each reference database are fairly unique. The comparison of all assigned genera (Figure S4), instead of just the top 100, contains a higher percentage of mismatches. This suggests that, comparatively, the most abundant sequences are more consistently matched to the same taxonomies, at least for the dataset tested in the current analysis.

To obtain an absolute measure of the accuracy of the taxonomic assignments, we used the representative sequences from the DC step for mock datasets as the query sequences and the expected taxonomic composition as the standard to compare against. We used the Bray-Curtis distance metric [64] to calculate the distance between the predicted and expected taxonomic distribution (Figure 3C). We find that none of the databases perform better than the others in absolute terms and that the dissimilarity with the expected composition is high (> 0.5 for Family and Genus and > 0.9 for Species), indicating that all the databases have some limitations when trying to recapture the expected taxonomic composition.

Since no database performs better than others against mock datasets, the choice of which database to use could be driven by other reasons (see Supplementary discussion). One reason to choose a particular database could be the frequency of updates and the potential for future growth. Both GG, due to its frequent use in the literature [63], and NCBI, due to its regular revision and maintenance,

could be good choices for taxonomy assignment. In our default pipeline, we choose GG as the default method.

The TA step results in a taxonomic counts table that is used as input to the subsequent steps of the pipeline. Note that the count tables at different levels can be obtained through aggregation; for example, Genus count tables were obtained by summing up the counts of the lower taxonomy levels (Species and OTU) that map to the same higher taxonomy level entity.

NI: Different network inference methods drastically affect edge-density and connectivity

The ten network inference methods we used in this step fall into two groups: the first set of methods (Pearson, Spearman, SparCC [38, 46], and propr [65]) infer pairwise correlations while the second set (SpiecEasi [47], FlashWeave [48], COZINE [66], HARMONIES [67], SPRING [68], and mLDM [69]) infer direct associations. In general, when we refer to co-occurrences we include the associations from both correlation based methods and direct association based methods. On the other hand, when mentioning correlations, we will exclusively refer to edges inferred by the correlation methods. Note that while Pearson and Spearman methods are included in the pipeline for completeness, they tend to generate a large number of spurious edges as they are not intended for compositional datasets. Thus, they are not included in subsequent quantitative analyses.

Filtered (see OP step in Methods) genus-level counts table obtained using the default settings in the previous steps were used as input for the different network inference algorithms (Figure 4). Even from a visual inspection (Figure 4A), one can see that the different networks differ vastly in their edge-density and connectivity, with common edges often displaying inverted signs.

To quantify the differences between the networks, we analyzed the distribution of common nodes and edges (Figure 4 B and 4C) using UpSet plots [70]. The node intersection analysis shows that the

networks have 33 out of 68 total unique nodes in common and that no network possesses a unique node. Edge intersections in contrast show that only 8 edges (out of 202 total unique edges) are in common between all the methods and each network has many unique edges. These results showed a substantial rewiring of connections in different inferred networks and prompted us to identify associations robust across methods, through consensus algorithms.

NI: The scaled-sum consensus method shows high precision on benchmark datasets

Inspired by previous approaches [71, 72], we developed two methods that take into consideration the evidence offered by each network inference algorithm and generate a consensus network that contains the common edges among the inferred networks.

Both of our approaches - simple voting (SV) and scaled-sum (SS) - combine appropriately filtered networks inferred from correlation-based and direct association methods (see Methods). We chose the scaled-sum method as the pipeline default since this method takes into account the weights of the associations in the determination of the final consensus. The pipeline enables the selection of any subset of methods for the consensus calculation. Currently, by default, all direct association methods (SpiecEasi, COZINE, HARMONIES, SPRING, mLDM, and FlashWeave) are used, together with SparCC and propr.

Similar to what was done for the previous steps of the pipeline, and in analogy with previous estimations of network inference accuracy [47, 53], we evaluated the network inference algorithms and the final consensus network using synthetic interaction data. For this purpose, we generated synthetic interaction data using the "NorTA" [47] and "seqtime" [73] methods (see Methods). For each method, an OTU counts table was generated based on the selected parameters and abundance distributions. This counts table was used as the input to the MiCoNE pipeline to generate predicted

associations. The interaction network used to generate the counts table was used as the source of true interactions to calculate the precision and sensitivity (Figure 5) of the consensus algorithms. These values are also compared with the precision and sensitivity of four individual network inference methods (two correlation-based, i.e. propr, FlashWeave, and two direct association methods, i.e. SparCC, and SpiecEasi), which were selected based on their high precision (see Figure S5 and Figure S6). As shown in Figure 5 the consensus algorithm, especially the scaled-sum method, captures true associations with high precision (through the removal of edges that are either not present in most of the inference methods or whose association strength is low across methods). Although this increase in precision is associated with a drop in sensitivity (as the consensus parameter θ increases), the consensus networks provide valuable and practically helpful results, in the form of a short list of high-confidence associations. Overall, the scaled-sum method for $\theta = 1.000$ performs the best (precision = 1.000 for both NorTA and seqtime). Figure S5 and Figure S6 show the precision and sensitivity values of all network inference and consensus algorithms for each interaction network in the synthetic datasets. The scaled-sum method for $\theta = 0.333$ (default option in the pipeline) shows a high precision (0.956 with NorTA; 0.688 with seqtime), without displaying a significant reduction in sensitivity (Figure 5, Figure S5 and Figure S6). However, if higher precision is required $\theta > 0.5$ can be considered.

Impact of different pipeline steps on co-occurrence networks

In order to analyze the effect of different processing methods on the inferred co-occurrence networks (before consensus estimation), we generated networks using all possible combinations of methods and quantified the variability due to each choice (Figure 6A). This was achieved by building a linear model of the edges of the network as a function of the various steps in the pipeline workflow (see Methods). Figure 6A, shows the percentage of total variation among the co-occurrence networks due

to the different steps of the pipeline. The TA step, or more specifically the choice of 16S reference database, contributes the most (65.4%) to the variation in the networks, followed by the OP step (26.8%). This result highlights the importance of the taxonomy assignment step in the 16S data analysis workflow, implying that a change in the reference database will result in drastically different inferred networks. This is likely due to the differential assignment of representative sequences to taxonomic entities (Figure 3 and Figure S4), which drastically alter the nodes and hence the underlying network topology.

The effects of the different steps of the pipeline on the inferred networks can be visualized through dimensionality reduction. The PCA in Figure 6B shows all the above networks, colored by the tools used in the DC, TA, OP, and NI steps in each subfigure. The major effect of the TA step choice, shown before in Figure 6A, is also reflected in the PCA plot, where networks segregate based on the database used (Figure 6B and Figure S1). Additionally, the plot also shows that the variation between the networks decreases when the low abundance OTUs are removed from the network. It is also evident that in the NI step, some networks, especially those inferred using the direct association network inference methods, are much closer in the PCA plot regardless of the reference database used. The network variance analysis performed on the supplementary dataset [74] (stool samples from radiation-exposed bank vole) shown in Figure S8 support these observations, implying that these findings are fairly consistent across different datasets. These results suggest that the most important criterion for accurate comparative analysis of co-occurrence networks is the taxonomy reference database followed by the level of filtering of the taxonomy tables and the network inference algorithm used.

The default pipeline

The systematic analyses in the previous sections illustrate that the choice of tools and parameters can have a big impact on the final consensus co-occurrence network. However, the mock communities and synthetic data provide an opportunity to select combinations of tools that yield the most accurate and robust results. As highlighted in the above sections for individual steps, we propose a set of tools and parameters as the defaults for the pipeline (Table 1).

Figure 7 shows the co-occurrence networks inferred for the healthy subjects (control) and subjects with autism specific disorder (ASD) in the fecal microbiome transplant study [55] (constructed using the default tools and parameters from Table 1). This figure demonstrates a typical use case of comparative analysis of networks using the MiCoNE pipeline. As a consequence of using the consensus network algorithm, the final co-occurrence networks are sparse and can be visually compared and examined.

The analysis of the rewiring of associations in the ASD samples with respect to the control provides a guide for the identification of key genera that could be linked to dysbiosis. We observed 22 unique links in the network for control samples, 12 unique links in the network for ASD subjects, and 7 edges in common between the two networks. Although these unique associations do not imply actual interactions, they can still serve as potential starting points for literature surveys and further experimental exploration of mechanistic processes underlying dysbiosis. For example, *Prevotella* and *Porphyromonas*, genera previously implicated in ASD [55, 75] and cognitive impairment [76] display modified connectivity in our network, suggesting that the observed associations may be relevant for understanding the role of these bacteria in disease. Additional visualization and comparison of networks can be performed using the Microbial Interaction Network Database (MIND) [54].

Figure S7 shows a sensitivity analysis in which we compared the default network against networks

generated by altering one of the steps of the pipeline relative to the default. This result, both visually (Figure S7 A), and quantitatively (Figure S7 B) suggests that the most significant changes occur when the OP or TA steps are changed from the default value.

Discussion

Why MiCoNE?

A myriad of tools and methods have been developed for different parts of the workflow for inference of co-occurrence networks from 16S rRNA data. Our analyses have shown that networks generated using different combinations of tools and approaches can be substantially different from each other, highlighting the need for a clear evaluation of the source of variability and for tools that provide the most robust and accurate results. Our newly developed software, MiCoNE, is a customizable pipeline for the inference of co-occurrence networks from 16S rRNA data that enables users to compare networks generated by multiple possible combinations of tools and parameters. Importantly, in addition to revisiting the test cases presented in this work, users will be able to explore the effect of various tool combinations on their own datasets of interest. The MiCoNE pipeline has been built in a modular fashion; its plug-and-play architecture enables users to add new tools and steps, either using existing packages that have not been examined in the present work or those developed in the future. The MiCoNE Python package provides functions and methods to perform a detailed analysis of the count matrices and the co-occurrence networks. The inferred networks are exported to a custom JSON format (see Supplementary) by default, but can also be exported to Cytoscape [77], GML [78], and many other popular formats via the Python package.

While several tools/workflows such as QIIME2 [51] and NetCoMi [79] can be used to generate co-occurrence networks from 16S sequencing data, no single tool exist that integrates the complete

process of inferring microbial interaction networks from 16S sequencing reads. MiCoNE is unique as it offers this functionality packaged in a workflow that can be run locally, on the compute cluster, or in the cloud.

The default pipeline and recommended tools

Through MiCoNE, in addition to transparently revealing the dependence of co-occurrence networks on tool and parameter choices (see Discussion in Supplementary Text for details on the DC, TA, and OP steps), we have taken advantage of our spectrum of computational options and the availability of mock and synthetic datasets, to suggest a default standard setting that streamlines comparisons across datasets. Additionally, we have developed a consensus approach that can reliably generate fairly robust networks across multiple tool choices. Even if, in the current analysis, we have shown the relevance of our approach to two very different types of microbiome datasets (human and vole stool), it is important to remember that there is no universal standard for microbial interaction data, and our conclusions are based on the specific datasets used in our analysis. While our analysis is based on several mock and synthetic datasets that cover a diverse range of abundance distributions and network topologies, datasets with drastically different distributions may require a re-assessment of the best settings. However, the MiCoNE pipeline provides a platform for easy evaluation of accuracy, variance, and other properties at each workflow step for any other dataset of interest.

The networks generated by different network inference methods show considerable differences in edge-density and connectivity, partially due to the underlying assumptions regarding sparsity, distribution, and compositionality. To address this issue, we have developed two consensus algorithms (simple voting and scaled-sum method) that generate networks whose links have evidence based on multiple inference algorithms.

We find that the scaled-sum method performs the best on synthetic datasets, and is therefore

chosen as the default for the NI step of the pipeline. Notably, the consensus network displays a higher precision and returns a concise list of robust associations representing a valuable set for experimental validation follow-up.

Future directions

Future work building upon our current results could enhance the network inference process in multiple ways. The current analyses make use of one fecal microbiome transplant dataset with healthy and ASD samples, three mock community datasets, and several datasets generated by two synthetic interaction methods. Incorporating datasets from a broad spectrum of biomes with varying microbial distributions into MiCoNE will likely increase the robustness and generalizability of the results from these analyses.

