MODELING ASSOCIATION IN MICROBIAL COMMUNITIES WITH CLIQUE LOGLINEAR MODELS

BY ADRIAN DOBRA††, CAMILO VALDES§§, DRAGANA AJDIC¶¶, BERTRAND CLARKE∥∥ AND JENNIFER CLARKE∥∥

University of Washington‡, Florida International University§, University of Miami¶ and University of Nebraska-Lincoln∥

November 27, 2018

There is a growing awareness of the important roles that microbial communities play in complex biological processes. Modern investigation of these often uses next generation sequencing of metagenomic samples to determine community composition. We propose a statistical technique based on clique loglinear models and Bayes model averaging to identify microbial components in a metagenomic sample at various taxonomic levels that have significant associations. We describe the model class, a stochastic search technique for model selection, and the calculation of estimates of posterior probabilities of interest. We demonstrate our approach using data from the Human Microbiome Project and from a study of the skin microbiome in chronic wound healing. Our technique also identifies significant dependencies among microbial components as evidence of possible microbial syntrophy.

1. Introduction. Microbiomes – the communities of micro-organisms peculiar to specific environments such as mammalian skin or managed agricultural soil – play key roles in a diverse set of biological phenomena, from plant growth to wine cultivation to human health and disease. Metagenomics is the study of genetic material recovered directly from a specific microbiome or environment without knowledge of the composition of the sample. Thus, metagenomic-based studies generate valuable information about the composition of microbiomes and differences in their composition that may be related to environmental differences. Traditionally, studying complex microbiome samples relied on intensive microbiological techniques involving the

---

*This work was supported in part by the National Science Foundation through grants DMS/MPS-1737746 and DMS-1120255 to University of Washington, and grants DMS-1410771 and DMS-1419754 to University of Nebraska-Lincoln. Address correspondence to: adobra@uw.edu.
†A. Dobra and C. Valdes contributed equally to this work, and are joint first authors
MSC 2010 subject classifications: Primary 62H17; secondary 62-09
Keywords and phrases: contingency tables, graphical models, model selection, microbiome, next generation sequencing
isolation and culturing of individual organisms followed by phenotypic or genotypic analysis. These techniques precluded microbial community profiling within a single sample. However, recent advances in high-throughput DNA sequencing technologies now permit whole-genome metagenomic sequencing (i.e., whole metagenome sequencing) without such isolation or culturing. This means that characterization of complex microbial communities is now possible.

Whole metagenome sequencing (WMS) has served as the primary tool for several high profile, collaborative research endeavors such as the U.S. National Institute of Health Human Microbiome Project (NIH HMP Working Group et al., 2009), the U.S. Department of Energy Joint Genome Institute’s Integrated Microbial Genomes (IMG) system (Markowitz et al., 2014), and the Canadian Institutes of Health Research Canadian Microbiome Initiative. Often, metagenome sequencing means that next generation sequencing (NGS) techniques are used. These techniques differ from classical Sanger sequencing in that instead of sequencing a whole DNA molecule nucleotide by nucleotide, the sequencing is done in parallel at many points of the DNA molecule resulting in short reads, or simply reads, typically ranging in length from 50 to 250 nucleotides. Usually, a key step in the analysis of NGS data is aligning the reads to a collection of consensus sequences or reference genomes for a collection of organisms. WMS is the general case for which our formal reasoning is designed: our examples use whole genome sequencing (WGS) and 16S sequencing. In WGS, sequencing libraries are prepared from the extracted whole-DNA sequences of bacteria in the sample to be analyzed. The resulting sequencing short-reads consequently represent the putative DNA sequences of the bacterial populations in the sample (Thoendel et al., 2016; Hasman et al., 2014). In contrast, 16S rRNA sequencing libraries are prepared from the sequences of the highly conserved 16S ribosomal gene. The reads from 16S sequencing represent the sequences of the 16S genes in the bacterial populations in the sample. In downstream analyses, WGS data are analyzed at the sequencing-read and genome levels, while 16S reads are assembled into clusters of reads called Operational Taxonomic Units (OTU), and analyzed as abstract representations of taxonomic groups (Nguyen et al., 2016; Charuvaka and Rangwala, 2011). WGS captures a larger region of a bacterium’s genome than 16S, and can achieve better detection given the appropriate depth and breadth of sequencing coverage (Ranjan et al., 2016).

It is well known that compositional studies of microbiomes alone provide no information about potential symbiosis, or syntrophy – settings in which the metabolic waste products from one microbe provide nutrients for another – among species or strains (Levy and Borenstein, 2013). Indeed, mi-
Microbial communities in diverse settings have been shown to form syntrophic relationships. Such relationships have been posited to drive pathogenicity (de Kievit and Iglewski, 2000; Koch et al., 2014). A simple approach to infer possible syntrophic relationships is to examine rates of co-occurrence of micro-organisms in the same habitat across samples (Hoffmann et al., 2013). However, these methods cannot be used with a single sample as they rely on co-occurrence across many samples. In addition, in most metagenomic studies based on sequencing there is a portion of sequencing reads that cannot be associated with any known microorganisms in a particular environment, and these reads are often discarded inappropriately.

In this paper, we address these limitations by proposing methods to identify bacteria and associations among them at various taxonomic levels (e.g., genus, species, or strain). In WMS, a collection of reads is sampled from a biological community within one sample. By aligning these reads to a database of microbial reference genomes, a categorical dataset showing the reference genomes to which each read aligns is obtained. In these data, one row corresponds to one read, and one column corresponds to a genome – indicating the genomes to which each read maps. The rows are independent if the reads are from different organisms and, often nearly independent even when they are from the same organism. Although initially counter-intuitive, this is seen empirically in a Bayesian context in Clarke et al. (2015). In fact, assuming independence among a large number of reads is a reasonable first approximation because i) the number of nucleotides in the DNA molecules is very large so dependence will be rare, and ii) even when reads are regarded as dependent this rests partially on their gene products. Here we are not looking at gene products so dependence among them is irrelevant, making the dependence among reads smaller than one would initially expect.

We introduce a statistical approach based on a special class of loglinear models which we call clique loglinear models as a tool for statistical inference about the presence of various strains, species, and genera of bacteria and their associations within a given taxonomic level. Our goal is to assess associations among bacteria within a single sample (or across samples), and the likelihood of a specific bacterium, including a previously unknown bacterium, being in the sample. To represent these associations, we produce connectivity graphs showing which bacterial genera (or other taxonomic unit) are related by higher order interaction terms. This is possible because a clique loglinear model is a compound of disjoint collections of higher order terms, each collection permitting all possible interactions amongst the categories at the taxonomic level under study. Clique loglinear models are a sparse subset of all hierarchical loglinear models (Bishop et al., 2007),
and this is operationally satisfactory since the associations among bacterial strains are often sparse as well. Stated in another way, the class of clique loglinear models is small enough to be tractable, yet large enough to be used for data summarization and model selection. Given the increasing speed of computing and accumulating knowledge about which bacteria are in which microbiome, this task is likely to be easier in the future than it is now.

In Section 2 we formally introduce clique loglinear models, describe their properties, and develop model selection methods. In Section 3 we present a series of simulations to verify that our methodology qualitatively generates the results one would anticipate. In Sections 4 and 5 we analyze two datasets, and interpret our results in their scientific contexts. For the first of these our results are consistent with the findings from a more traditional approach to analysis of the same data. For the second, we generate results that seem plausible given the experimental context; there is no previous analysis for comparison purposes. This shows that our modeling framework provides a viable alternative to expensive laboratory work. Finally, in Section 6 we discuss how several features of our formalism relate to the real biological questions we have addressed.

2. Analyzing NGS data using clique loglinear models. In this section we motivate and outline our methodology for using metagenomic NGS data to detect associations among, say, bacterial strains or genera.

2.1. Representing NGS data as a sparse contingency table. Because of the way sequencing reads are generated, they can match none, one, or several bacterial strains on a list \( \{ C_1, \ldots, C_B \} \) of known bacterial genomes. A sample of \( R \) reads can be represented as a \( R \times B \) matrix \( (c_{rb})_{RB} \) that we call a connectivity matrix, in which

\[
c_{rb} = \begin{cases} 
1, & \text{if read } r \text{ aligns to strain } b, \\
0, & \text{if read } r \text{ does not align to strain } b.
\end{cases}
\]

Each row of \( (c_{rb})_{RB} \) may be regarded as a vector valued outcome of the vector valued random variable \( X_B = (X_1, \ldots, X_B) \) in which \( X_b = X_b(r) \) is the indicator variable for a sampled read \( r \) to align (or match) to genome \( b \). Each outcome of \( X_B \) assumes one of \( 2^B \) patterns of zeroes and ones in \( \lambda_B = \{0, 1\}^B \). These vectors of length \( B \) generate a \( B \)-dimensional contingency table \( n_B \) in which the count \( n_B(x_B) \) in cell \( x_B \in \lambda_B \) gives the number of reads that share the same pattern of alignments to the \( B \) genomes.

We want to model the joint distribution of \( X_B \) to obtain estimates of interesting cell probabilities \( p_B(x_B) = P(X_B = x_B) \), and relations amongst
TABLE 1
An example connectivity matrix. The first read only matches bacterial strain 1, while
reads 2, . . . , R each match at least two strains.