The network analyses in this study are primarily at the Genus level, wherein the lowest resolution of a node is a Genus and if an entity cannot be resolved to the Genus level, the next lowest taxonomic level is used (for example, Family). Consequently, two entities belonging to the same lineage where one entity is resolved to the Genus level and another is resolved to the Family level are treated as two different nodes in the network. Thus, developing an overlap metric to compare nodes with shared lineages within and across networks could enable more biologically and phylogenetically relevant comparisons.

In thinking about the possible biological interpretations of the co-occurrence network computed by us and others, it is important to remember that there is no solid basis for assuming that these networks carry information about physical or metabolic interactions [80, 81]. Comparing co-occurrence networks and directly measured interactions remains a major unresolved challenge [82, 83], which needs to be investigated further. Understanding this connection will be benficial for predicting interactions in systems where direct interaction measurements cannot be taken. Further,

benchmarking of co-occurrence networks could also be pursued through the use of literature-based interactions [1] or biological benchmark interaction data [84]. Additionally, MiCoNE could be extended to enable the processing of metagenomics sequencing data, facilitating the analysis of a much larger and diverse range of datasets and domains of life.

Although in the current analysis, we have only used default parameter values recommended by the tool creators, the MiCoNE pipeline could be used in the future to explore any combinations of parameters and to optimize these values for improved network inference. Overall, there likely is no "best method" for the various steps of 16S data analysis, and hence, MiCoNE is intended to help researchers to identify the methods and algorithms that are most suitable for their datasets in an easy-to-use and reproducible manner.

We envision that MiCoNE, and its underlying tools and databases, will be increasingly useful for building large comparative analyses across studies. It enables rapid, configurable, and reproducible inference of microbial networks and furthers the formulation of hypotheses about the role of these interactions on community composition and stability. These comparative analyses will require coupled network analysis and visualization tools (such as MIND [54]) and need systematic access to datasets, shared in accordance with FAIR standards [81].

Materials and Methods

16S rRNA sequencing datasets

This study utilized two types of 16S rRNA sequencing datasets: biological datasets and mock/synthetic datasets. Biological datasets are collections of sequencing reads obtained from naturally occurring microbial community samples. The current analysis used stool samples from a fecal microbiome transplant study of autism [55] as the biological dataset. This dataset was chosen because the

sequences were easily accessible on Qiita [85] and optimally pre-processed according to the Earth Microbiome Project (EMP) [2] protocol, allowing them to be used directly as input to the MiCoNE pipeline. The study was composed of multiple sequencing runs. The runs that contained paired-end reads (run 2 (10M reads), run 3 (750K reads) and run 4 (16M reads)), were downloaded from Qiita [85] (study ID 10532) and used as input sequences for the MiCoNE pipeline. Sequences from both control (212 samples including neurotypical and donors) and autism spectrum disorder (ASD) (126 samples) patients were included in the analyses. All the network analyses in the study, unless explicitly mentioned, were performed on the healthy and ASD samples in the fecal microbiome transplant study. The mock community 16S datasets are experimental sequencing data obtained for artificially assembled collections of DNA of species in known proportions. The mock datasets used for this study, obtained from mockrobiota [56], are labeled mock4, mock12, and mock16. The mock4 community is composed of 21 bacterial strains. Two replicate samples from mock4 contain all species in equal abundances, and two additional replicate samples contain the same species in unequal abundances. The mock12 community is composed of 27 bacterial strains that include closely related taxa with some pairs having only one to two nucleotide differences from one another. The mock16 community is composed of 49 bacteria and 10 Archaea, all represented in equal amounts. In addition to these datasets, we have utilized a dataset containing stool samples from radiation-exposed bank vole [74] (Qiita study ID 13114) for supplementary analyses.

MiCoNE

The flowchart describing the workflow of MiCoNE (Microbial Co-occurrence Network Explorer), our complete 16S data-analysis pipeline, is shown in Figure 1. The pipeline integrates many publicly available tools as well as custom R or Python modules and scripts to extract co-occurrence associations from 16S sequence data. Each of these tools corresponds to a distinct module that

recapitulates the relevant analyses. All such individual modules are available as part of the MiCoNE package. The inputs to the pipeline by default are raw untrimmed 16S rRNA sequence reads, but the software can be alternatively configured to use trimmed sequences, OTU tables and other types of intermediate data (see documentation). The configuration and modular nature of the MiCoNE package enables users to start and end the pipeline at any point in the workflow, and to run parts of the pipeline in isolation. The pipeline supports both paired-end and single-end reads, and additionally supports independently processing reads from multiple runs and merging the OTU tables in the DC step. The final output of the pipeline is the inferred network of co-occurrence relationships among the microbes present in the samples.

The MiCoNE pipeline provides both a Python API together with a command-line interface and only uses a single configuration file (`nextflow.config`) to encode the configuration parameters. The MiCoNE Python API provides several OTU table and network-related functions and methods, enabling detailed comparison of counts tables and inferred networks if desired. Exploring the effects of these combinations of methods on the resultant networks is difficult and inconvenient since different tools differ in their input and output formats and require interconversions between the various formats. The pipeline facilitates this comparative exploration by providing a variety of modules for interconversion between various formats, and by allowing for easy incorporation of new tools as modules. It also contains helper functions that can help in parsing taxonomies and communicate with the NCBI taxonomy database to query taxonomy by name or taxonomic IDs. The configuration file along with the run file (`main.nf`) lists the inputs, output, and the steps to be performed during runtime, along with the parameters to be used (if different from defaults) for the various steps. The default settings of the pipeline are shown in Table 1 (with default parameter values shown in Tables S2-S6 at https://github.com/segrelab/MiCoNE-pipeline-paper). Since the entire pipeline run is stored in the form of a text file (the configuration file), subsequent runs are highly

reproducible and changes can be easily tracked using version control. The pipeline makes use of the nextflow workflow manager [86] under the hood, making it readily usable on the local machine, cluster, or cloud with minimal configuration change. It also allows for automatic parallelization of all possible processes, both within and across samples. The pipeline is designed to be modular: each tool or method is organized into modules that can be easily modified or replaced. This modular architecture simplifies the process of adding new tools (refer to the modules section in the MiCoNE documentation). The main components of the pipeline are detailed in the subsequent sections.

Sequence Processing (SP)

This module deals with processing the raw multiplexed 16S sequence data into demultiplexed, quality-controlled, trimmed sequences. It consists of the demultiplexing and trimming processes. The demultiplexing process deals with separating the multiplexed sequences into individual samples based on barcodes. The trimming process handles the quality control steps such as trimming adapters and low-quality nucleotide stretches from the sequences. The parameters and tools in this process are fixed and are not available for user customization. The various tools used for the processes were adapted from QIIME2 v2021.8.0 [51]. The list of tools used in this step, along with their modules and references are provided in Table 1.

Denoising and Clustering (DC)

This module deals with processing the quality-controlled, trimmed 16S sequence data into OTU or ESV count tables. It consists of the following processes: denoising (or clustering) and chimera checking. The denoise/cluster process handles the conversion of the demultiplexed, trimmed sequences into OTU or ESV count tables (some methods, like closed reference and open reference clustering, make use of a taxonomy reference database for clustering). The chimera checking process handles the removal of chimeric sequences created during the Polymerase Chain Reaction (PCR)

step. The output of this module is a matrix of counts, that describes the number of reads of a particular OTU or ESV (rows of the matrix) present in each sample (columns of the matrix). The options currently available in the pipeline for denoising and clustering are: open reference clustering, closed reference clustering and de novo clustering methods from the vsearch plugin of QIIME2 v2021.8.0 [51] and denoising methods from DADA2 v1.14 [42] (from the DADA2 R package) and Deblur v1.1.0 [43] (from the deblur plugin of QIIME2). The quality filtering and chimera checking tools are derived from those used in QIIME2 v2021.8.0 (uchime-denovo method) and DADA2 (remove bimera method). The list of tools used in this step, along with their modules and references are provided in Table 1.

For the UniFrac analysis in Figure 2, we had set a count threshold of 10, such that if the count of the representative sequences in a particular sample is less than the threshold, it is omitted from the analysis. Additionally, for Figure 2C and 2D, the expected sequences from the mock communities were trimmed to the V4 region before being subject to UniFrac analyses.

Taxonomy Assignment (TA)

This module deals with assigning taxonomies to the representative sequences (OTUs or ESVs). In order to assign taxonomies to a particular sequence, a taxonomy database and a query tool are necessary. The taxonomy database contains a collection of 16S sequences of microorganisms and the query tool allows one to compare a sequence of interest to all the sequences in the database to identify the best matches. Finally, a consensus method is used to identify the most probable match from the list of best matches. The pipeline incorporates GG 13_8 [57] (99% identity), SILVA 138 [61] (99% identity) and the NCBI (16S RefSeq as of Oct 2021) [62] databases for taxonomy assignment. SILVA and GG are two popular 16S databases used for taxonomy identification and the NCBI RefSeq nucleotide database contains 16S rRNA sequences as a part of two BioProjects -

33175 and 33317. The three databases vastly differ in terms of their last update status - GG was last updated on May 2013, SILVA was last updated on August 2020 at the time of writing and NCBI is updated regularly as new sequences are curated. These databases were downloaded and built using the RESCRIPt QIIME2 plugin [87]. The Naive Bayes classifier and the NCBI blast used as the query tools in this study were from the QIIME2 package and the parameters used were the defaults of the package. The consensus algorithm used is the default method used by the classifiers in QIIME2. During the assignment, a representative sequence might be assigned an "unknown" Genus for one of two reasons: the first is if the taxonomy identifier associated with the sequence in the database did not contain a given Genus; the second, more likely reason, is that the database contains multiple sequences that are very similar to the query (representative) sequence and the consensus algorithm (from QIIME2) is unable to assign one particular Genus at the required confidence. The assignments in SILVA were originally substantially different from the other two databases (40% mismatch) even at the Phylum level. However, this was corrected via minor adjustments to the taxonomic names, such as changing Bacteroidota to Bacteroidetes in the SILVA Phylum assignments. The full list of changes can be found in `figure4ab_data.py` in the data and scripts repository. The list of tools used in this step, along with their modules and references are provided in Table 1.

OTU and ESV Processing (OP)

This module deals with normalization, filtering, forking, grouping, and applying transformations to the OTU or ESV counts matrix. Normalization of the count matrix involves converting the count matrix of read counts into a count matrix containing relative abundances. The module also supports rarefaction, which is a normalization technique used to overcome the bias that might arise due to variable sampling depth in different samples. This is performed either by sub-sampling of the matrix to a specified rarefaction depth [45] in order to obtain samples with equal library

sizes. However, due to the potential biases and false positives [88, 89] that might arise during the process, the rarefaction module is disabled by default and can be enabled in the configuration if needed. Hence, although the pipeline supports rarefaction, it is turned off by default. In addition to rarefaction, the MiCoNE pipeline also supports total sum scaling (TSS) and the centered log-ratio (clr) transformation (from the speiceasi R package). However, since most of the network inference methods perform normalization and other transformation operations on the counts matrix as a part of their workflow, the analyses reported in the paper do not explicitly normalize the counts matrices. Filtering, is performed to remove samples or features (OTUs or ESVs) from the counts matrix that are sparse. By default, when the OP module is "on", the samples are filtered out if the total reads in a sample are less than 500 and features are filtered out if the relative abundance is less than 1%, prevalence (percentage of samples containing feature) is less than 5% and count sum across all the samples is less than 100. When the OP module is "off", the filtering is still performed but threshold parameters are much more relaxed. The parameters used are given in Table 1. The forking operation splits the count matrix into multiple matrices based on sample metadata column, this is useful for example to compare case vs. control. The group operation transforms the OTU or ESV count matrix into a taxonomic count matrix at the requested level by adding up counts that map to the same taxonomy and is carried out at the end of the OP step. Finally, transformations are performed in order to correct for and overcome the compositional bias that is inherent in the counts matrix (in the analysis performed in the study these were disabled and directly handled by the network inference algorithm). All the modules in this step were implemented using functions from the biom-format Python package [90].

Network Inference (NI)

This module deals with the inference of co-occurrence associations from the processed OTU or ESV counts matrix. The input count matrices are collapsed to the Genus level (or any other required taxonomy level) using the group module at the OP step. These collapsed matrices are used as input to the network inference methods to produce association matrices at the appropriate taxonomy level. These associations can be represented as a network, with nodes representing the taxonomies of the microorganisms and edges representing the associations between them.