<table>
<thead>
<tr>
<th>Read</th>
<th>Genome 1</th>
<th>Genome 2</th>
<th>. . .</th>
<th>Genome B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Match</td>
<td>No match</td>
<td>. . .</td>
<td>No match</td>
</tr>
<tr>
<td>2</td>
<td>Match</td>
<td>Match</td>
<td>. . .</td>
<td>No match</td>
</tr>
<tr>
<td>3</td>
<td>Match</td>
<td>No match</td>
<td>. . .</td>
<td>Match</td>
</tr>
<tr>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
<td>.</td>
</tr>
<tr>
<td>R</td>
<td>Match</td>
<td>Match</td>
<td>. . .</td>
<td>Match</td>
</tr>
</tbody>
</table>

Table 1

them. For instance,

\[ P(X_1 = 0, \ldots, X_B = 0) \]

is the probability that a sampled read aligns to none of the \( B \) reference genomes. If the estimate of (2.1) is high, we might infer that we have found a bacterium or other microbial source not amongst the \( C_b \)'s. By contrast,

\[ P(X_b = 1, \{X_b = 0, \forall b \neq b^*\}) \]

is the probability that a sampled read comes from \( C_b \) and does not come from any of the other \( (B - 1) \) genomes. To identify the bacteria that are most likely to be present, we choose the \( C_b \)'s with the highest values of (2.2).

Once the data form a connectivity matrix, capturing associations in the resulting multi-way contingency table can be done by determining models for the joint distributions of the observed categorical variables while recognizing that these random variables do not vary independently of each other. However, the association structure within a microbial community is likely to be sparse because most of the possible higher order interaction terms are likely to be discarded. This happens because, given the length of the list of reference genomes, most bacteria only occur jointly with a relatively small number of other bacteria. We argue that classes of hierarchical loglinear models (Bishop et al., 2007) are well suited to represent associations among bacterial taxa in a community.

2.2. Clique loglinear models. For a set \( C \subseteq B \), we denote \( \mathcal{X}_C = \{0, 1\}^{|C|} \), where \(|C|\) stands for the number of elements of \( C \). The subvector \( X_C \) of \( X_B \) takes values \( x_C \in \mathcal{X}_C \). The \( C \)-marginal \( n_C \) of \( n_B \) has cell counts \( n_C(x_C) = \sum_{x_{B \setminus C}} n_B(x_C, x_{B \setminus C}) \). The corresponding marginal cell probabilities are \( p_C(x_C) = P(X_C = x_C) \).

Consider a hierarchical loglinear model \( M \) with \( k \) generators \( C(M) = \{C_1, \ldots, C_k\} \), where \( C_j \subseteq B \), for \( j = 1, \ldots, k \), and \( k \geq 1 \) (Bishop et al.,
Under this model, the cell probabilities associated with $X_B$ are represented as (Whittaker, 1990):

$$\log p_B(x_B) = u_{\emptyset} + \sum_{\{C: \emptyset \neq C \subseteq C_j \text{ for some } j \in \{1, \ldots, k\}}} u_C(x_C). \quad (2.3)$$

Here $u_{\emptyset}$ is an intercept, and $\{u_C(x_C) : x_C \in X_C\}$ is the $|C|$-way interaction associated with the subvector $X_C$ of $X_B$. This model can be made identifiable either by imposing the sum to zero constraints $\sum_{x_C \in x_C} u_C(x_C) = 0$, or by imposing the baseline equal with zero constraints that set $u_C(x_C) = 0$ if one element of $x_C$ is zero. For the latter, the loglinear expansion (2.3) becomes:

$$\log p_B(x_B) = u_{\emptyset} + \sum_{\{C: \emptyset \neq C \subseteq C_j \text{ for some } j \in \{1, \ldots, k\}}} u_C \prod_{i \in C} x_i, \quad (2.4)$$

where $u_C = u_C(1, \ldots, 1)$.

A hierarchical loglinear model $M$ is a clique loglinear model if its generators form a partition of $\mathcal{B}$: $\bigcup_{j=1}^k C_j = \mathcal{B}$, $C_{j_1} \cap C_{j_2} = \emptyset$ for $j_1 \neq j_2$. In this case, the cell probabilities (2.4) are written as:

$$\log p_B(x_B) = u_{\emptyset} + \sum_{j=1}^k \sum_{\{C: \emptyset \neq C \subseteq C_j\}} u_C \prod_{i \in C} x_i. \quad (2.5)$$

Thus, under a clique loglinear model, the log cell probabilities are decomposed as a sum of groups of interaction terms in which each group represents a collection of categorical variables that may interact with each other in all possible ways, but do not interact at all with categorical variables in other groups. Formally, the interpretation of clique loglinear models comes from this result:

**Proposition 2.1.** Let $D_1$ and $D_2$ be two subsets of $\mathcal{B}$ that are also subsets of two different generators of a clique loglinear model $M$. Then the random sub-vectors $X_{D_1}$ and $X_{D_2}$ are independent.

**Proof.** We collapse across the levels of $X_{\mathcal{B} \setminus (D_1 \cup D_2)}$ in the loglinear expansion (2.5). The marginal cell probabilities associated with $X_{D_1 \cup D_2}$ have the form:

$$\log p_{D_1 \cup D_2}(x_{D_1}, x_{D_2}) = u_{\emptyset} + \sum_{\{C: \emptyset \neq C \subseteq D_1\}} u_C \prod_{i \in C} x_i + \sum_{\{C: \emptyset \neq C \subseteq D_2\}} u_C \prod_{i \in C} x_i.$$ 

Since the first term is a constant, the second term is a function of the levels of $X_{D_1}$, and the third term is a function of the levels of $X_{D_2}$, it follows that $X_{D_1}$ and $X_{D_2}$ are indeed independent. $\square$
A consequence of Proposition (2.1) is that the cell probabilities of \( M \) decompose as a product of marginal cell probabilities associated with its generators:

\[
p_B(x_B) = \prod_{j=1}^{k} p_{C_j}(x_{C_j}).
\]  

(2.6)

We denote by \( u_M \) all the interaction terms that appear in (2.5). Under Multinomial sampling, the log-likelihood function is written as a function of the interaction terms as follows:

\[
l(u_M, n_B) = Ru_{\emptyset} + \sum_{j=1}^{k} \sum_{C: \emptyset \neq C \subseteq C_j} u_C n_C(x_C) \prod_{i \in C} x_i.
\]

(2.7)

By using Lagrange multipliers in (2.7) and (2.6) (Whittaker, 1990), it can be shown that the MLEs of the cell probabilities under \( M \) are

\[
\hat{p}_B(x_B) = R^{-k} \prod_{j=1}^{k} n_{C_j}(x_{C_j}).
\]

(2.8)

Equation (2.8) shows that the MLEs of the cell probabilities of a clique log-linear model exist if and only if the counts in the marginal tables associated with its generators are strictly positive. This existence criterion is easily applicable in a computationally efficient manner. By contrast, determining the existence of the MLEs for arbitrary hierarchical loglinear models is a difficult problem that has been solved theoretically (Fienberg and Rinaldo, 2007). However, at the present time, there do not seem to exist any implementable algorithms for assessing the existence of MLEs of hierarchical loglinear models that are also computationally efficient when the number \( B \) of categorical variables involved is large.

We named this class of loglinear models based on the representation of the interaction structure defined by the \( u \)-terms \( u_M \) as an independence graph (Whittaker, 1990). This is an undirected graph \( G \) with vertices \( B \) and set of edges \( E \). Each element \( b \in B \) is associated with the component \( X_b \) of \( X_B \). An edge \( e = (b_1, b_2) \) is included in \( E \) if there is a generator \( C_j \) of \( M \) such that \( \{b_1, b_2\} \subseteq C_j \). Proposition (2.1) implies that the independence graph \( G \) of a clique loglinear model \( M \) has a special structure: the generators of \( M \) are the connected components of \( G \), and are also maximal complete subgraphs or cliques (Lauritzen, 1996). As such, the independence graph of a clique loglinear model is obtained by putting together complete subgraphs
without adding any edge between them. These cliques are the generators of the loglinear model, and uniquely identify it.

The class of clique loglinear models is a subset of decomposable loglinear models, which, in turn, is a subset of graphical loglinear models that are themselves a subclass of hierarchical models – see Supplementary Material (Dobra et al., 2018), Section 3.1. The restriction to clique loglinear models offers key computational advantages: in addition to an easy way to calculate the MLEs and check their existence, these models are straightforward to interpret (Proposition (2.1)), and allow the development of computationally efficient model determination algorithms that scale well when \( R \) or \( B \) become large.