The pipeline includes 4 methods for pairwise correlation metrics, and 6 methods for direct association metrics (refer to Table 1). Pairwise correlation methods involve the calculation of the correlation coefficient between each pair of nodes (taxonomic entity like Genera) leading to the inclusion of spurious indirect connections. On the other hand, direct association methods use conditional independence to avoid the detection of correlated but indirectly connected taxonomic entities [31, 47]. A null model is created by re-sampling and permuting the counts matrix and recalculating the correlations (see next section for details on network analysis and statistics). These permuted association matrices are used to calculate the significance of the inferred correlations by calculating the p-values against this null model [46]. Brown's p-value merging method [91] is used for combining p-values from the pairwise correlations methods to obtain a consensus p-value, which can be used to filter for significance. The permutations and p-value calculations are only performed on the correlations-based methods. In the final module of this step, the consensus algorithms are used to create the final consensus network using associations from all the network inference methods (except Pearson and Spearman, by default). The outputs of this step are co-occurrence association networks encoded in the JSON format (refer to Supplementary section) and which can also be exportable to a variety of network formats. The list of tools used in this step, along with their modules and references are provided in Table 1.

Consensus network and p-value merging

The consensus methods combine networks inferred from both correlation-based and direct association methods. First, for the correlation-based methods, we calculate p-values using null models and then merge the p-values using Brown's p-value merging method [92, 93]. Second, we filter all the inferred networks based on an association strength threshold of 0.1 and a p-value cutoff of 0.05. Finally, we apply the consensus algorithms we have developed on these filtered networks. These steps are elaborated on in the subsequent sections.

Notation

This section defines the notation used below to describe the consensus network algorithm used in the MiCoNE pipeline. Note that all networks to be compared were updated to have the same number of nodes.

w, the number of co-occurrence networks to be integrated into the consensus network (by default, is equal to the total number of network inference methods excluding Spearman and Pearson, 8)

q, the number of unique nodes across all w co-occurrence networks

N^i, the matrix of edge weights for the i^{th} co-occurrence network. This is a $q \times q$ matrix, where $i \in \{1, \ldots, w\}$. $N^i_{a,b}$ represents edge (a, b) in network i

P^i, the matrix of p-values for all edges of the i^{th} co-occurrence network. This is a $q \times q$ matrix, where $i \in \{1, \ldots, w\}$

$\bar{N}^i$, the "flattened" version of the adjacency matrix N^i into a $q^2 \times 1$ column vector, where all columns are stacked onto each other into a q^2 long vector. Element $\bar{N}^i_j$ corresponds to the j^{th} edge in the i^{th} network.

$\bar{P}^i$, the "flattened" version of the adjacency matrix P^i into a $q^2 \times 1$ column vector, where all columns are stacked onto each other into a q^2 long vector.

Permutations and p-value calculation

For all correlation-based methods $k \leq w$, 1000 permutations of the original OTU counts data were generated [46]. The correlations in the permuted OTU tables are recalculated using the different correlation-based algorithms. Finally, the p-value is determined based on how often a more extreme association is observed for randomly permuted data. Note that, all the direct association-based methods used in the study have their own regularization methods built in and hence do not need to undergo this procedure.

p-value merging

The next step in the consensus algorithm workflow is to merge the p-values for the networks generated by the correlation-based methods. This step is performed using the Brown's p-value merging method [92, 93].

As described in more detail in the Supplementary section and in the original reference [92], the final combined p-value is given by:

$$\begin{aligned} \hat{P}_j &= 1.0 - \Phi_{2f}\left(\psi/c\right) \\ \text{where, } \psi &= -2\sum_{i=1}^{k}\log(\bar{P}_j^i) \quad \text{and} \quad \Phi_{2f} = \text{CDF}\left(\chi^2_{2f}\right) \end{aligned} \tag{1}$$

where, $\hat{P}_j$ is the combined p-value for the edge j, f is the number of degrees of freedom, and c is a scale factor.

Note that we do not use Pearson and Spearman methods in the p-value merging step to determine the consensus network. These methods are only used for demonstration and comparison. The combined p-values are used to threshold for significance right before the consensus algorithm is applied to the inferred networks.

Consensus methods

The consensus algorithm was designed to increase the precision (number of true positives) at the end of the network inference step. For this purpose, we developed two simple algorithms that combine the edges reported by the different network inference tools. Both the algorithms make use of a user-defined parameter θ ($0 \leq \theta \leq 1$), in order to threshold the edges from the individual methods. The inputs to both the algorithms are the co-occurrence networks (association matrices) $\bar{N}^i$ (flattened version of N^i) generated by each method i, and the threshold parameter θ. Here, the $\bar{N}^i$ each have the same set of nodes q and only differ by the value of the association inferred between every pair. Networks that do not have a particular node, are updated such that the node is added as an isolated component. In this manner, $\bar{N}^i_j$ represents edge j in network i.

Note that the consensus method is only used to filter relevant interactions. If a given pair of nodes is inferred to have edges that satisfy the consensus requirements, all corresponding edges from the w networks will be returned by the algorithm, as a multigraph. Based on this approach, MiCoNE reports as the default output, the consensus network where each edge is annotated with weights (correlations for the correlation-based methods and direct associations for the other methods) from all the methods used in the consensus algorithm.

Algorithm 1 - Simple voting: The simple voting method performs a voting-based consensus to determine whether an edge will exist between a given node-pair in the final consensus network [71, 72]. For each pair of nodes, we determine the number of network inference methods that report an edge j between them, i.e. $\bar{N}^i_j, \forall i \in \{1, \ldots, w\}$. Each node-pair will have an edge in the final consensus network if the number of reported edges is larger than the threshold (Equation 3).

The number of reported edges is computed as follows:

For each edge j, we obtain M_j which represents the number of networks in which edge j is

reported. Formally, M_j is calculated as the following function:

$$M_j = f(g(\bar{N}_j^{i=1}), \ldots, g(\bar{N}_j^{i=w})) \tag{2}$$

where, g and f are defined as follows:

$$g(x) = \begin{cases} 0, \text{ if } x = 0, \\ -1, \text{ if } x < 0, \\ 1, \text{ if } x > 0 \end{cases}$$

and

$$f(x_1, \ldots, x_w) = max\left(\#(i \mid x_i = -1), \#(i \mid x_i = 1)\right)$$

where, $\#$ refers to the cardinality of the set.

The edge j is selected to be present in the final consensus network if the number of networks in which j appears is greater than a threshold, i.e:

$$M_j \geq \lfloor \theta \times w \rfloor \tag{3}$$

where, θ is the user-defined threshold parameter.

The simple voting method returns the union of the networks when $0 \leq \theta \leq \frac{1}{w}$ and will return the intersection when $\frac{(w-1)}{w} \leq \theta \leq 1$. In general, if $\frac{(n-1)}{w} \leq \theta \leq \frac{n}{w}$, this algorithm will report an edge in the consensus network when at least n network inference methods report this edge.

Algorithm 2 - Scaled-sum method: This algorithm generates a consensus network based on the sum of all edges (weights of associations) reported between a pair of nodes [71, 72]. Since in

generating a consensus network using this method we sum the edges reported by direct association methods with those from correlation-based methods, summing of the edges is preceded by a pre-processing step, in which all networks are re-scaled.

First, the network generated by each network inference method ($\bar{N}^i$) is re-scaled into a normalized version ($\bar{S}^i$), as follows:

$$\bar{S}^i = \frac{\bar{N}^i}{\max(\mid \bar{N}^i \mid)}, \quad \forall i \in 1, \ldots, w \tag{4}$$

In this way, it is guaranteed that $\max(\mid \bar{S}^i \mid) = 1$.

Next, for each edge j, we sum the weights of all reported edges from the different networks.

$$s_j = \sum_{i=1}^{w} \bar{S}^i_j \tag{5}$$

An edge j will be included in the consensus network if s_j passes a threshold.

$$\mid s_j \mid > (w - 1) \times \theta \tag{6}$$

The advantage of this method over the simple voting method is that it also takes into account the strength of the association reported for that particular node in the inferred networks.

Network variability

Notation

This section defines the notation used for the network variability analysis performed for Figure 6.

W, the number of co-occurrence networks generated from all possible combinations of tools and parameters in the workflow. Note that this is different from w, which counted only the different network inference modules.

Q, the number of unique nodes across all W networks.

N^i, the edge weights of the i^{th} co-occurrence network represented as a $Q \times Q$ adjacency matrix, where $i \in 1, \ldots, W$. $N^i_{a,b}$ represents the edge (a, b) in network i

$\bar{N}^i$, the "flattened" version of the adjacency matrix N^i into a $Q^2 \times 1$ column vector, where all columns are stacked onto each other into a Q^2 long vector.

Principal Component Analysis and variability calculation

In order to compare across different networks and calculate the degree of variability induced by the choice of different modules, we organized multiple networks into a single mathematical structure that we could use for linear regression. First, we obtained the co-occurrence network $\bar{N}^i$ for each of the W possible tool and parameter combinations in the workflow. We then constructed a matrix $\bar{\mathbf{N}}$ whose i^{th} column is the flattened version of the i^{th} network, i.e. the column vector $\bar{N}^i$. Therefore, $\bar{N}^i_j$ is the weight of edge j in the network i. $\bar{N}^i_j$ is assigned a value of 0 if edge j did not exist in network i but was present in one of the other networks. Note that row j of $\bar{\mathbf{N}}$, $\bar{N}_j$ is the vector that encodes the values of edge j across all the networks.

$$\bar{\mathbf{N}} = \begin{bmatrix} \bar{N}^1_1 & \bar{N}^2_1 & \cdots & \bar{N}^W_1 \\ \bar{N}^1_2 & \bar{N}^2_2 & \cdots & \bar{N}^W_2 \\ \vdots & \vdots & \vdots & \vdots \\ \bar{N}^1_{Q^2} & \bar{N}^2_{Q^2} & \cdots & \bar{N}^W_{Q^2} \end{bmatrix}$$

To infer the variability contributed due to the different steps in the pipeline we can perform a linear regression on each edge in $\bar{\mathbf{N}}$ and a subsequent ANOVA to extract the within-group variances. A major issue with this approach is that the possibility of correlations existing between the edges of the network could lead to inaccurate estimates of the variance if a linear model were used to directly model the relationships between edges and steps in the workflow. Therefore, in order to remedy

this issue, we performed a PCA (Principal Component Analysis) on the matrix $\bar{\mathbf{N}}$ to obtain the $\mathbf{C}$ matrix ($W \times c$) of components for each network, such that we reduce the dimensions from the Q^2 dimensional edge space to a c dimensional component space.

We then use linear regression to express each component C_j (where $j \in 1 : c$) as a linear function of categorical variables that describe the possible options in each of the steps of the pipeline.

In particular, we infer parameters α_j such that:

$$C_j = \sum_{i=1}^{5} \left(\alpha_j^{DC(i)} \delta_j^{DC(i)} \right) + \sum_{i=1}^{2} \left(\alpha_j^{CC(i)} \delta_j^{CC(i)} \right) + \sum_{i=1}^{3} \left(\alpha_j^{TA(i)} \delta_j^{TA(i)} \right) + \sum_{i=1}^{2} \left(\alpha_j^{OP(i)} \delta_j^{OP(i)} \right) + \sum_{i=1}^{10} \left(\alpha_j^{NI(i)} \delta_j^{NI(i)} \right) + \epsilon_j \quad (7)$$

where, α_i are the coefficients of the regression, ϵ_i are the residuals and δ_i are the indicator variables that correspond to the processes utilized in the pipeline used to create the network N_i; for example, $\delta_i^{DC(1)} = 1$ if the DC(1) process was used in the generation of the network N^i.