The number of clique loglinear models for \( B \) categorical variables is the number of decompositions of \( B \) into positive integers (Abramowitz and Stegun, 1972):

\[
P(B) = \frac{1}{\pi \sqrt{2}} \sum_{j=1}^{\infty} \sqrt{j} A_j(B) \frac{d}{dB} \frac{\sinh\left(\frac{\sqrt{2}}{2} \left( B - \frac{1}{2} \right) \right)}{\sqrt{B - \frac{1}{2}}},
\]

where

\[
A_j(B) = \sum_{0 < h \leq (h, j) = 1} e^{\pi i \left( s(h, j) - \frac{2hn}{j} \right)}, \quad s(h, j) = \sum_{l=1}^{j-1} \frac{l}{j} \left( \left\lfloor \frac{hl}{j} \right\rfloor \right),
\]

with \((x) = x - \lfloor x \rfloor - \frac{1}{2}\) if \(x\) is an integer, and 0 otherwise, and \((h, j)\) is the greatest common divisor of \(h\) and \(j\). For example, \(P(100) = 190, 569, 292\), \(P(200) \approx 3.973e + 12\) and \(P(1000) \approx 1.321e + 19\) (Hankin, 2006). Therefore, although this is the smallest class of hierarchical loglinear models, it still contains a significantly large number of possible models that allow modeling various patterns of interactions among many categorical variables.

2.3. Existing loglinear model selection methods. Selection for loglinear models has been well studied in the literature (Edwards and Havranek, 1985; Whittaker, 1990). Since large values of \( B \) arise naturally in metagenomics, answering questions about interactions within taxonomic levels with loglinear models must involve methods for model selection in high-dimensional contingency tables. The sparsity of these tables is extremely problematic as it leads to the invalidation of asymptotic approximations to the null distribution of the generalized likelihood ratio test statistic (Fienberg and Rinaldo, 2007). Another key difficulty is the size of the space of possible loglinear
models. For example, when \( B = 5 \) the number of possible hierarchical log-linear models is 7580; for \( B = 8 \) variables this number increases to \( 5.6 \times 10^{22} \) (Dellaportas and Forster, 1999).

Because the space of possible models is extremely large, various stochastic search schemes have been used to identify models with high posterior probability. Dellaportas and Forster (1999) is a key reference, although there are other papers that develop stochastic search schemes for discrete data (Madigan and Raftery, 1994; Madigan and York, 1995, 1997; Tarantola, 2004; Dellaportas and Tarantola, 2005; Dobra and Massam, 2010). One feature of these and other stochastic searches on spaces of hierarchical, graphical and decomposable loglinear models – see, for example, Massam et al. (2009) – is that these models involve repeated transitions from one model to another model. This necessitates ensuring the next model is still in the target space of models. For instance, for decomposable graphs, transitioning from a current decomposable graph to another decomposable graph involves checks that the decomposability property is preserved. While such checks can be done relatively quickly for graphs with few vertices, for graphs that involve hundreds of vertices the running time of stochastic searches increases rapidly. Since considerable computational effort is required to visit loglinear models sequentially by adding and removing higher order terms, restricting the model space to, say, clique loglinear models provides a necessary reduction in the running time of model determination algorithms. This makes the required computations intensive yet feasible.

Copula Gaussian graphical models (Dobra and Lenkoski, 2011) have successfully been used to analyze a 16-dimensional table. Several related methods based on efficiently determining Gaussian graphical models with many variables in a latent space have also been subsequently proposed (Mukherjee and Rodriguez, 2016; Mohammadi et al., 2017). However, conditional independence relationships in a latent space do not necessarily translate into similar relationships in the observed discrete variables space. For this reason, inferring multivariate interactions from latent Gaussian graphical models has limited practical relevance.

Dirichlet process mixture models have recently emerged in the analysis of categorical data. Canale and Dunson (2011) developed Bayesian nonparametric kernel mixtures for multivariate count data. Ultra-sparse high-dimensional contingency tables have been analyzed using probabilistic low rank tensor factorizations induced through a Dirichlet process mixture model of product multinomial distributions (Dunson and Xing, 2009; Bhattacharyya and Dunson, 2012; Kunihama and Dunson, 2013; Zhou et al., 2015). These papers present simulation studies and real-world data examples that involve
A stochastic search method for clique loglinear models. Let \( M \) vary over the collection \( \mathcal{M} \) of \( B \)-dimensional clique loglinear models for which the MLEs exist. We want to find \( M \)'s that fit the data well and are parsimonious. For this purpose we develop a stochastic search procedure based on the Bayesian information criterion (BIC). For large sample sizes, it is well known that the BIC is an approximation to the mode of a posterior distribution over a model space. The BIC is also optimal in a Bayes testing sense (Schwarz, 1978). The calculation of BIC for clique loglinear models proceeds as follows. Denote by \( \mathcal{C}(M) = \{C_1, \ldots, C_k\} \) the generators of \( M \). The MLEs of the mean cell values under \( M \) are calculated based on (2.8):

\[
\log \hat{m}_B(x_B) = \log(R\hat{p}_B(x_B)) = \sum_{j=1}^{k} \log n_{C_j}(x_{C_j}) - (k - 1) \log R,
\]

for all \( x_B \in \mathcal{X}_B \). From (2.5) we see that the number of free interaction terms that appear in \( M \) is equal with the sum of the number of nonempty subsets of the generators of \( M \). Therefore the BIC of \( M \) is given by

\[
BIC(M) = -2 \sum_{\{x_B \in \mathcal{X}_B : n_B(x_B) > 0\}} n_B(x_B) \log \hat{m}_B(x_B)
\]

\[
+ \left( \sum_{j=1}^{k} 2^{|C_j|} - k + 1 \right) \log R.
\]

Equations (2.10) and (2.11) show that the BIC of a clique loglinear model can be efficiently calculated even for large contingency tables since no iterative numerical optimization methods are involved as it would have been the case for arbitrary graphical and hierarchical loglinear models. The calculation of the log mean cell values can also be performed using a formula for decomposable loglinear models (Lauritzen, 1996), but the calculation of the
number of free interaction terms of these models would have been complicated by their overlapping sets of generators. For this reason, the calculation of BIC for clique loglinear models is easier as compared to any other loglinear model that does not belong to this class.

Consider the following distribution over $\mathcal{M}$:

\begin{equation}
\pi(\mathcal{M}) \propto \exp(-BIC(\mathcal{M})).
\end{equation}

Finding clique loglinear models with smaller values of BIC is equivalent to finding models at or close to the modes of the distribution (2.12). We can think of $\pi(\mathcal{M})$ as a posterior distribution over $\mathcal{M}$ obtained by assuming a flat prior over $\mathcal{M}$. Thus, the $\pi(\mathcal{M})$’s can be considered to be the Bayes model weights, and, in the sequel, these weights will be used to perform model averaging using Occam’s window. This methodology originates in Madigan and Raftery (1994), and has been developed in numerous other contexts, e.g., dynamic linear models (Onorante and Raftery, 2016), “large p” regression (Hans et al., 2007; Dobra, 2009), and graphical models (Madigan et al., 1995; Dobra and Massam, 2010; Lenkoski and Dobra, 2011).

Our goal is to find clique loglinear models that have large posterior weights (2.12). The largest would achieve

$$\hat{M}_{opt} = \operatorname*{arg \ max}_{\mathcal{M}} \pi(\mathcal{M}).$$

However, models that have posterior weights comparable to that of the optimal model $\hat{M}_{opt}$ are also relevant. The stochastic search algorithm we propose below is devised to seek the set of models

\begin{equation}
\mathcal{S}(c) = \left\{ \mathcal{M} \in \mathcal{M} : \pi(\mathcal{M}) \geq c\pi(\hat{M}_{opt}) \right\},
\end{equation}

where $c \in (0, 1)$ is a constant that needs to be specified before the start of the algorithm. The clique loglinear models that do not belong to $\mathcal{S}(c)$ are discarded. The idea of eliminating models with low posterior probability compared to the highest posterior probability model is based on the Occam’s window principle of Madigan and Raftery (1994).

For ease of exposition we begin by stating our procedure informally. Our stochastic search procedure moves towards models with larger values of $\pi(\mathcal{M})$. The models that are visited in a run are collected as if in a bag. Each run of the stochastic search algorithm collects models until it appears to reach a local optimum. At that point the stochastic search algorithm will likely visit only models that are already in the bag. We use many different runs, and combine all the bags of models collected in each run into a larger
bag $\mathcal{S}$. Out of this bag, we only retain those models that have comparable posterior weights with the best model identified across all runs:

$$\hat{S}(c) = \left\{ M \in \mathcal{S} : \pi(M) \geq c \max_{M \in \mathcal{S}} \pi(M) \right\}. \tag{2.14}$$

Across multiple runs that were sufficiently long, we would hope that $\hat{S}(c)$ from (2.14) will approximate well $S(c)$ in (2.13). This is very likely to happen if $\hat{M}_{\text{opt}}$ has been visited and included in $\mathcal{S}$. An empirical test for figuring out whether $\hat{M}_{\text{opt}}$ was indeed identified is to determine the proportion of runs that reached $\arg \max_S \pi(M)$. A high proportion of runs that ended up visiting the best model in $\mathcal{S}$ represents a good indication that $\hat{M}_{\text{opt}}$ might indeed be in $\mathcal{S}$. In the sequel, we perform Bayes model averaging using the models in $\hat{S}(c)$ with weights in (2.12), and this lets us estimate the quantities of interest. Models not in $\hat{S}(c)$ are discarded; this is justified if $\hat{S}(c)$ comprises most models that have large posterior probabilities.

Our stochastic algorithm for identifying $\hat{S}(c)$ from (2.13) proceeds as follows. We start with a randomly generated clique loglinear model. If any of the marginals associated with the generators of this model contain counts of zero, the MLEs of this model do not exist and another random model is generated. We repeat these steps until a valid clique loglinear model is generated; we denote this model with $M_0$. Starting with $M_0$ we generate a chain of models $x_{M_t}$ for $t = 1, 2, \ldots$. At step $t$, with equal probability, we select one of the following four ways of producing a valid (i.e., for which the MLEs (2.8) exist) candidate clique loglinear model $M'$:

(i) Split a random clique of $M_t$ into two cliques.
(ii) Join two random cliques of $M_t$ into a clique.
(iii) Switch two random elements that belong to two random cliques of $M_t$.
(iv) Move a random element of a random clique of $M_t$ to another random clique of $M_t$.

After sampling a move of type (i), (ii), (iii) or (iv), we produce a clique loglinear model $M'$ by applying a move of that type to model $M_t$. For moves of type (i), we uniformly select a clique of $M_t$ and divide the elements in that clique $C'$ into two disjoint sets $C'_1$ and $C'_2$ that become two new cliques of $M'$. The other cliques of $M_t$ are also cliques for $M'$. For moves of type (ii), we uniformly sample two cliques $C'_1$ and $C'_2$ of $M_t$, and form a new clique $C' = C'_1 \cup C'_2$. The other cliques of $M_t$ together with $C'$ are the cliques of $M'$. For moves of type (iii), we uniformly sample two cliques $C'_1$ and $C'_2$ of $M_t$, and also uniformly sample a element $v_1 \in C'_1$ and a element $v_2 \in C'_2$. We form two new cliques $C''_1 = C'_1 \setminus \{v_1\} \cup \{v_2\}$ and $C''_2 = C'_2 \setminus \{v_2\} \cup \{v_1\}$. The cliques $C''_1, C''_2$ together with the other cliques of $M_t$ give the candidate model
M'. For moves of type (iv), we uniformly sample two cliques \( C'_1 \) and \( C'_2 \) of \( M_t \), and also uniformly sample a element \( v_1 \in C'_1 \). We form two new cliques \( C''_1 = C'_1 \setminus \{v_1\} \) and \( C''_2 = C'_2 \cup \{v_1\} \). The cliques \( C''_1, C''_2 \) together with the other cliques of \( M_t \) give the candidate model \( M' \).

If the MLEs of \( M' \) do not exist, we set \( M_{t+1} = M_t \). If the MLEs of \( M' \) exist, we set \( M_{t+1} = M' \) with probability \( \min\{1, \pi(M')/\pi(M_t)\} \). Otherwise we set \( M_{t+1} = M_t \). This stochastic search algorithm typically moves to models with larger \( \pi(M) \). If the sampled candidate model \( M' \) happens to have a smaller \( \pi(M) \) than the current model \( M_t \), the algorithm could still visit it with positive probability. This is useful because sometimes models with smaller \( \pi(M) \) must be visited before finding models with larger \( \pi(M) \). This is the case when \( M \) is a local maximum but not a global maximum. The geometry of the space of models affects this: getting stuck in a local maximum is not a problem if it is a global maximum; on the other hand, the model space is discrete so it is possible that the models with the largest \( \pi(M) \) are not very similar to each other.

Only moves of type (i) and (ii) are needed to connect a clique loglinear model for which the MLE exists with any other clique loglinear model in \( \mathcal{M} \). However, we empirically found that an algorithm that included moves of types (iii) and (iv) was less likely to get stuck in local maxima of \( \pi(\cdot) \). We note that this is a stochastic search procedure, not a Markov Chain Monte Carlo procedure so that the acceptance probabilities are not relevant; see the Supplementary Material (Dobra et al., 2018), Section 3.2 for a discussion of this point. Furthermore, sparse contingency tables such as we are studying here frequently have unbalanced counts. This is rarely a problem for the BIC, as discussed in the Supplementary Material (Dobra et al., 2018), Section 3.3.

2.5. *Bayesian model selection and inference for clique loglinear models.* As we will see in the simulated and real world data analysis examples, selecting clique loglinear models based on the BIC leads to results that are easily interpretable, and are obtained with a low to moderate computational effort. However, there are several shortcomings related to the use of the BIC. First, the BIC limits the candidate clique loglinear models to those models for which the MLEs exist. Second, although asymptotically approximating posterior distributions using the BIC is a well-established technique (Berger et al., 2003), for sparse contingency tables the assumed limiting behavior might not occur. Third, the BIC is equivalent to using a uniform prior over the model space. In some applications, employing other priors over the model space could be desirable. For these reasons, in this section we describe a full Bayesian framework for model selection and inference.
In the sequel we follow Dawid and Lauritzen (1993). We start with a fictive prior table of positive numbers \( n_B^0 \). A default specification for this prior table involves setting its grand total (the sum of its cell counts, or sample size) to a value \( \alpha > 0 \) much smaller than the grand total \( R \) of the observed table \( n_B \) (e.g., \( \alpha = 1 \)), then setting all its cells to an equal value: \( n_B^0(x_B) = \alpha/|B|, \) for all \( x_B \in X_B \). The effect of the choice of the values of \( \alpha \) on the loglinear models selected based on Bayes factors has been studied empirically in Massam et al. (2009), and theoretically from a geometrical perspective in Letac and Massam (2012). These papers found that, for larger values of \( \alpha \), more interaction terms appear in the hierarchical loglinear models with the largest posterior probabilities. When \( \alpha \) becomes smaller with values close to 0, the hierarchical loglinear models selected contain fewer interaction terms that involve a smaller number of variables.

Consider a clique loglinear model \( M \) with generators \( C(M) = \{C_1, \ldots, C_k\} \). For each generator \( C \in C(M) \) of \( M \), we let \( \text{Dir}(n_C^0) \) denote the Dirichlet distribution for the marginal cell probabilities \( p_C \):

\[
P(p_C \mid n_C^0) \propto \prod_{x_C \in X_C} p_C(x_C)^{n_C^0(x_C)-1}.
\]

Since the generators of \( M \) do not overlap, the Dirichlet priors (2.15) are pairwise hyper-consistent, and define a unique prior for the cell probabilities \( p_B \) under model \( M \):

\[
P(p_B \mid n_B^0) = \prod_{j=1}^{k} P(p_{C_j} \mid n_{C_j}^0).
\]

This prior is called the hyper Dirichlet prior for \( p_B \), and it is denoted by \( \text{HyperDir}_M(n_B^0) \). The hyper Dirichlet prior is conjugate for the Multinomial likelihood, and yields a posterior distribution for the cell probabilities

\[
P(p_B \mid n_B) \propto P(n_B \mid p_B) P(p_B \mid n_B^0),
\]

that is hyper Dirichlet \( \text{HyperDir}_M(n_B^*), \) where \( n_B^* = n_B + n_B^0 \). The marginal likelihood under model \( M \) is

\[
P(n_B \mid M, n_B^0) = \prod_{j=1}^{k} P(n_{C_j} \mid n_{C_j}^0),
\]

where

\[
P(n_{C_j} \mid n_{C_j}^0) = \frac{\Gamma(\alpha)}{\Gamma(\alpha + R)} \prod_{x_C \in X_C} \frac{\Gamma(n_{C_j}(x_C)+\alpha)}{\Gamma(n_{C_j}^0(x_C)+\alpha)}.
\]
for each $C \in \mathcal{C}(M)$. The posterior distribution of $M$ is

$$(2.18) \quad P(M \mid n_S) \propto P(n_S \mid M, n_S^0) P(M),$$

where $P(M)$ is a prior distribution on the set of all clique loglinear models. A prior on this space that penalizes for model complexity is (Jones et al., 2005):

$$(2.19) \quad P(M) \propto \prod_{j=1}^{k} \left( \frac{\beta}{1 - \beta} \right)^{\binom{|C_j|}{2}},$$

where $|C|$ denotes the number of elements in the set $C$, and $\beta \in (0, 1)$ represents the probability of including an additional edge in the corresponding independence graph. A related prior with desirable multiplicity correction properties is obtained by marginalizing out $\beta$ in (2.19) with respect to a $\text{Beta}(a, b)$ distribution (Carvalho and Scott, 2009). A more general type of priors on the models space is defined through product distributions for random partitions (Barry and Hartigan, 1992):

$$(2.20) \quad P(M) \propto \prod_{j=1}^{k} q(C_j),$$

where $q(C)$ is a cohesion function for subsets $C \subseteq B$. Priors (2.20) exploit the special structure of clique loglinear models that partition variables into cliques. The stochastic search method from Section 2.4 can be used to identify clique loglinear models with high posterior probabilities. For this purpose, the distribution (2.12) needs to be replaced by the posterior distribution (2.18). The rest of the model search procedure remains unchanged.