Here,

1. $DC(i) \in \{$CR, OR, DN, D2, DB$\}$
2. $CC(i) \in \{$remove bimera, uchime-denovo$\}$
3. $TA(i) \in \{$NaiveBayes(GG), NaiveBayes(SILVA), BLAST(NCBI)$\}$
4. $OP(i) \in \{$Filter(on), Filter(off)$\}$
5. $NI(i) \in \{$SparCC, propr, Spearman, Pearson, SpiecEasi, COZINE, HARMONIES, SPRING, mLDM, FlashWeave$\}$

The variance contributed by each step of the pipeline was calculated for every component in

689 $\mathbf{C}$ matrix through ANOVA using the Python statsmodels [94] package and is shown in Figure 6A.
690 The total variance for the network was calculated by adding the variances for each connection and
691 normalizing with the degrees of freedom. The merged network table $\bar{\mathbf{N}}$ was used as the input to the
692 PCA analysis to generate Figure 6B.

693 ## Synthetic interaction data

694 We generated synthetic interaction data using two methodologies previously used for benchmarking
695 network inference methods.

696 The first method, "seqtime" [73], used generalized Lotka-Volterra (gLV) equations to model
697 the microbial community dynamics and utilized the Klemm–Eguıluz algorithm to generate a
698 clique-based interaction network [26]. We used the seqtime R package to simulate communities
699 with number of species (N) varying from 10 to 150 (10, 25, 50, 100, 150 and 200). The initial
700 species concentrations were randomly sampled from a Poisson distribution and the simulation was
701 rerun to generate a number of samples (S) varying from 50 to 500 (50, 100, 200, 500) for different
702 communities. The abundance values of the species in the community at the end of the simulation
703 time were used to create the OTU table.

704 The second method, "NorTA", used the Normal to Anything (NorTA) approach coupled with
705 a given interaction network topology to generate the abundance distribution of the microbial
706 community [47]. We used the spieceasi R package [47] to simulate communities with different
707 network topologies (scale-free, cluster, block, Erdos-Renyi, band and hub) and target abundance
708 distributions (Negative Binomial, Poisson, Zero-Inflated Negative Binomial). The OTU table was
709 generated using the American Gut Project example in the spieceasi package (`amgut1.filt`) with
710 the default parameter options.

711 For each method, we generated the OTU table depicting the abundances of species and used this

as input to generate association networks using MiCoNE pipeline. The interaction matrix was used as the source of expected (true) interactions and the associations predicted using MiCoNE were the source of predicted interactions. Finally, for each dataset we evaluated the precision and sensitivity of the associations predicted by the individual network inference methods as well as the consensus (Figures 5, S5, and S6).

Statistical analyses

DC step

In order to compare the representative sequences generated by the various methods in the DC step, we employed both the weighted [58] (Figure 2A) and unweighted UniFrac method [59] (Figure 2B). The UniFrac distance metric (unique fraction metric) is a beta-diversity measure that computes the distance between two sets of taxa as the fraction of the branch length of the tree that leads to descendants from either one environment or the other, but not both [59]. The weighted UniFrac distance metric takes into account the abundances of the representative sequences when calculating shared and unshared branch lengths, whereas the unweighted UniFrac distance metric does not and hence gives equal weights to each sequence. In Figure 2 the distances between methods are the distance between the reference sequence distribution for a pair of methods averaged over every sample in the dataset. All UniFrac calculations were performed using the `scikit-bio` [95] v0.5.6 Python package.

TA step

In Figure 3C, we used the Bray-Curtis distance metric to calculate the distance between the predicted (using the taxonomy databases in the TA step) and expected taxonomic distribution. The Bray-Curtis distance is used to quantify the compositional dissimilarity between two different taxonomic

distributions defined by vectors u and v. It is defined as:

$$d = \frac{\sum_i |u_i - v_i|}{\sum_i |u_i + v_i|}$$

The Bray-Curtis distance calculations were performed using the `scipy` [64] v1.8.0 Python package.

NI step

In Figure 5 we evaluated the precision and sensitivity of the inferred association networks (using the various network inference algorithms and the consensus methods) against the original interaction network used to create the taxonomic distribution. We used the following formulations of precision and sensitivity to calculate the accuracy of the predictions:

Precision = $\frac{TP}{TP+FP}$

Sensitivity = $\frac{TP}{FN+TP}$

where, TP - true positives, FP - false positives and FN - false negatives

Code and Data Availability

Pipeline: https://github.com/segrelab/MiCoNE

Documentation: https://micone.readthedocs.io

Data and scripts: https://github.com/segrelab/MiCoNE-pipeline-paper

Synthetic data and scripts: https://github.com/segrelab/MiCoNE-synthetic-data

Acknowledgments

We are grateful to members of the Segrè lab for helpful discussions and for feedback on the manuscript. This work was partially funded by grants from the National Institutes of Health

(National Institute of General Medical Sciences, award R01GM121950; National Institute of Dental and Craniofacial Research, award number R01DE024468; National Institute on Aging, award number UH2AG064704; and National Cancer Institute, grant number R21CA260382), the U.S. Department of Energy, Office of Science, Office of Biological & Environmental Research through the Microbial Community Analysis and Functional Evaluation in Soils SFA Program (m-CAFEs) under contract number DE-AC02-05CH11231 to Lawrence Berkeley National Laboratory, the National Science Foundation (grants 1457695, NSFOCE-BSF 1635070 and the NSF Center for Chemical Currencies of a Microbial Planet) and the Human Frontiers Science Program (RGP0020/2016 and RGP0060/2021). DK acknowledges support by the Kilachand Multicellular Design Program graduate fellowship. KSK was supported by Simons Foundation Grant #409704, by the Research Corporation for Science Advancement through Cottrell Scholar Award #24010, by the Scialog grant #26119, and by the Gordon and Betty Moore Foundation grant #6790.08.

Contributions

Designed the research project: DK, KK, DS, ZH, CDL. Performed analysis: DK, GB. Wrote the first draft of the manuscript: DK. Revised and wrote the final version of the manuscript: DK, DS, KK.

References

1. Lima-Mendez, G. *et al.* Determinants of Community Structure in the Global Plankton Interactome. *Science* **348,** 1262073 (May 22, 2015).
2. Thompson, L. R. *et al.* A Communal Catalogue Reveals Earth's Multiscale Microbial Diversity. *Nature* **551,** 457. ISSN: 0028-0836 (Nov. 2017).
3. Royo-Llonch, M. *et al.* Compendium of 530 Metagenome-Assembled Bacterial and Archaeal Genomes from the Polar Arctic Ocean. *Nature Microbiology* **6,** 1561–1574. ISSN: 2058-5276. pmid: 34782724 (Dec. 2021).
4. Tedersoo, L. *et al.* Fungal Biogeography. Global Diversity and Geography of Soil Fungi. *Science (New York, N.Y.)* **346,** 1256688. ISSN: 1095-9203. pmid: 25430773 (Nov. 28, 2014).
5. Danko, D. *et al.* A Global Metagenomic Map of Urban Microbiomes and Antimicrobial Resistance. *Cell* **184,** 3376–3393.e17. ISSN: 0092-8674, 1097-4172. pmid: 34043940 (June 24, 2021).
6. McLellan, S. L., Fisher, J. C. & Newton, R. J. The Microbiome of Urban Waters. *International microbiology : the official journal of the Spanish Society for Microbiology* **18,** 141–149. ISSN: 1139-6709. pmid: 27036741 (Sept. 2015).
7. Human Microbiome Project Consortium, B. A. *et al.* A Framework for Human Microbiome Research. *Nature* **486,** 215–21. ISSN: 1476-4687. pmid: 22699610 (June 2012).
8. Zelezniak, A. *et al.* Metabolic Dependencies Drive Species Co-Occurrence in Diverse Microbial Communities. *Proceedings of the National Academy of Sciences of the United States of America* **112,** 6449–6454. ISSN: 1091-6490. pmid: 25941371 (May 19, 2015).
9. Ghoul, M. & Mitri, S. The Ecology and Evolution of Microbial Competition. *Trends in microbiology* **24,** 833–845. ISSN: 1878-4380. pmid: 27546832 (Oct. 2016).
10. Coyte, K. Z. & Rakoff-Nahoum, S. Understanding Competition and Cooperation within the Mammalian Gut Microbiome. *Current biology: CB* **29,** R538–R544. ISSN: 1879-0445. pmid: 31163167 (June 3, 2019).
11. D'Souza, G. *et al.* Ecology and Evolution of Metabolic Cross-Feeding Interactions in Bacteria. *Natural Product Reports* **35,** 455–488. ISSN: 0265-0568 (May 2018).
12. Hu, T., Chitnis, N., Monos, D. & Dinh, A. Next-Generation Sequencing Technologies: An Overview. *Human Immunology. Next Generation Sequencing and Its Application to Medical Laboratory Immunology* **82,** 801–811. ISSN: 0198-8859 (Nov. 1, 2021).
13. Buermans, H. P. J. & den Dunnen, J. T. Next Generation Sequencing Technology: Advances and Applications. *Biochimica et Biophysica Acta (BBA) - Molecular Basis of Disease. From Genome to Function* **1842,** 1932–1941. ISSN: 0925-4439 (Oct. 1, 2014).

14. Narihiro, T. & Kamagata, Y. Genomics and Metagenomics in Microbial Ecology: Recent Advances and Challenges. *Microbes and environments* **32,** 1–4. ISSN: 1347-4405. pmid: 28367917 (2017).

15. Ju, F. & Zhang, T. 16S rRNA Gene High-Throughput Sequencing Data Mining of Microbial Diversity and Interactions. *Applied Microbiology and Biotechnology* **99,** 4119–4129. ISSN: 1432-0614 (May 1, 2015).

16. Jovel, J. *et al.* Characterization of the Gut Microbiome Using 16S or Shotgun Metagenomics. *Frontiers in microbiology* **7,** 459. ISSN: 1664-302X. pmid: 27148170 (2016).

17. Quince, C., Walker, A. W., Simpson, J. T., Loman, N. J. & Segata, N. Shotgun Metagenomics, from Sampling to Analysis. *Nature Biotechnology* **35,** 833–844. ISSN: 1546-1696 (9 Sept. 2017).

18. Sedlar, K., Kupkova, K. & Provaznik, I. Bioinformatics Strategies for Taxonomy Independent Binning and Visualization of Sequences in Shotgun Metagenomics. *Computational and Structural Biotechnology Journal* **15,** 48–55. ISSN: 2001-0370 (Jan. 1, 2017).

19. Gilbert, J. A. *et al.* Meeting Report: The Terabase Metagenomics Workshop and the Vision of an Earth Microbiome Project. *Standards in Genomic Sciences* **3,** 243–248. ISSN: 1944-3277. pmid: 21304727 (Dec. 25, 2010).

20. Proctor, L. M. *et al.* The Integrative Human Microbiome Project. *Nature* **569,** 641–648. ISSN: 1476-4687 (7758 May 2019).

21. Lloyd-Price, J., Abu-Ali, G. & Huttenhower, C. The Healthy Human Microbiome. *Genome medicine* **8,** 51. ISSN: 1756-994X. pmid: 27122046 (2016).

22. Lubbe, A., Bowen, B. P. & Northen, T. Exometabolomic Analysis of Cross-Feeding Metabolites. *Metabolites* **7,** E50. ISSN: 2218-1989. pmid: 28976938 (Oct. 4, 2017).

23. Jian, X. *et al.* Microbial Microdroplet Culture System (MMC): An Integrated Platform for Automated, High-throughput Microbial Cultivation and Adaptive Evolution. *Biotechnology and Bioengineering* **117,** 1724–1737. ISSN: 0006-3592 (June 2020).

24. Hsu, R. H. *et al.* Microbial Interaction Network Inference in Microfluidic Droplets. *Cell Systems* **9,** 229–242.e4. ISSN: 24054720. pmid: 31494089 (Sept. 2019).

25. Zuñiga, C., Zaramela, L. & Zengler, K. Elucidation of Complexity and Prediction of Interactions in Microbial Communities. *Microbial Biotechnology* **10,** 1500–1522 (2017).

26. Röttjers, L. & Faust, K. From Hairballs to Hypotheses–Biological Insights from Microbial Networks. *FEMS Microbiology Reviews* **42,** 761–780. ISSN: 1574-6976 (Nov. 2018).

27. Faust, K. *et al.* Microbial Co-occurrence Relationships in the Human Microbiome. *PLOS Computational Biology* **8,** e1002606. ISSN: 1553-7358 (July 12, 2012).