3. Simulation results. To benchmark the performance of the method, we created a synthetic experiment with a known community dependency structure. We obtained 2,273 bacterial genomes from the National Center for Biotechnology Information (NCBI) GenBank database. These genomes were collected from GenBank’s complete genome set, or those genomes that are considered to have a final DNA sequence per genomic structure (chromosomes and/or plasmids). From these 2,273 genomes, we randomly chose 200 genomes and created a population connectivity matrix representing 1,000 synthetic genomic reads that indicates the connectivity among the genomes. Each simulated read has a corresponding row in the connectivity matrix with a match for at least one genome; this is indicated by 1s. The matrix is based on a file supplied to the simulation program that indicates
which genomes are present and what cliques they form. If two genomes are in the same clique they are given 1s for 80% of their joint reads (as assigned by i.i.d. Bernoulli($0.8$) random variables). The remaining cells in the $1000 \times 200$ connectivity matrix are randomly filled with 0s and 1s sampled from a Bernoulli($0.2$) distribution. This procedure gives a connectivity matrix consistent with a chosen clique structure on genomes. Note that not all 200 genomes are shown because only some were in nontrivial cliques. Further details are given in the Supplementary Material (Dobra et al., 2018), Section 4.

We use the connectivity matrix to generate a connectivity graph. A connectivity graph has vertices that represent distinct organisms and edges that represent higher order interactions between their reads. This definition will be made more precise in Section 4 when we deal with real data. The connectivity graph for the synthetic reads is in Figure 1. We verified that two genomes are connected in Figure 1 if and only if they are connected by reads in the connectivity matrix.

To check that our method is able to recover the connectivity graph from Figure 1, we applied the stochastic search method described in Section 2.4 using 200 chains each of length 200,000. We set $c = 10^{-4}$ in (2.14). We calculated BMA estimates of the posterior inclusion probabilities of edges in the corresponding independence graph based on the models in the set $p^c$. This generates a connectivity graph that is shown in Figure 1. We comment that there is nothing unique about the value $10^{-4}$; it was chosen for convenience, was not discredited by the individual posterior probabilities we found, and a sensitivity analysis showed that it was a reasonable choice within a range of possible cutoff values. In practice, the choice of cutoff value would be data-driven to ensure appropriate robustness of the inferences.

In Figure 1, two genomes are connected if and only if the sum of the posterior weights of the best models containing higher order terms between the two genomes is above $0.1$. Loosely, this is intuitively equivalent to saying that the posterior probability that the two genomes are associated (in the sense of higher order interaction terms in loglinear models) is at least $0.1$. We see that true connectivity structure has been fully recovered by the clique loglinear models, but the clique loglinear models identified additional interactions that were absent from the true connectivity graph. So, in this simple case, our method returns the full set of cliques built into the data. Nevertheless, even though clique loglinear models are a restricted class of loglinear models, they can over-detect interactions. The reason is that in the simulated connectivity matrix, the cliques that are present are strongly built into the matrix; they will be found as long as enough reads are included.
FIG 1. Actual and recovered connectivities in the simulated data. The connectivities that were recovered by the clique loglinear models but were absent in the true connectivity graph are marked with dotted edges and round vertices. The connectivities marked with solid edges and square vertices give the true connectivity graph.
However, cliques that are not present may also be found by our method from the random 1s in the connectivity matrix that do not correspond to genomes in any pre-defined cliques.

The average computation time for calculating the BIC of a clique loglinear model for 200 genomes and 1000 reads was 0.26 seconds. Our implementation in plain R is slower due to the limits imposed by this software package, and could be made significantly faster if implemented in C. For a discussion of comparison with other methods, please see Section 3.5 in the Supplementary Material (Dobra et al., 2018).

4. Example: characterizing associations in a microbial community. The Human Microbiome Project (HMP) is an ongoing collaborative study funded by the U.S. National Institutes of Health (NIH) to provide data and tools for studying the role of human microbiomes in human health and disease. Started in 2007 it has generated ground-breaking publications (Fierer et al., 2010; Zhao et al., 2012; Minot et al., 2013), and a plethora of metagenomic data on human microbiomes. Our method from Section 2.4 can represent the associations from an HMP sample with an independence graph so we can infer the bacterial taxa present and their associations.

Human metagenome sample SRS015072, obtained from the vaginal microbiome of a female participant of the HMP Core Microbiome Sampling Protocol A (HMP-A) dbGaP study, was downloaded via FTP from the HMP Data Analysis and Coordination Center (DACC). The sample consisted of 495,256 paired-end, 100 base pair reads (with an average mate-distance of 81bp) sequenced and provided in Illumina FASTQ format. These reads were aligned to the collection of 4,940 bacterial genomes, from the Integrated Microbial Genomes and Metagenomes (IMG, version 4.0) database (Markowitz et al., 2014) using the Bowtie2 aligner (Langmead and Salzberg, 2012). Of the sample reads, 369,633 aligned to one or more of the reference genomes. The number of reads that aligned to each bacterial strain, species, and genera was calculated and connectivity tables were generated for analysis at the genera level.

The first step in each analysis is to identify those genera that cannot be involved in higher order interactions (i.e., cannot be part of a clique with two or more vertices). Note any two genera that define a marginal two-by-two table (disregarding all other genera) whose counts are not strictly positive cannot be part of the same clique because the MLEs of any clique loglinear model that involves that two-way interaction do not exist. We refine the definition of a connectivity graph as follows: it is a graph whose vertices correspond to categorical variables, i.e., the presence of a genus. Given two
vertices, there is an edge joining them if the two-way marginal contingency table associated with the two categorical variables contains only strictly positive counts. Within each analysis we ran the stochastic search from Section 2.4 for 100,000 iterations from 100 random starting clique graphs.

A total of 95 genera had component species or strains to which reads aligned. Two genera were said to be connected if and only if each had at least one strain that shared a read. The genera with shared reads are shown in Figure 2. It is seen that 15 genera did not share reads across other genera (though each did within its own genus). This shows two facts: i) 15 genera can be dropped from subsequent analysis at the genus level, and ii) the hairball showing the 80 genera that share at least one read is complex enough that further analysis is worth doing, i.e., it is worthwhile to use clique loglinear models to seek higher order interaction terms.

Table 1 from Supplementary Material (Dobra et al., 2018) gives the degree (the number of neighbors) of each element (genus) in the raw connectivity graph. Several of these genera, including Lactobacillus, Prevotella, and Staphylococcus, have been identified as common members of vaginal microbiome communities (Huang et al., 2014). Note that the 15 isolated genera in Figure 2 cannot form cliques with any of the other genera because in the reduced table, i.e., grouping all strains into genera, the two-way marginals they form contain two counts of zero. As such, they cannot be accommodated by
The best clique loglinear model identified for the genera data.

clique loglinear models – except as cliques of size one – and dropping them amounts to a significant reduction in computational running time. This is important because the number of possible clique loglinear models increases rapidly with the number of vertices.

We have reduced the data to an 80-dimensional contingency table. It has 377 cells with strictly positive counts. The largest positive count is 332,117 while the second largest is 11,614. We perform 100 runs of 100,000 iterations of the stochastic search procedure from Section 2.4. Of all the clique loglinear models visited, we found 1133 whose BICs were within $c = 10^{-4}$ of the BIC of the best clique loglinear model identified across all 100 chains. As in Section 3, we used an Occam’s window form of BMA limited to the best models visited while renormalizing (2.12) to reflect this. The clique structure of the best model is shown in Figure 3. Forty of the 100 runs identified the same best model, while the rest of the chains were trapped in local modes, and did not reach this model.

We have also generated the independence graph resulting from the Occam’s window BMA in Figure 4. The strength of the connectivities between genera are indicated by different types of lines, and reflect ranges of posterior probabilities calculated from the Occam’s window BMA probability using
Fig 4. Pairs of variables that have strictly positive posterior probabilities of belonging to an higher order term of a clique loglinear model in the genera data under the Occam’s window BMA. Solid, dot dashed, dotted, and dashed lines denote posterior probabilities that belong to the intervals $(0.9, 1]$, $(0.5, 0.9]$, $(0.1, 0.5]$, and $(0, 0.1]$, respectively.

the models amongst the 1133 best models for which a given collection of higher order terms is present. Observe that Figure 4 does not have a clique structure because BMAs of clique loglinear models do not in general form another clique loglinear model. More specifically, in both Figures 3 and 4, the edges correspond to pairs of variables/genera that have strictly posterior probabilities of belonging to a collection of higher order terms in the 1133 clique loglinear models as evaluated by the posterior probabilities given by the Occam’s window BMA.

Several of the links in Figure 4 are supported by biological findings regarding the vaginal microbiome. For instance, i) Ureaplasma and Mycoplasma are bacterial genera from the same bacterial family and are commonly found in the reproductive tract of both men and women, ii) Polaromonas and Verminephrobacter, and Yersinia and Caldicellulosiruptor, are from the same bacterial family and have been validated by experimentation, and iii) Melissococcus and Carnobacterium are both main genera producers of bacteriocins, ribosomally synthesized antibacterial peptides/proteins that either kill or inhibit the growth of closely related bacteria and are considered antimicrobial microbes.

We also produced estimates of the probabilities in (2.1) and (2.2) which,
from here on, will be referred to as individual existence probabilities. The BMA estimates of the top 5 individual existence probabilities are as follows: Lactobacillus 0.86, “Unknown” 0.08, Pseudomonas 0.05, Acinobacter 0.01, and Gardnerella 0.002 (probabilities do not add to one because of rounding). While we do not have standard errors for these estimates, it is obvious that Gardnerella is present in only trace amounts and the presence of an unidentified genus is not zero. We return to the interpretation of unidentified genera in Section 6, but note that our findings are consistent with those from analyses of vaginal microbiome samples based on 16S rDNA sequence data that also identified the presence of previously unknown bacterial taxa (Fettweis et al., 2012).

5. Example: the diabetic foot wound microbiome. One of the complications of diabetes, particularly in elderly patients, is the development and impaired healing of foot ulcers. Diabetes is the primary cause of non-traumatic lower extremity amputations in the United States; approximately 14-24% of patients with diabetes who develop a foot ulcer eventually require an amputation. A diabetic foot ulcer is an open sore or wound that occurs in about 15% of patients with diabetes, usually on the bottom of the foot.

It has been hypothesized that an altered skin microbiome may play a role in the compromised healing of diabetic foot ulcers (Smith et al., 2016). The purpose of this study was to investigate the bacteria involved in chronic wound healing. Samples were taken from three locations – the wound bed, the wound edge, and the peripheral healthy skin of the foot – of 10 patients at two time points, the time of initial visit and one week after the initial visit. Half of the patients were considered healers, and the remaining patients were considered nonhealers based on clinical assessment of their wounds. Samples were prepared and submitted for V4 16S rRNA gene sequencing with the Illumina MiSeq platform, with services provided by Second Genome, Inc. Out of the total of 60 samples (20 from wound base, 20 from wound edge, and 20 from healthy skin), we could not PCR-amplify bacterial DNA from 10 health skin samples. Sequences from the remaining 50 samples passed quality filtering and were mapped to a set of representative consensus sequences to generate an abundance table of Operational Taxonomic Units (OTUs); an OTU is simply a cluster of closely related reads. This table was analyzed using an overdispersed Poisson model (Robinson et al., 2010) to identify OTUs that were significantly differentially expressed between healers and nonhealers at each of the three sample locations (FDR corrected p-value < 0.05). The results consisted of three lists of significant OTUs, one for each location. Further details can be found in the Supplementary Material (Dobra
et al., 2018), Section 5, that also extend our method to the comparison of two populations.

Our clique loglinear analysis is based on the counts of the number of sequencing reads assigned to significant OTUs for each sample, with a separate table for the significant OTUs from each location. This analysis is different from our previous example in that: i) our interest is on the association among samples that may be reflected in components of the microbiome, and ii) any associations will be based on sharing of OTUs across samples as opposed to sharing of reads across genomes.

An initial exploratory data analysis using hierarchical clustering and principal components analysis revealed that samples from the same subject cluster together, and that subjects cluster into two groups with patients 4 and 5 (healers) and patient 7 (nonhealer) forming a cluster distinct from the remaining subjects (see Supplementary Material (Dobra et al., 2018), Section 5). For each of our three analyses (one for each significant OTU list) all samples could form cliques with any of the other samples because all two-way marginals contain only nonzero counts. We ran the stochastic search from Section 2.4 for 100,000 iterations from 100 random starts. Due to the smaller size of the table relative to the previous example (50 vs. 80 binary variables) we noted convergence to a best graph in less than 50,000 iterations.

The strongest factor in clique formation is subject/patient origin followed by sample location; the distinction between healers and nonhealers is not evident despite the focus on significant OTUs.

Although some cliques appear in all three best graphs, e.g., samples from the wound edge of patient 2, most cliques shift subtly with the changes in the significant OTUs; see Figure 5. For example, all samples from patient 6 form a single clique in the best graph based on OTUs that are significantly different in the wound bed between healers and nonhealers. However, in the best graph based on significant OTUs in the wound edge, one of the wound bed samples from patient 6 forms a clique with the wound bed samples from patient 11, while the remaining samples from patient 6 maintain a clique. Not surprisingly, the cliques involving samples from patient 6 change again in the best graph based on significant OTUs in healthy samples (note that patient 6 has no healthy samples). As expected from our exploratory analysis, samples from patients 4, 5, and 7 formed cliques only among themselves and not with samples from other patients.

Thus, we have analyzed the data to identify the OTUs and to search for associations among samples. To the best of our knowledge, the data in this example have not been analyzed in any way analogous to our methodology before, either biologically or statistically. Thus, we are unable to corroborate
The best clique loglinear model of samples from the wound microbiome based on significant OTUs in the wound bed between healers and nonhealers. MP = microbiome patient. The sample location is wound bed (a and b), wound edge (c and d), or healthy skin (e and f).

our findings from other sources although a priori our findings do not appear unreasonable. This example demonstrates that our methodology has the potential to obviate a lot of expensive microbiological work.

6. Discussion. We have developed a statistical methodology for contingency table analysis based on clique loglinear models. This methodology can be broadly used in the context of high dimensional tables, and it accounts for model uncertainty by Bayesian model averaging. Our methodology can infer the presence/absence of specific taxa as well as associations among members of taxa. We have demonstrated the use of our approach in both simulated and real data contexts relevant to applications to metagenomics. We described an efficient method for selecting clique loglinear models based on the BIC. We also provided a Bayesian framework for model determination of clique loglinear models. We note that the advantages of using the Bayesian framework versus the BIC are that all clique loglinear models will be candidate models (as opposed to those models for which the MLEs exist), and that more flexible priors on the model space (2.19) and (2.20) can be employed. On the other hand, hyper Dirichlet priors for cell probabilities together with priors on the model space must be specified, hence the
sensitivity of the results with respect to these priors should be investigated. In this sense, the BIC could be seen as a conceptually simpler alternative to the full Bayesian framework especially from the point of view of applied researchers.

An issue that repeatedly comes up in this sort of analysis is how to account for the dependence among reads. As noted in Section 1, this dependence is frequently small – and in our experience, the higher the quality of the data the smaller the dependence among reads is, though it is never zero. As a first approximation, therefore, assuming independence is reasonable and parallels the bag-of-words approach that has been applied with much success in natural language processing. In recent years this has been improved to random ‘N-grams’ and an analogous improvement may be possible with NGS reads. Of more immediate relevance, the bag-of-words model has been applied to several branches of bioinformatics with success (Lovato, 2015). It is important to distinguish between genuine associations among taxa and simply having reads in common for some other reason – syntrophy or evolution for example – something our methodology does not address. However, this level of study remains in its infancy.

An important question is how much information is really contained in the reads. In this context, one can ask if there is adequate read converge to infer reliably which genomes are present in the sample and hence in the population. Obviously, this is a function of the number of reads, diversity within the sample, and complexity of the population. This is not a question that can be addressed statistically after the reads have been generated, although in principle it could be partially addressed at the design stage of the read generation. Read coverage will typically be incomplete and typically will be a limitation on analytic methods. This may increase the uncertainty of downstream inferences but the task is to reflect uncertainty accurately not to under-represent it. Methodologies that compensate for uncertainty or evaluate uncertainty by, say, robustness criteria, remain to be developed.

More specifically, when reads are shared by two taxa they only mean that the two taxa are similar in the regions that were sampled. Strictly speaking this does not tell us anything about the co-existence of the two taxa. However, first, if shared reads from two taxa are found we have ruled out the case that neither taxon is present. Moreover, if we have reads present that are unique to one taxon then we have established its presence. Finally, if we have reads that are unique to the other taxon then we have established its presence. We dropped reads that are unique to one taxon they were the singleton vertices in the connectivity
graphs so we could focus on higher order terms that represented two reads.

We comment that in our wound microbiome example, unlike the HMP example, our analysis may have failed to capture all of the information in the available data as the OTU table was converted from counts (i.e., each cell gives the number of reads that align to a given OTU in a given sample) to binary (i.e., each cell indicates if any reads align to a given OTU in a given sample) prior to analysis. An extension of our method to raw data consisting of counts, e.g., with each subject \( S_j, 1 \leq j \leq b \), we associate a categorical random variable \( X_j \) that takes value \( k \) if \( k \) sampled reads align to OTU \( i, 1 \leq i \leq k \) and takes value 0 otherwise, is a topic for future research.

On the other hand our method extends to comparing two populations on the basis of their connectivity. The difference in dependencies can be regarded as indicators of which associations are present or absent in the normal case (say) versus the diseased case. This amounts to looking at the different structure of the graphs and interpreting what the cliques mean in terms of reads. Our treatment in Section 5 was subject-by-subject. In Section 5 of the Supplementary Material (Dobra et al., 2018) we compare two collections of metagenomic samples from two populations, healers and non-healers.

Our inference of the significant presence/absence of bacterial taxa, possibly unknown, is based on posterior estimates of probabilities (2.1) and (2.2). We refer to these as existence probabilities; however, this terminology belies the subtleties regarding their interpretation. These probabilities are estimated from the model averaged joint posterior distribution, and hence are conditional on clique loglinear models with high posterior probabilities, i.e., an Occam’s window approach. If a bacterial taxon (say, genus) to which reads uniquely align does not appear in any of these models, these reads will impact our probability estimates. For example, the estimate of the probability of an unidentified taxon will be inflated, while the estimate of the probability of presence of a taxon appearing in the model average will be deflated. The extent of this impact may be small, as any taxon to which many reads align should appear in the model average, but this cannot be guaranteed and warrants further study.