28. Lee, K. K., Kim, H. & Lee, Y.-H. Cross-Kingdom Co-Occurrence Networks in the Plant Microbiome: Importance and Ecological Interpretations. *Frontiers in Microbiology* **13.** ISSN: 1664-302X (2022).

29. Faust, K. & Raes, J. Microbial Interactions: From Networks to Models. *Nature Reviews. Microbiology* **10,** 538–550. ISSN: 1740-1534. pmid: 22796884 (July 16, 2012).

30. Ma, B. *et al.* Earth Microbial Co-Occurrence Network Reveals Interconnection Pattern across Microbiomes. *Microbiome* **8,** 82. ISSN: 2049-2618. pmid: 32498714 (June 4, 2020).

31. Menon, R., Ramanan, V. & Korolev, K. S. Interactions between Species Introduce Spurious Associations in Microbiome Studies. *PLOS Computational Biology* **14** (ed Allesina, S.) e1005939. ISSN: 1553-7358 (Jan. 2018).

32. Gilbert, J. A. *et al.* Microbiome-Wide Association Studies Link Dynamic Microbial Consortia to Disease. *Nature* **535,** 94–103. ISSN: 14764687. pmid: 27383984 (2016).

33. Wu, G. D. *et al.* Comparative Metabolomics in Vegans and Omnivores Reveal Constraints on Diet-Dependent Gut Microbiota Metabolite Production. *Gut* **65,** 63–72. ISSN: 0017-5749. pmid: 25431456 (Jan. 2016).

34. Wang, B., Yao, M., Lv, L., Ling, Z. & Li, L. The Human Microbiota in Health and Disease. *Engineering* **3,** 71–82. ISSN: 20958099 (Feb. 2017).

35. Belizário, J. E. & Napolitano, M. Human Microbiomes and Their Roles in Dysbiosis, Common Diseases, and Novel Therapeutic Approaches. *Frontiers in microbiology* **6,** 1050. ISSN: 1664-302X. pmid: 26500616 (2015).

36. Fierer, N. Embracing the Unknown: Disentangling the Complexities of the Soil Microbiome. *Nature Reviews Microbiology* **15,** 579–590. ISSN: 1740-1534 (10 Oct. 2017).

37. Jiao, S., Chen, W. & Wei, G. Resilience and Assemblage of Soil Microbiome in Response to Chemical Contamination Combined with Plant Growth. *Applied and Environmental Microbiology* **85.** ISSN: 10985336. pmid: 30658982 (Mar. 2019).

38. Friedman, J. & Alm, E. J. Inferring Correlation Networks from Genomic Survey Data. *PLoS Computational Biology* **8** (ed von Mering, C.) e1002687. ISSN: 1553-7358 (Sept. 2012).

39. Bharti, R. & Grimm, D. G. Current Challenges and Best-Practice Protocols for Microbiome Analysis. *Briefings in Bioinformatics* **2019,** 1–16. ISSN: 1477-4054 (Dec. 2019).

40. Pollock, J., Glendinning, L., Wisedchanwet, T. & Watson, M. The Madness of Microbiome: Attempting To Find Consensus "Best Practice" for 16S Microbiome Studies. *Applied and Environmental Microbiology* **84,** e02627–17 (Mar. 19, 2018).

41. Caporaso, J. G. *et al.* QIIME Allows Analysis of High-Throughput Community Sequencing Data. *Nature Methods* **7,** 335–336. ISSN: 1548-7091 (May 2010).

42. Callahan, B. J. *et al.* DADA2: High-resolution Sample Inference from Illumina Amplicon Data. *Nature Methods* **13,** 581–583. ISSN: 1548-7091 (July 2016).

43. Amir, A. *et al.* Deblur Rapidly Resolves Single-Nucleotide Community Sequence Patterns. *mSystems* **2,** e00191–16. ISSN: 2379-5077. pmid: 28289731 (Apr. 2017).

44. Bokulich, N. A. *et al.* Optimizing Taxonomic Classification of Marker-Gene Amplicon Sequences with QIIME 2's Q2-Feature-Classifier Plugin. *Microbiome* **6,** 90. ISSN: 2049-2618. pmid: 29773078 (May 17, 2018).

45. Weiss, S. *et al.* Normalization and Microbial Differential Abundance Strategies Depend upon Data Characteristics. *Microbiome* **5,** 27. ISSN: 2049-2618 (Mar. 3, 2017).

46. Watts, S. C., Ritchie, S. C., Inouye, M. & Holt, K. E. FastSpar: Rapid and Scalable Correlation Estimation for Compositional Data. *Bioinformatics (Oxford, England)* (ed Stegle, O.) ISSN: 1367-4803 (Aug. 2018).

47. Kurtz, Z. D. *et al.* Sparse and Compositionally Robust Inference of Microbial Ecological Networks. *PLOS Computational Biology* **11** (ed von Mering, C.) e1004226. ISSN: 1553-7358 (May 2015).

48. Tackmann, J., Matias Rodrigues, J. F. & von Mering, C. Rapid Inference of Direct Interactions in Large-Scale Ecological Networks from Heterogeneous Microbial Sequencing Data. *Cell Systems* **9,** 286–296.e8. ISSN: 24054712 (Sept. 2019).

49. Keegan, K. P., Glass, E. M. & Meyer, F. MG-RAST, a Metagenomics Service for Analysis of Microbial Community Structure and Function. *Methods in Molecular Biology (Clifton, N.J.)* **1399,** 207–233. ISSN: 1940-6029. pmid: 26791506 (2016).

50. Gonzalez, A. *et al.* Qiita: Rapid, Web-Enabled Microbiome Meta-Analysis. *Nature Methods* **15,** 796–798. ISSN: 1548-7105 (10 Oct. 2018).

51. Bolyen, E. *et al.* Reproducible, Interactive, Scalable and Extensible Microbiome Data Science Using QIIME 2. *Nature Biotechnology* **37,** 852–857. ISSN: 1546-1696 (8 Aug. 2019).

52. Golob, J. L., Margolis, E., Hoffman, N. G. & Fredricks, D. N. Evaluating the Accuracy of Amplicon-Based Microbiome Computational Pipelines on Simulated Human Gut Microbial Communities. *BMC Bioinformatics* **18,** 283. ISSN: 1471-2105 (2017).

53. Weiss, S. *et al.* Correlation Detection Strategies in Microbial Data Sets Vary Widely in Sensitivity and Precision. *The ISME journal* **10,** 1–13. ISSN: 1751-7362. pmid: 26905627 (2016).

54. Hu, Z. *et al. A Resource for the Comparison and Integration of Heterogeneous Microbiome Networks* preprint (Systems Biology, Aug. 7, 2022).

55. Kang, D.-W. *et al.* Microbiota Transfer Therapy Alters Gut Ecosystem and Improves Gastrointestinal and Autism Symptoms: An Open-Label Study. *Microbiome* **5,** 10. ISSN: 2049-2618 (Dec. 2017).

56. Bokulich, N. A. *et al.* Mockrobiota: A Public Resource for Microbiome Bioinformatics Benchmarking. *mSystems* **1.** ISSN: 2379-5077. pmid: 27822553 (2016).

57. DeSantis, T. Z. *et al.* Greengenes, a Chimera-Checked 16S rRNA Gene Database and Workbench Compatible with ARB. *Applied and environmental microbiology* **72,** 5069–72. ISSN: 0099-2240. pmid: 16820507 (July 2006).

58. Lozupone, C. A., Hamady, M., Kelley, S. T. & Knight, R. Quantitative and Qualitative Beta Diversity Measures Lead to Different Insights into Factors That Structure Microbial Communities. *Applied and environmental microbiology* **73,** 1576–85. ISSN: 0099-2240. pmid: 17220268 (Mar. 2007).

59. Lozupone, C. & Knight, R. UniFrac: A New Phylogenetic Method for Comparing Microbial Communities. *Applied and environmental microbiology* **71,** 8228–35. ISSN: 0099-2240. pmid: 16332807 (Dec. 2005).

60. Nearing, J. T., Douglas, G. M., Comeau, A. M. & Langille, M. G. Denoising the Denoisers: An Independent Evaluation of Microbiome Sequence Error-Correction Approaches. *PeerJ* **6,** e5364. ISSN: 2167-8359 (Aug. 2018).

61. Quast, C. *et al.* The SILVA Ribosomal RNA Gene Database Project: Improved Data Processing and Web-Based Tools. *Nucleic Acids Research* **41,** D590–D596. ISSN: 0305-1048 (Nov. 2012).

62. Sayers, E. W. *et al.* Database Resources of the National Center for Biotechnology Information. *Nucleic acids research* **37,** D5–15. ISSN: 1362-4962. pmid: 18940862 (Database issue Jan. 2009).

63. Balvočiūtė, M. & Huson, D. H. SILVA, RDP, Greengenes, NCBI and OTT — How Do These Taxonomies Compare? *BMC Genomics* **18,** 114. ISSN: 1471-2164 (Mar. 2017).

64. Virtanen, P. *et al.* SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods* **17,** 261–272. ISSN: 1548-7091, 1548-7105 (Mar. 2, 2020).

65. Quinn, T. P., Richardson, M. F., Lovell, D. & Crowley, T. M. Propr: An R-package for Identifying Proportionally Abundant Features Using Compositional Data Analysis. *Scientific Reports* **7,** 16252. ISSN: 2045-2322 (1 Nov. 24, 2017).

66. Ha, M. J., Kim, J., Galloway-Peña, J., Do, K.-A. & Peterson, C. B. Compositional Zero-Inflated Network Estimation for Microbiome Data. *BMC Bioinformatics* **21,** 581. ISSN: 1471-2105. pmid: 33371887 (Suppl 21 Dec. 28, 2020).

67. Jiang, S. *et al.* HARMONIES: A Hybrid Approach for Microbiome Networks Inference via Exploiting Sparsity. *Frontiers in Genetics* **11.** ISSN: 1664-8021 (2020).

68. Yoon, G., Gaynanova, I. & Müller, C. L. Microbial Networks in SPRING - Semi-parametric Rank-Based Correlation and Partial Correlation Estimation for Quantitative Microbiome Data. *Frontiers in Genetics* **10.** ISSN: 1664-8021 (2019).

69. Yang, Y., Chen, N. & Chen, T. Inference of Environmental Factor-Microbe and Microbe-Microbe Associations from Metagenomic Data Using a Hierarchical Bayesian Statistical Model. *Cell systems* **4,** 129–137.e5. ISSN: 2405-4712. pmid: 28125788 (Jan. 2017).

70. Lex, A., Gehlenborg, N., Strobelt, H., Vuillemot, R. & Pfister, H. UpSet: Visualization of Intersecting Sets. *IEEE transactions on visualization and computer graphics* **20,** 1983–1992. ISSN: 1941-0506. pmid: 26356912 (Dec. 2014).

71. *Fuzzy Sets and Their Extensions: Representation, Aggregation, and Models: Intelligent Systems from Decision Making to Data Mining, Web Intelligence, and Computer Vision* (eds Bustince, H., Herrera, F. & Montero, J.) *Studies in Fuzziness and Soft Computing* **220.** 674 pp. ISBN: 978-3-540-73722-3 (Springer, Berlin ; New York, NY, 2008).

72. Tsarev, R. Y., Durmuş, M. S., Üstoglu, I. & Morozov, V. A. Application of Majority Voting and Consensus Voting Algorithms in N-version Software. *Journal of Physics: Conference Series* **1015,** 042059. ISSN: 1742-6588, 1742-6596 (May 2018).

73. Faust, K. *et al.* Signatures of Ecological Processes in Microbial Community Time Series. *Microbiome* **6,** 120. ISSN: 2049-2618 (Dec. 2018).

74. Shaffer, J. P. *et al.* Standardized Multi-Omics of Earth's Microbiomes Reveals Microbial and Metabolite Diversity. *Nature Microbiology* **7,** 2128–2150. ISSN: 2058-5276 (12 Dec. 2022).

75. Ho, L. K. H. *et al.* Gut Microbiota Changes in Children with Autism Spectrum Disorder: A Systematic Review. *Gut Pathogens* **12,** 6. ISSN: 1757-4749 (Feb. 3, 2020).