Stated in another way, the category of genomes we have called unidentified may be only an artifact of the modeling. Indeed, if the genome list contains all the genomes in the sample, the probability of an unidentified genome is simply a residual reflecting the short reads that do not align in sufficiently large numbers to any genome in the models in the model average. On the other hand, if the genome list is incomplete, the probability of an unidentified genome is the sum of two parts: the probability of a genome we know but
that was not included amongst the $B$ reference genomes plus the probability of something that we have not encountered before.

SUPPLEMENTARY MATERIAL

Supplement A: Additional proofs, maps, figures and tables (doi: COMPLETED BY THE TYPESETTER; .pdf). In this online supplementary material, we describe the data that were used. We also present the computational experiments performed, the details of the simulations, and further details on the software that was developed in this article.

References.


of Informatics, University of Verona.


---

**Department of Statistics**, University of Washington, USA
Box 354322, Seattle, WA 98195
E-mail: adobra@uw.edu

**School of Computing and Information Sciences**, Florida International University, USA
Box 354322, Miami, FL 33199
E-mail: hobbes182@gmail.com

**Department of Dermatology and Cutaneous Surgery, and Department of Microbiology and Immunology**, Miller School of Medicine, University of Miami, USA
RMSB Room 2089, 1600 NW 10th Ave.
MIAMI, FL 33136
E-mail: d.ajdic@med.miami.edu

**Department of Statistics**, University of Nebraska-Lincoln, USA
340 HARDIN HALL NORTH WING
LINCOLN, NE 68583
E-mail: bclarke3@unl.edu

**Department of Statistics, and Department of Food Science and Technology**, 340 HARDIN HALL NORTH WING
LINCOLN, NE 68583
E-mail: jclarke3@unl.edu
1. **Additional Tables.** Table 1 gives the degree of each vertex in the raw connectivity graph for the genera data.

2. **Data preparation.** The bacterial genomes data used in our experiments were obtained from the Human Microbiome Project (HMP) (NIH HMP Working Group et al., 2009), and the Joint Genome Institute’s (JGI) Integrated Microbial Genomes (IMG) database (Markowitz et al., 2014), version 4.0.

2.1. **Bacterial genomes.** A number of 456,865 genomic bacterial reference sequences, in FASTA format, were procured from the Integrated Microbial Genomes and Metagenomes (IMG, version 4.0) database (Markowitz et al., 2014). The 456,865 reference sequences accounted for 5,168 bacterial genomes (at strain level) which included sequences from bacterial genomes and bacterial plasmids. The 5,168 genomic references were isolated by relying on bacterial taxon names and identifiers obtained from the Genome Browser at the IMG website:

https://img.jgi.doe.gov/cgi-bin/w/main.cgi?\section=TaxonList&page=taxonomyAlpha&domain=Bacteria

2.2. **Metagenomic samples.** A human metagenomic sample was obtained from the Human Microbiome Project (NIH HMP Working Group et al., 2009). The human metagenome sample SRS015072, obtained from a female participant of the HMP Core Microbiome Sampling Protocol A (HMP-A) dbGaP study, was downloaded from the HMP FTP site. The sample consisted of 495,256 paired-end, 100 bp reads (with an average mate-distance of 81bp) in Illumina FASTQ format (Cock et al., 2010).

*A. Dobra and C. Valdes contributed equally to this work, and are joint first authors*
Table 1
Degree of each vertex in the raw connectivity graph for the genera data, expressed as a count and as a proportion of the total number of vertices.

<table>
<thead>
<tr>
<th>Degree</th>
<th>Genera</th>
</tr>
</thead>
<tbody>
<tr>
<td>55 (57.89%)</td>
<td>G43 - Lactobacillus</td>
</tr>
<tr>
<td>41 (43.16%)</td>
<td>G68 - Pseudomonas</td>
</tr>
<tr>
<td>35 (36.84%)</td>
<td>G1 - Acinetobacter</td>
</tr>
<tr>
<td>27 (28.42%)</td>
<td>G55 - Mycobacterium</td>
</tr>
<tr>
<td>23 (24.21%)</td>
<td>G41 - Herbaspirillum</td>
</tr>
<tr>
<td>20 (21.05%)</td>
<td>G40 - Gardnerella</td>
</tr>
<tr>
<td>18 (18.95%)</td>
<td>G37 - Flavobacterium</td>
</tr>
<tr>
<td>16 (16.84%)</td>
<td>G19 - Candidatus, G50 - Methylobacterium, G76 - Sphingopyxis</td>
</tr>
<tr>
<td>15 (15.79%)</td>
<td>G15 - Burkholderia</td>
</tr>
<tr>
<td>14 (14.74%)</td>
<td>G79 - Stenotrophomonas, G89 - Verminephrobacter</td>
</tr>
<tr>
<td>13 (13.68%)</td>
<td>G23 - Cupriavidus</td>
</tr>
<tr>
<td>12 (12.63%)</td>
<td>G28 - Desulfotomaculum, G45 - Lewinella, G88 - Ureaplasma</td>
</tr>
<tr>
<td>11 (11.58%)</td>
<td>G80 - Streptococcus</td>
</tr>
<tr>
<td>9 (9.47%)</td>
<td>G14 - Bryantella, G20 - Carnobacterium, G54 - Mollicutes, G63 - Polaromonas</td>
</tr>
<tr>
<td>8 (8.42%)</td>
<td>G38 - Francisella, G47 - Maritimibacter, G65 - Prevotella</td>
</tr>
<tr>
<td>7 (7.37%)</td>
<td>G17 - Caldicellulosiruptor, G48 - Melissococcus, G51 - Methylobacterium, G56 - Mycoplasma, G60 - Ochrobactrum</td>
</tr>
<tr>
<td>6 (6.32%)</td>
<td>G4 - Agrobacterium, G5 - Allobaculum, G22 - Corynebacterium, G31 - Enterobacter, G39 - Frankia, G73 - Shewanella</td>
</tr>
<tr>
<td>4 (4.21%)</td>
<td>G7 - Bacillus, G11 - Blautia, G12 - Bradyrhizobiaceae, G35 - Escherichia, G42 - Lachnospiraceae, G61 - Odoribacter, G72 - Saccharomonospora, G77 - Spirochaeta, G82 - Sutterella, G94 - alpha</td>
</tr>
<tr>
<td>3 (3.16%)</td>
<td>G2 - Actinomyces, G9 - Bartonella, G13 - Brucella, G27 - Desulfarculus, G30 - Eggerthella, G34 - Erysipelotrichaceae, G44 - Legionella, G57 - Neisseria, G64 - Porphyromonas, G71 - Rickettsia, G83 - Syntrophobacter, G90 - Vibrio</td>
</tr>
<tr>
<td>2 (2.11%)</td>
<td>G3 - Aggregatibacter, G18 - Campylobacter, G26 - Dermacoccus, G62 - Pelotomaculum, G78 - Staphylococcus, G86 - Treponema, G91 - Weissella</td>
</tr>
<tr>
<td>0 (0%)</td>
<td>G6 - Amycolatopsis, G16 - Butyrivibrio, G21 - Chlorobium, G29 - Dethiosulfovibrio, G33 - Erwinia, G52 - Microbispora, G53 - Micromonospora, G59 - OP11, G66 - Propionibacterium, G69 - Pseudonocardia, G70 - Rhodococcus, G81 - Streptomyces, G87 - Uncultured, G93 - Zymomonas, G95 - gamma</td>
</tr>
</tbody>
</table>
2.3. Reference sequence alignment. The HMP dataset (SRS015072) was aligned to the 456,865 bacterial references using the Bowtie2 (Langmead and Salzberg, 2012) aligner. Bowtie2 requires that the reference sequences be indexed so that the reads can be efficiently aligned. The bacterial genomic references were prepared for alignment using the bowtie2-build indexer program. The indexer program was run with default values along with the “-f” flag (FASTA sequences). Once the reference sequence index had been built, Bowtie2 used in local-alignment mode, was used to align the HMP data using the following command:

```bash
bowtie2 --local -D 20 -R 3 -N 0 -L 20 -i S,1,0.50 --time -f -x -S
```

Samtools (Li et al., 2009) (version 0.1.18) was then used to parse the alignments.

3. Some methodological and computational details. There are a variety of methodological and computational questions that arise in complex, big data analyses such as the ones we present in this article. This section is intended to address several of the most important ones.

3.1. From loglinear models to clique loglinear models. A loglinear model arises from using an ANOVA model for the log expected frequencies in a contingency table. As it is well known, the number of terms in a complete ANOVA model is exponential in the number of factors. So, to be useful, we must have a way to reduce the number of models considered. One way to do this is to restrict the class of loglinear models. The nested subclasses are shown in Figure 1. The first restriction we impose is that the loglinear models be hierarchical. A loglinear model is hierarchical if and only if the presence of a higher-order interaction term requires the presence of any or all of its lower-order interaction terms (Bishop et al., 2007). The model may not be identifiable without imposing some constraints on the interaction terms (Agresti, 1990).