76. Chi, L. *et al.* Porphyromonas Gingivalis-Induced Cognitive Impairment Is Associated With Gut Dysbiosis, Neuroinflammation, and Glymphatic Dysfunction. *Frontiers in Cellular and Infection Microbiology* **11.** ISSN: 2235-2988 (2021).

77. Shannon, P. *et al.* Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks. *Genome research* **13,** 2498–2504 (2003).

78. Himsolt, M. *GML: A Portable Graph File Format* in (2010).

79. Peschel, S., Müller, C. L., von Mutius, E., Boulesteix, A.-L. & Depner, M. NetCoMi: Network Construction and Comparison for Microbiome Data in R. *Briefings in Bioinformatics.* ISSN: 1477-4054 (bbaa290 Dec. 3, 2020).

80. Fisher, C. K. & Mehta, P. Identifying Keystone Species in the Human Gut Microbiome from Metagenomic Timeseries Using Sparse Linear Regression. *PLOS ONE* **9,** e102451. ISSN: 1932-6203 (July 23, 2014).

81. Pacheco, A. R., Pauvert, C., Kishore, D. & Segrè, D. Toward FAIR Representations of Microbial Interactions. *mSystems* **0,** e00659–22 (Aug. 25, 2022).

82. Hirano, H. & Takemoto, K. Difficulty in Inferring Microbial Community Structure Based on Co-Occurrence Network Approaches. *BMC Bioinformatics* **20,** 329. ISSN: 1471-2105 (June 13, 2019).

83. Goberna, M. & Verdú, M. Cautionary Notes on the Use of Co-Occurrence Networks in Soil Ecology. *Soil Biology and Biochemistry* **166,** 108534. ISSN: 0038-0717 (Mar. 1, 2022).

84. Sung, J. *et al.* Global Metabolic Interaction Network of the Human Gut Microbiota for Context-Specific Community-Scale Analysis. *Nature Communications* **8,** 15393. ISSN: 2041-1723 (1 June 6, 2017).

85. *Qiita - Open-Source Microbial Study Management Platform*

86. Tommaso, P. D. *et al.* Nextflow: A Tool for Deploying Reproducible Computational Pipelines. *F1000Research* **4** (July 2015).

87. Ii, M. S. R. *et al.* RESCRIPt: Reproducible Sequence Taxonomy Reference Database Management. *PLOS Computational Biology* **17,** e1009581. ISSN: 1553-7358 (Nov. 8, 2021).

88. McMurdie, P. J. & Holmes, S. Waste Not, Want Not: Why Rarefying Microbiome Data Is Inadmissible. *PLOS Computational Biology* **10,** e1003531. ISSN: 1553-7358 (Apr. 3, 2014).

89. Chao, A. & Jost, L. Coverage-Based Rarefaction and Extrapolation: Standardizing Samples by Completeness Rather than Size. *Ecology* **93,** 2533–2547. ISSN: 1939-9170 (2012).

90. McDonald, D. *et al.* The Biological Observation Matrix (BIOM) Format or: How I Learned to Stop Worrying and Love the Ome-Ome. *GigaScience* **1,** 2047-217X-1–7. ISSN: 2047-217X (Dec. 1, 2012).

91. Brown, M. B. 400: A Method for Combining Non-Independent, One-Sided Tests of Significance. *Biometrics. Journal of the International Biometric Society* **31,** 987–992. ISSN: 0006-341X. JSTOR: 2529826 (1975).

92. Poole, W., Gibbs, D. L., Shmulevich, I., Bernard, B. & Knijnenburg, T. A. Combining Dependent P-values with an Empirical Adaptation of Brown's Method. *Bioinformatics (Oxford, England)* **32,** i430–i436. ISSN: 1367-4803 (Sept. 2016).

93. Faust, K. & Raes, J. CoNet App: Inference of Biological Association Networks Using Cytoscape. *F1000Research* **5,** 1519. ISSN: 2046-1402. pmid: 27853510 (Oct. 14, 2016).

94. Seabold, S. & Perktold, J. *Statsmodels: Econometric and Statistical Modeling with Python* in *9th Python in Science Conference* (2010).

95. Scikit-bio development team, T. *Scikit-Bio: A Bioinformatics Library for Data Scientists, Students, and Developers* version 0.5.7. 2022.

96. Cordasco, G. & Gargano, L. *Community Detection via Semi-Synchronous Label Propagation Algorithms* in *2010 IEEE International Workshop on: Business Applications of Social Network Analysis (BASNA)* 2010 IEEE International Workshop on: Business Applications of Social Network Analysis (BASNA) (Dec. 2010), 1–8.

97. Hagberg, A. A., Schult, D. A. & Swart, P. J. *Exploring Network Structure, Dynamics, and Function Using NetworkX* in *Proceedings of the 7th Python in Science Conference* (eds Varoquaux, G., Vaught, T. & Millman, J.) (Pasadena, CA USA, 2008), 11–15.

98. Fisher, R. A. 224A: Answer to Question 14 on Combining Independent Tests of Significance. (1948).

99. Kost, J. T. & McDermott, M. P. Combining Dependent P-Values. *Statistics & Probability Letters* **60,** 183–190. ISSN: 0167-7152 (Nov. 15, 2002).

100. Olesen, S. W., Duvallet, C. & Alm, E. J. DbOTU3: A New Implementation of Distribution-Based OTU Calling. *PLoS ONE* **12,** 1–13. ISSN: 19326203 (2017).

101. Schloss, P. D. *et al.* Introducing Mothur: Open-Source, Platform-Independent, Community-Supported Software for Describing and Comparing Microbial Communities. *Applied and environmental microbiology* **75,** 7537–41. ISSN: 1098-5336. pmid: 19801464 (Dec. 2009).

102. Park, S.-C. & Won, S. Evaluation of 16S rRNA Databases for Taxonomic Assignments Using a Mock Community. *Genomics & Informatics* **16,** e24. ISSN: 1598-866X. pmid: 30602085 (Dec. 2018).

103. Murali, A., Bhargava, A. & Wright, E. S. IDTAXA: A Novel Approach for Accurate Taxonomic Classification of Microbiome Sequences. *Microbiome* **6,** 140. ISSN: 2049-2618 (Aug. 9, 2018).

104. Matias Rodrigues, J. F., Schmidt, T. S. B., Tackmann, J. & von Mering, C. MAPseq: Highly Efficient k-Mer Search with Confidence Estimates, for rRNA Sequence Analysis. *Bioinformatics* **33,** 3808–3810. ISSN: 1367-4803 (Dec. 1, 2017).

105. Gwak, H.-J. & Rho, M. Data-Driven Modeling for Species-Level Taxonomic Assignment From 16S rRNA: Application to Human Microbiomes. *Frontiers in Microbiology* **11.** ISSN: 1664-302X (2020).

106. Ritari, J., Salojärvi, J., Lahti, L. & de Vos, W. M. Improved Taxonomic Assignment of Human Intestinal 16S rRNA Sequences by a Dedicated Reference Database. *BMC Genomics* **6,** 1056. ISSN: 1471-2164 (Dec. 2011).

107. R Marcelino, V., Holmes, E. C. & Sorrell, T. C. The Use of Taxon-Specific Reference Databases Compromises Metagenomic Classification. *BMC genomics* **21,** 184. ISSN: 1471-2164. pmid: 32106809 (Feb. 27, 2020).

Tables and Figures

Table 1: **Tools used in the MiCoNE pipeline**. The tools highlighted in gray are the defaults for the pipeline that are recommended based on the benchmarks with the mock and synthetic datasets. The consensus algorithm in the Network Inference (NI) step incorporates all the modules (permutations, direct association, and correlation-based) to generate the consensus network.

Figure 1: **The workflow of the MiCoNE pipeline**. The steps of the workflow can be broken down into five major groups: **(SP) S**equence **P**rocessing, **(DC) D**enoising and **C**lustering, **(TA) T**axonomy **A**ssignment, **(OP) O**TU and ESV **P**rocessing, and **(NI) N**etwork **I**nference. Each step incorporates several processes (blue boxes), each of which in turn has several alternative algorithms for the same task (indicated by the text to the right of the blue boxes). Each arrow describes the data that is being passed from one step to another. The inputs to the pipeline are 16S rRNA sequencing reads, and the final output is the consensus network generated from the inferred co-occurrence networks. For details on each process and the different outputs, see Methods.

Figure 2: **The representative sequences generated by the different denoising and clustering methods differ in their identification of sequences that are low in abundance**. **(A)** The average weighted UniFrac distance between the representative sequences shows that the representative sequences and their compositions are fairly identical between the methods (with the exception of Deblur (DB) due to the low ESV count). **(B)** The relatively larger average unweighted UniFrac distance indicates that methods differ in their identification of sequences that are lower in abundance. The number of OTUs or ESVs generated by the respective methods are provided in the parenthesis next to their names. The data used for the analysis in (A, B) were the samples from the fecal microbiome transplant (FMT) dataset [55], containing both healthy subjects and subjects with autism spectrum disorder (ASD). **(C, D)** The distributions of the average weighted and unweighted UniFrac distance between the predicted sequence profile and the expected sequence profile in the mock datasets. The average weighted UniFrac distances show that de novo (DN) and open reference (OR) were the best-performing methods in most of the datasets, while they are the worst-performing methods under the unweighted UniFrac metric. The good performance of dada2 (D2) under both distance metrics combined with its approach of identifying ESVs using de novo methods, prompts us to use it as the default method for the DC step. The data used for the analysis in (C, D) were the mock4, mock12, and mock16 datasets from mockrobiota [56].

Figure 3: **Taxonomic reference databases vary in terms of their taxonomy assignments below the Order level**. **(A)** The taxonomic assignments of the top 50 representative sequences using the three different reference databases. This result illustrates how the same sequences are assigned to different genera under different databases. A significant portion of the representative sequences are assigned to an "unknown" Genus in two of three databases (GG and NCBI). The number of assigned genera for each database is displayed at the top of each column. **(B)** The number of representative sequences assigned to the same taxonomic label when using different reference databases (for the top 100 sequences). The mismatches are fewer at higher taxonomic levels, but, even at the Order level there exists greater than 51% of mismatches, demonstrating the poor agreement in taxonomic labels assigned by the different databases. The data used for the analysis in (A, B) were samples (healthy and ASD) from the FMT dataset. **(C)** The Bray-Curtis dissimilarity between the predicted taxonomy profile and expected taxonomy profile in the mock datasets shows that there is no singular best choice of database for every dataset, as all the databases show similar performances. The GG database and the Naive Bayes classifier are chosen as the defaults for the TA step of MiCoNE due to their popularity. The datasets used for the analysis in (C) were the mock datasets from mockrobiota.

Figure 4: **Networks generated using different network inference methods show notable differences in terms of edge-density and connectivity**. **(A)** The nine different networks (excluding mLDM) generated by the different network inference methods. The nodes for each network (representing taxa) are arranged in the same positions in a circular layout and the differences in the connections can be directly visualized and compared. The green links are positive associations and the orange links represent negative associations. The networks look dissimilar and vary widely in terms of connectivity, and it is notable that the correlation-based methods generally produce networks with higher edge-densities. A threshold of 0.3 was set for the correlation-based methods (sparcc, propr, spearman and pearson) and a threshold of 0.01 was set for the direct association methods (flashweave, spieceasi, cozine, harmonies, and spring). **(B)** The node overlap Upset plot indicates that all the networks have a large proportion of common nodes involved in connections (33 out of 68). Conversely **(C)**, the edge overlap Upset plot shows that a very small fraction of these connections are actually shared (8 out of 202). The data used in this analysis were the healthy stool samples from the FMT dataset. mLDM is not shown in the comparisons because the algorithm failed to converge for the particular network combination used here (default setting of the MiCoNE pipeline).

Figure 5: **The associations generated by the scaled-sum consensus method show high precision in benchmarks using synthetic datasets**. The different points on the box plot show the precision (A, C) and sensitivity (B, D) of co-occurrence networks generated through individual network inference methods and consensus network construction approaches. Precision and sensitivity are estimated based on the comparisons with two sets of synthetic benchmark datasets ("NorTA" and "seqtime", see Methods). The independent algorithms chosen for the comparison are the two best-performing correlation-based (propr, sparcc) and direct-association-based (spieceasi, flashweave) methods. For consensus network inference, we used the scaled-sum (SS) and simple voting (SV) methods. A weight threshold of 0.1 and a p-value threshold of 0.05 was applied to each network before the calculation of precision and sensitivity. The purpose behind the construction of the consensus algorithms is to capture true associations in the data through the removal of associations that have a lower probability of being present in the networks inferred by different inference algorithms. Therefore, an increase in precision is followed by a decrease in sensitivity. The scaled-sum consensus method consistently obtained the overall best precision for $\theta \geq 0.333$ on both benchmark datasets. Among all the individual network inference methods, spieceasi shows the best average precision. When using the presence of edges in all inferred networks as a requirement ($\theta = 1.000$), the simple voting method also outperforms spieceasi on average precision. Therefore, we set the scaled-sum consensus method with $\theta = 0.333$ as the default tool for consensus network inference, since this option provides a good balance of precision and sensitivity (see also Figure S5 and S6). The correlation-based methods (propr, and SparCC) and the simple-voting consensus method return networks with higher sensitivities.

Figure 6: **The choice of reference database has the largest impact on network variance**. **(A)** The percentage of variance in the networks (from the FMT dataset) contributed by the Denoising and Clustering (DC), CC (chimera checking), Taxonomy Assignment (TA), OTU Processing (OP) and Network Inference (NI) steps of the pipeline calculated using ANOVA on a linear model (see Methods). A weight threshold of 0.1 and a p-value threshold of 0.05 were applied to each network before the analysis. The taxonomy database contributes most to the variance between the networks (65.4%) followed by the filtering of the counts matrix (26.8%) in the OP step. The variation due to the NI, DC and CC steps are much smaller in comparison (6.553%, 0.648%, and 0.003% respectively). The negligible fraction labeled as the residual is an artifact that arises when multiple steps are changed at the same time. **(B)** All the inferred networks generated from various combinations of tools are shown as points on a PCA plot. Each point on the PCA plot represents a network inferred using different combinations of tools and parameters that are available in the MiCoNE pipeline. The color of the points corresponds to the tools used at each step of the pipeline (DC, TA, OP, and NI). The points on the PCA plot can be grouped based on the TA step, but the extent of this separation decreases when the filtering is turned on in the OP step, confirming that the variability in the networks decreased upon filtering out the taxonomic entities at low abundance. Some algorithms, especially the direct association methods, at the NI step can also be seen to generate networks that are less variable compared to the others. The DC step does not seem to have any correlation with the variation in the networks on the PCA plot.

Figure 7: **Comparison of networks generated from the control and ASD samples of the FMT dataset using the MiCoNE pipeline**. The networks for the control (left) and ASD (right) samples were generated using the default tools and parameters recommended by the MiCoNE pipeline as described in Table 1. There are 22 unique links in the network for control samples, 12 unique links in the network for ASD subjects, and 7 edges in common between both networks. The changes in these connections can serve as potential starting points for further experimental validations or literature surveys.

Supplementary Tables and Figures

Table S1: **Table of global network metrics for networks inferred from all possible combinations of tools**. In each row, one tool in a particular step is kept constant, and the metric is calculated for every possible combination of tools for the other steps of the pipeline. Therefore, each row shows the grouped average metric for each tool in every step of the pipeline. The network inference methods show the most variation in the global network metrics compared to tools in other steps of the pipeline.

Figure S1: **The t-SNE plot of all the inferred networks clusters the networks based on the taxonomy reference database used**. Each point on the t-SNE plot represents a network inferred using different combinations of tools and parameters that are available in the MiCoNE pipeline. The points are colored by the tools and parameters used in DC step (A), TA step (B), OP step (C) and NI step (D). The separation of the points based on taxonomy reference database shows that the points cluster based on reference database in high-dimensional space.

Figure S2: **The UniFrac distance between the 1000 most abundant representative sequences is higher than that when all sequences are considered**. Each value is the average UniFrac distance between the reference sequences generated by the various methods in the DC step (similar to Figure 2). There is an increase in both weighted and unweighted UniFrac distances compared to when all the representative sequences are considered. This shows that the 1000 most abundant representative sequences generated by the DC methods are not as similar to each other. And since the weighted UniFrac is much smaller than the unweighted UniFrac distance, we can conclude that those reference sequences that are present in the middle of the abundance distribution (considering all sequences) are dissimilar.

Figure S3: **The weighted and unweighted UniFrac distances are small for the representative sequences generated using remove bimera and uchime for each denoising method**. With the exception of de novo and open reference under the unweighted UniFrac metric, all the other methods have high similarity, implying that the two chimera checking methods, uchime and remove bimera, produce similar outputs. This is especially true for the DADA2 and Deblur methods which are the recommended denoising methods in the MiCoNE pipeline. Therefore, remove bimera is recommended as the default chimera method if one is using DADA2 and uchime-denovo when one is using Deblur, since these methods were developed for these respective algorithms (QIIME2 uses uchime-denovo in the Deblur workflow).

Figure S4: **The pairwise comparison of assignments generated using different databases for all representative sequences has a higher proportion of mismatches**. The comparison made here is similar to Figure 3B, but instead of the top 100 taxonomic entities (by abundance), all the assignments from one database are matched with those from the other two databases. The higher percentage of mismatches implies that the assigned taxonomies in the more abundant sequences (top 100) match more consistently.

Figure S5: **The precision and sensitivity of the inferred networks on the "NorTA" synthetic interaction data**. The different consensus methods used are scaled-sum (SS) and simple voting (SV) methods. Pearson and Spearman methods are not used in the calculation of the consensus. Among all the independent network inference methods, SpiecEasi has the best average precision (0.944), but the overall best precision was consistently obtained by the scaled-sum method (0.956, 0.985, and 1.000). The simple voting method when using the presence of edges in all inferred networks as a requirement ($\theta = 1.000$), also outperforms SpiecEasi on average precision (0.969). Although SpiecEasi has a higher sensitivity, if the goal of network inference is to obtain the list of associations that have a high probability of existing in the real microbial community, then the consensus methods perform better.

Figure S6: **The precision and sensitivity of the inferred networks on the "seqtime" synthetic interaction data.** The different consensus methods used are scaled-sum (SS) and simple voting (SV) methods. Pearson and Spearman methods are not used in the calculation of the consensus. Among all the independent network inference methods, SpiecEasi has the best average precision (0.624), but the overall best precision was consistently obtained by the scaled-sum method (0.688, 0.820, and 1.000). The simple voting method when using the presence of edges in all inferred networks as a requirement ($\theta = 1.000$), also outperforms SpiecEasi on average precision (0.692). These results show that the scaled-sum method is not only much better suited for inferring robust and accurate interactions from count data generated from network topologies (NorTA), but it is also capable of accurately extracting real associations from Lotka-Volterra simulations.

Figure S7: **Sensitivity analysis of the default settings of the MiCoNE pipeline**. **(A)** The network constructed using the default pipeline parameters (DC=DADA2, CC=remove bimera, TA=GG, OP=Filter(on), NI=scaled-sum consensus) is compared against networks generated when one of the steps uses a different tool. The layout is created by fixing the positions of all the nodes from all networks and then drawing only the relevant edges, making the connections directly comparable. The edges colored green are positive associations and those in red are negative associations. We observe that changing the TA and OP steps leads to the creation of the most number of unique edges. **(B)** The dot plot showing the fraction of nodes (left) and edges (right) in common between the default network and the networks generated by changing one step of the default pipeline. The low value of the common fraction for TA and OP steps shows that these steps induce the biggest changes in nodes and edges. The NI step is not shown in this analysis because the consensus methods use edges from the individual network inference methods and a comparison would be biased.

Figure S8: **Network variance analysis of the stool samples from radiation-exposed bank vole (EMP dataset)**. **(A)** The percentage of variance in the networks contributed by the Denoising and Clustering (DC), CC (chimera checking), Taxonomy Assignment (TA), OTU Processing (OP) and Network Inference (NI) steps of the pipeline calculated using ANOVA on a linear model (see Methods). A weight threshold of 0.1 and a p-value threshold of 0.05 were applied to each network before the analysis. The taxonomy database contributes most to the variance between the networks (57.96%) followed by the filtering of the counts matrix (33.48%) in the OP step. The variation due to the NI, DC and CC steps are much smaller in comparison (6.69%, 1.30%, and 0.00% respectively). The negligible fraction labeled as the residual is an artifact that arises when multiple steps are changed at the same time. **(B)** All the inferred networks generated from various combinations of tools are shown as points on a PCA plot. Each point on the PCA plot represents a network inferred using different combinations of tools and parameters that are available in the MiCoNE pipeline. The color of the points corresponds to the tools used at each step of the pipeline (DC, TA, OP, and NI). The points on the PCA plot can be largely grouped based on the TA step, and the extent of this separation decreases when the filtering is turned on in the OP step. However, unlike Figure 6 B, here we observe that the networks that utilize GG and NCBI databases have the variation along similar axes in the 2 dimensional PCA plot. This could imply that the taxonomy assignments returned by these two databases for this dataset, might be more similar to each other compared to those in the FMT dataset.

Text S1: Supplemental text describing data processing, synthetic interaction data generation, network metrics, p-value merging, additional network variance analysis and supplementary discussion sections.

Supplementary Text

Processing the FMT data

Data download and pre-processing

The main biological dataset used in this study was the collection of 16S rRNA sequencing reads from stool samples (healthy and autistic individuals) for a fecal microbiome transplant study [55]. The data containing the 16S sequencing reads (V4 region) was downloaded from Qiita [50] (study ID: 10532). Only runs 2, 3, and 4 were used for the subsequent analysis as these runs consisted of paired-end sequencing data, and run 1 contained single-end data. The sample metadata was updated to contain only BMI, sex, height, weight, and experimental group. This was necessary as two of the network inference algorithms (mLDM and FlashWeave) required information about environmental heterogeneity. However, these environmental correlations were not included in the current analyses.

Processing using the MiCoNE pipeline

The data was then processed using the MiCoNE pipeline starting at the SP step and ending at the NI step with the consensus algorithm. The configuration files (main.nf and nextflow.config) used to run the MiCoNE pipeline as well the details of the pipeline execution (dag, report, timeline and trace) are in the "runs/FMT" directory of the data and scripts repository (https://github.com/segrelab/MiCoNE-pipeline-paper) The results of the pipeline execution for reproducing the analyses in the manuscript are stored on Zenodo.

Processing the mock data

Data download and pre-processing

The mock datasets, mock4, mock12 and mock16 used for this study, were obtained from mockrobiota [56]. Mock 4 is a mock community composed of 21 bacterial strains represented in equal abundances in two replicate samples, and the same strains represented in uneven abundances in two other replicate samples. Mock 12 is composed of 27 bacterial strains containing closely related taxa, the members of which were chosen in part for their well-separated 16S rRNA gene sequences. Some pairs of strains differ by as little as one nucleotide, but all the strains are distinguishable over the sequenced region of the 16S rRNA gene. Mock 16 is a mock community composed of even amounts of purified genomic DNA from 49 bacteria and 10 archaea. The datasets did not require any preprocessing and could be directly used as input to the pipeline

Processing using the MiCoNE pipeline

The data was processed using the MiCoNE pipeline starting at the SP step and ending at the OP step with the filtered taxonomic tables as the final output. The configuration files (main.nf and nextflow.config) used to run the MiCoNE pipeline as well the details of the pipeline execution (dag, report, timeline and trace) are in the "runs/mock*" directory of the data and scripts repository (https://github.com/segrelab/MiCoNE-pipeline-paper) The results of the pipeline execution for reproducing the analyses in the manuscript are stored on Zenodo.

Interpretation of Unifrac results in the DC step

In Figure S3, we observe that both the weighted and unweighted UniFrac distances are increased for the top 1000 representative sequences, implying that the top representative sequences generated by the different methods are not as similar to each other. Therefore, since the weighted UniFrac distances are lower than the unweighted distances, we conclude that the representative sequences in the middle range of the abundance distribution are those that must be the most similar between the methods.

Open-reference and de novo clustering methods perform the best under the weighted UniFrac metric and the worst (marginally) under the unweighted UniFrac metric (Figure 2C and 2D). This result can be attributed to the large number of low abundance representative sequences that are generated by these methods. Deblur performs poorly under weighted Unifrac and although its performance on the mock4 dataset is the best under unweighted UniFrac, its performance on the other datasets is average. The Deblur method returns a very small number of representative sequences (2388) and this could account for the reason for the high dissimilarity with the other methods as well as irregular performance on the mock data.

Synthetic interaction data

Data generation

The synthetic interaction data for the study were generated using two methods. The first method, "seqtime" [73] utilized generalized Lotka-Volterra (gLV) equations to model the microbial community dynamics and made use of the Klemm–Eguıluz algorithm to generate clique-based interaction networks [26]. We used the seqtime R package to simulate communities with different numbers of species and samples (see Methods for details). The second method, "NorTA" used the Normal to Anything (NorTA) approach coupled with a given interaction network topology to generate the abundance distribution of the microbial community [47]. We used the spieceasi R package to simulate communities with different abundance distributions and network topologies (see Methods for details). The scripts to generate these datasets can be found in the synthetic data and scripts repository (https://github.com/segrelab/MiCoNE-synthetic-data)

Processing using the MiCoNE pipeline

The data was processed using the MiCoNE pipeline using only the NI step with the consensus networks as the final output. The configuration files (main.nf and nextflow.config) used to run the MiCoNE pipeline as well the details of the pipeline execution (dag, report, timeline and trace) are in the "runs/norta" and "runs/seqtime" directories of the data and scripts repository (https://github.com/segrelab/MiCoNE-pipeline-paper) The results of the pipeline execution for reproducing the analyses in the manuscript are stored on Zenodo.

Network metrics

In Table S1 we show various global network metrics calculated for each tool in the pipeline. All the networks that make use of a particular tool are grouped together, and the following average metrics are calculated for each group:

1. The average shortest path length describes the average of all the shortest paths in the graph. No number is reported if the graph is not connected, therefore, the results indicate that none of the networks that make use of HARMONIES, COZINE, SPRING, SpiecEasi and Pearson are connected.
2. The average clustering is the average clustering coefficient of the graph. The closer the value is to 1.0, the more densely connected is the graph. We can observe that the networks that use correlation-based methods have the highest values while the direct association based methods have the lowest.
3. The number of connected components is the highest for the direct association based methods and the lowest for the correlation-based methods. In the case of propr, all the networks have only one giant component.
4. The modularity metric is the modularity over all partitions in a graph calculated using a label propagation algorithm [96]. Positive values imply that there are more edges between vertices of the same type than we would expect by chance, and negative implies that there are less. The networks inferred by mLDM report very few edges, and skew the average modularity scores. This could also be an artifact of incomplete convergence of the mLDM algorithm for some combinations.
5. Node connectivity refers to the minimum number of nodes that must be removed from the graph to make it disconnected. We observe that only the networks generated using propr have a high value since most of these networks are connected.
6. Degree assortativity coefficient measures the similarity of connections in the graph with respect to the node degree. Again we observe that the direct association based methods have a negative degree of assortativity, meaning that there are many hubs in these networks. The correlation-based methods have positive values implying that in these networks nodes with similar degrees attach to one another.

A weight threshold of 0.1 and a p-value threshold of 0.05 were applied to each network before the

analysis. All the metrics were calculated using the `networkx` Python package [97].

p-value merging

Fisher [98] proposed that for k independent p-values, each generated by k different methods and denoted by $\bar{P}^i$ (notations are same as used in the "Consensus network and p-value merging" subsection of the Methods). The following will hold true for the statistic Ψ:

$$\Psi = \sum_{i=1}^{k} -2\log\left(\bar{P}^i\right)$$
$$\Psi \sim \chi^2_{2k}$$

Brown [91] extended Fisher's method to dependent p-values by using a re-scaled χ^2 distribution:

$$\Psi \sim c\chi^2_{2f}$$

where, f is the degrees of freedom and c is the scale factor and are given by:

$$f = \frac{\mathrm{E}[\Psi]^2}{\mathrm{Var}[\Psi]} \quad \text{and} \quad c = \frac{\mathrm{Var}[\Psi]}{2\mathrm{E}[\Psi]} = \frac{k}{f}$$

We can calculate $E[\Psi]$ and $Var[\Psi]$ under the null hypothesis that the data are drawn from a multivariate Gaussian with some covariance matrix. We then use these values to parametrize a χ^2 distribution from which the p-value corresponding to $\frac{\psi}{c}$ can be calculated. Furthermore, Brown showed that $\mathrm{E}[\Psi]$ and $\mathrm{Var}[\Psi]$ can be calculated via the following numerical approximation:

$$\mathrm{E}[\Psi] = 2k \quad \text{and} \quad \mathrm{Var}[\Psi] = 4k + 2\sum_{i<j} \mathrm{Cov}\left(-2\log(\bar{P}^i), -2\log(\bar{P}^j)\right)$$

The above formulation was improved by Kost and McDermott [99] by further fitting a third-order polynomial to approximate the covariance

$$\mathrm{Cov}\left(-2\log(\bar{P}^i), -2\log(\bar{P}^j)\right) \approx 3.263\rho_{ij} + 0.710\rho_{ij}^2 + 0.027\rho_{ij}^3 \tag{8}$$

where, ρ_{ij} is the correlation between method i and method j

Using $E[\Psi]$ and $Var[\Psi]$ we then fit a χ^2 distribution with the parameters c and f. Note that since, in general, f will not be an integer, this should be understood as a Gamma distribution with a shape parameter f, as mentioned by Brown [91]. Using this, we calculate the test ψ and compute the p-value from the CDF of the χ^2 distribution, given in Equation 9. Therefore, the final combined

p-value [92] is then given by:

$$\hat{P}_j = 1.0 - \Phi_{2f}\left(\psi/c\right) \text{ where, } \psi = -2\sum_{i=1}^{k}\log(\bar{P}_j^i) \quad \text{and} \quad \Phi_{2f} = \text{CDF}\left(\chi^2_{2f}\right) \tag{9}$$

The p-value merging and consensus method in MiCoNE (see Methods) uses Equation 8 to estimate the covariance of the p-values and Equation 9 to merge the p-values (obtained from bootstrapping) from the different correlation methods. Note that we do not use Pearson and Spearman methods in the p-value merging step and these algorithms are only used for demonstration and comparison. The combined p-values are used to threshold for significance in the correlation-based networks during the consensus network step.

The JSON network format and network exports

The default format MiCoNE uses for storing the network files is the JSON (JavaScript Object Notation) format which is supported by the Microbial Interaction Network Database (MIND) [54]. The custom JSON schema we have designed is able to store all network-related information pertaining to nodes, links, and the metadata related to the links and datasets. Additionally, MiCoNE also supports exporting of networks into a variety of other formats such as edge lists, GML, and Cytoscape formats. Since we make use of `networkx` [97] for the export functionality, networks can be exported to all formats supported by the package. However, not all the corresponding metadata will be exported appropriately, as most formats do not support this additional metadata. The details of the format and information about importing/exporting it and other network formats can be found in the MiCoNE documentation.

Network variance analysis of stool samples from radiation-exposed bank vole (EMP dataset)

In order to verify the consistency of the network variance analysis, we processed an additional 16S rRNA dataset through the MiCoNE pipeline. Specifically, we used stool samples from radiation-exposed bank vole that were a part of the Earth Microbiome Project (EMP) [74]. We chose a dataset from the EMP because the MiCoNE pipeline inherently supports the EMP amplicon protocol with minimal pre-processing requirements. The data containing the 16S sequencing reads (V4 region) was downloaded from Qiita [50] (study ID: 13114). For the analysis, the paired-end sequencing data extracted from stools samples of radiation-exposed bank vole (named "mousseau") were chosen from run "EMP500_6-9". The data was then processed using the MiCoNE pipeline starting at the SP step and ending at the NI step with the consensus algorithm. The configuration files (main.nf and nextflow.config) used to run the MiCoNE pipeline as well the details of the pipeline execution (dag, report, timeline and trace) are in the "runs/EMP" directory of the data and scripts repository

(https://github.com/segrelab/MiCoNE-pipeline-paper)

Similar to the previous network variance analysis performed in Figure 6, we analyzed the effect of different processing methods on the inferred co-occurrence networks generated using all possible combinations of methods. Figure S8A, shows the percentage of total variation among the co-occurrence networks due to the different steps of the pipeline. The TA step, or more specifically the choice of 16S reference database, contributes the most (57.96%) to the variation in the networks, followed by the OP step (33.48%). These results are very similar to those from the network variance analysis of the FMT dataset (Figure 6A). This implies that the effects of various workflow steps on the inferred networks are fairly consistent across different datasets.

The effects of the different steps of the pipeline on the inferred networks can be visualized through dimensionality reduction. The PCA in Figure S8B shows all possible inferred networks, colored by the tools used in the DC, TA, OP, and NI steps in each subfigure. We observe that the networks largely segregate based on the database used (TA step), and the extent of this separation decreases when the filtering is turned on in the OP step. However, unlike the case of the FMT dataset (Figure 6B), here we observe that the networks that utilize GG and NCBI databases have the variation along similar axes in the 2 dimensional PCA plot. This could imply that the taxonomy assignments returned by these two databases for this dataset, might be more similar to each other compared to those in the FMT dataset. The variation in the networks due to the other workflow steps is consistent with that observed in the FMT data analysis (Figure 6B).

Supplementary discussion

It is worth pointing out some additional, more specific, conclusions stemming from the individual steps of our analysis. The different denoising/clustering methods differ mostly in their identification of sequences that are in low abundance. Hence, they do not have much of an impact on the inferred co-occurrence networks when the sequences of low abundance are removed (Figure 1). Comparison of inferred and expected reference sequences and their abundances in mock community datasets has allowed us to identify DADA2 as the method which best recapitulates the expected sequence composition. For the chimera checking module, we suggest using the remove bimera method since it was developed in conjunction with DADA2 and its performance does not significantly differ from uchime-denovo. For the current work, we have decided to focus on the tools most widely used at the time of the analysis. Some tools which were not as widely used (e.g. dbOTU3 [100]) as well as older popular methods like mothur [101] have not been included in the study but could be added into the pipelines in future updated analyses.

The choice of taxonomy database was found to be the most important factor in the inference of microbial co-occurrence networks, contributing 65.4% of the total variance. The frequent changes in the taxonomy nomenclature coupled with the frequency of updates to the various 16S reference databases create inherent differences [63] in taxonomy hierarchies in these databases. Our analysis revealed that no particular reference database performs better than the others across the different mock dataset benchmarks. The default reference database in the pipeline is currently chosen to be

the GG reference database along with the "Naive Bayes" classifier as the query tool. The reason for our choice stems from the popularity of the GG database [102] in taxonomic studies, which would enable easy comparison across datasets. However, we recommend using the SILVA database for newer studies due to its size and taxonomic comprehensiveness [87] and since GG has not been updated since 2013. Additionally, a particular database might be more appropriate than the rest based on specific requirements. For example, to generate a dataset that is compatible with the MIND platform [54], NCBI, is the most appropriate choice as it guarantees compatibility of taxonomic hierarchy and therefore comparability with other datasets. A detailed study of other taxonomy mapping approaches [103, 104] and integrative approaches combining different databases [105] might offer orthogonal information and better matches. Furthermore, we also enable users to use custom databases [87, 106] with the BLAST and Naive Bayes classifiers that are incorporated into the pipeline (from QIIME2). We suggest that the choice of the database should be made based on possible reported or inferred biases in the representation of given biomes in a specific databases [63, 87], as choosing taxon-specific databases have also been observed to compromise classification [107].

The OP step of the pipeline is second in its contribution to total network variance. This can be attributed to the large number of nodes that are added to the final networks when the filtering is turned off. Additionally, a very large number of nodes also decreases the accuracy of the network inference algorithms for the same sample size [79] and increases the computational complexity [48]. We observe that filtering out taxa that are present in low abundances in all samples increases the proportion of taxa in common between taxonomy tables generated using different reference databases (Figure S4), providing another reason for filtering. We also observe that the reduction in the number of taxa leads to a better agreement in the networks inferred through different methods (Figure 6 and Figure S1). Moreover, filtering is necessary in order to increase the power in tests of significance when the number of taxa is much greater than the number of samples.