The second restriction is to hierarchical loglinear models that are also graphical. The idea is that an undirected graph $G$, with vertices $\mathcal{B}$ and edges $E$, can be associated to any hierarchical loglinear model. These graphs are usually called independence graphs (Whittaker, 1990). Specifically, an edge $e = (b_1, b_2)$ appears in $G$ if and only if the variables $X_{b_1}$ and $X_{b_2}$ appear together in an interaction term. A hierarchical loglinear model is graphical if and only if the subsets of $\mathcal{B}$ that are the vertices of the complete subgraphs of $G$ that are maximal with respect to inclusion, are also maximal interaction terms in the loglinear model (Whittaker, 1990). If a model is graphical, the absence of an edge $e = (b_1, b_2)$ in $G$ is equivalent with the
conditional independence of variables $X_{b_1}$ and $X_{b_2}$ given the rest of the variables under the joint distribution for $X_B$. Moreover, if there is no path in $G$ from vertex $b_1$ to vertex $b_2$, then $b_1$ and $b_2$ are in two distinct and fully connected components of $G$, and the corresponding variables $X_{b_1}$ and $X_{b_2}$ are independent.

The usual notation for this is to abbreviate the components in $X$ to their indices $b \in B$, and indicate loglinear models in terms of sets of indices that show the clique structure of $G$. For example, consider the loglinear models $M_1$ and $M_2$ indicated by their generators, i.e., generated by the noted collections of sets of indices, $C(M_1) = \{(1, 2, 3), \{1, 3, 4\}\}$ and $C(M_2) = \{(1, 2), \{2, 3\}, \{1, 3\}, \{3, 4\}, \{1, 4\}\}$. The graphs associated with $M_1$ and $M_2$ are identical, and both have edges as given in $C(M_2)$. However, the graph $G$ for $M_1$ is generated by two cliques $\{1, 2, 3\}$ and $\{1, 3, 4\}$ that include the third order interaction terms for $(1, 2, 3)$ and $(1, 3, 4)$, whereas the list of edges $M_2$ forms a graph $G'$ that does not include the third order interaction terms for $\{1, 2, 3\}$ and $\{1, 3, 4\}$. Indeed, even if the edges are regarded as cliques of size two, the third order interaction terms necessary for a graphical model as indicated by the cliques in its graph are not included. Thus $M_1$ is a graphical model, but $M_2$ is not a graphical model.

The third restriction is to graphical loglinear models that are decomposable (Lauritzen, 1996). This idea is that a graphical model is decomposable
Figure 2. Graphical loglinear models $M_1$ (left) and $M_3$ (right). $M_1$ is decomposable while $M_3$ is not.

if its graph can be broken into components that are cliques without losing any information. For example, Figure 2 shows the graphical loglinear model $M_1$ and a second model $M_3$. $M_1$ is generated by $C(M_1) = \{\{1, 2, 3\}, \{1, 3, 4\}\}$ and is decomposable; its graph has two cliques $\{1, 2, 3\}$ and $\{1, 3, 4\}$, and one separator $S_2 = \{1, 3\}$. Separators are the intersections of cliques and have important properties not discussed here, but see Lauritzen (1996). By contrast, $M_3$ is generated by $C(M_3) = \{\{1, 2\}, \{2, 3\}, \{3, 4\}, \{1, 4\}\}$ and is graphical. However, it is not decomposable. Essentially, decomposable models are graphical models for which closed form MLEs exist. In terms of their graphs, this means they are triangulated, that is their chordless cycles contain no more than three vertices. The equivalence of these two formulations is beyond the scope of this brief review; see Lauritzen (1996) for details.

One of the key advantages of decomposable graphical models is that testing the existence of MLEs is straightforward and computationally less expensive even when $B$ is large.

3.2. Acceptance rates in an Metropolis-Hastings (MH) procedure. Our stochastic search algorithm uses the mechanism of Metropolis-Hastings (MH) (Madigan and York, 1995). The MH acceptance rates will primarily be a function of the richness of the model space – which is largely a subjective choice – not an indicator of whether a satisfactory collection of good models has been found although the two may be related. Indeed, there is a bias-variance tradeoff on the level of model lists: a model list that is too big can give excessive variance, while a model list that is too small can give bias. Finding a model list that achieves an optimal variance bias tradeoff is an open question.

With that in mind, we note that the top four clique loglinear models for genera from Section 4 (main article) represent a very high – around 98-99% – of the posterior probability. Thus, as a verification that the Occam’s
window is not loosing too much of the posterior probability, we note that Figure 3 shows the 5-th through 29-th individual posterior probabilities of the various genera. The labeling on the y-axis is just a coding we used for the genera and is not of concern. What is of concern is that with the 1133 models, the top 4 individual probabilities have something like 99% of the posterior probability and the probability of something unknown is around .08. This is also discussed in Section 6 (main article). Figure 3 simply shows that including more models in the Occam’s window BMA would not make much difference.

Thus, although we have not looked at acceptance probabilities, our methodology has generated a collection of models that captures the vast majority of the posterior probability so the Occam’s windowing is not losing too much information. Otherwise put, it seems as though the 1133 models resulting from our search procedure provides a reasonable approximation to the actual posterior. In our view acceptance probabilities are only interesting if one believes the model list is physically meaningful and issues of sparsity are not relevant. Indeed, sparsity will tend to force bigger jumps (when they occur) leading to an erratic pattern of acceptance rates where non-sparsity will tend to be continuous. The relationship between sparsity and acceptance rates – while interesting – has not been well investigated as far as we
know and is beyond our present scope.

3.3. **Skewed data and the BIC.** The BIC is not an approximation to the posterior: it is an approximation to the mode of the posterior and the location of the mode indicates a good model (and arguably a good parameter value). The detailed behavior of the BIC (or any information based model selection criterion) is a general problem that will not likely be resolved in our lifetimes and we are not sure how to verify conclusively that the BIC convergence our method needs to hold since the true models are unknown (and quite possibly unknowable given the dynamic nature of organisms).

Here, then, is the state of play: the BIC converges for most well-behaved distributions with sample sizes not too different from those required for analogous convergences for the CLT. After all, posterior convergence (when conditioning on all the data) is much at one with the frequentist CLT. So, as a generality, for some linear regression type models that are similar to loglinear models you want around 30 data points/parameter to be assured of good convergence. This assumes the models on the model list are not too similar. We have a high dimensional contingency table with $2^\#(\text{genomes})$ where $\#(\text{genomes})$ is in the thousands. Thus, in our first real world example (Section 4 of the main article) we used 95 genera. As noted there were 377 cells with strictly positive counts. Loosely, therefore, we have 377 sets of parameters (the cardinality of the sets of parameters being due to the hierarchical structure of the loglinear models). In terms of cell counts, the largest was 332k and the second largest was 11.6k and there are a lot that are very small – see Figure 4. Convergence will be determined by the total cell count which in turn is largely determined by the depth of sequencing.

It is obvious that the positive cell counts are unbalanced and strongly skewed; this is typical for high dimensional contingency tables and indicates sparsity. It is also well known that loglinear models might not be able to capture the unbalancedness well because the expected cell counts under a loglinear model must be strictly positive for all the cells including the ones with an observed count of zero. Since the grand total of the table of expected counts must equal the grand total of the observed table, the total sum of counts gets redistributed to the many cells with an observed count of zero. Hence what you see is that the expected counts for the cells with the largest observed counts are much smaller than those observed counts.

As such, loglinear models might fail to properly capture the imbalance of the cell counts. However, capturing the magnitude of the cell counts is not our end goal. Instead, our target is the determination of the interaction structure that exists among the variables in the data. Thus, the sharp
skewness in Figure 4 has nothing to do with the BIC per se, i.e., with the way we select loglinear models. Informally, the $n$ in the BIC is the grand total of the observed table. Since the sum of the counts is in the hundreds of thousands, we hope that even given the unbalancedness of the cell counts, the BIC will perform well. Schwarz (1978) established an optimality property of the BIC in terms of hypothesis testing, i.e., decision making, so it is not at all clear that there is a better choice for a model selection principle. Moreover, to give an indication of the scale of how well BIC works, Lv and Liu (2014) examine the use of the BIC in a model mis-specification setting involving logistic regression, and show the BIC works reasonably well even when $n = 200$ and $p = 1000$.

3.4. Robustness and sparsity. Statistical analyses of the kind we are developing here have the right sort of robustness to be credible. In our view, the use of the BMA increases robustness and the restriction to clique loglinear provides the required robustness. The lack of robustness is due to sparsity and this is typical with sparse methods.

To see what happens we did simulations. In the connectivity matrix for the simulation example from Section 3 (main article), we changed 0s to 1s using a Bernoulli with $p = .01, .05, .1$, and looked at the downstream effects on the models and the resulting graphs from the BMAs. We did this for 1000 reads total, as in the original simulation example, and for 5000 reads total. Figure 5 shows the performance of our stochastic search algorithm. As can be seen, as the number of 1s increases, the steps smooth out. This holds for 1000

Figure 4. Histogram of the positive cell counts for the HMP data.
Figure 5. The upper left panel shows the BIC of the best models identified in 100 runs of our stochastic search method. The other three panels correspond to adding 1’s to the connectivity matrix using Bernoulli’s with $p = .01$, top right; $p = .05$, lower left, and $p = .1$ lower right.

total reads and 5000 total reads. This suggests that as sparsity decreases a single model becomes more and more reasonable. We suspect that this is an artifact of simply having more reads and hence more interaction terms. In the limit, all possible interactions terms will be present and this is not helpful.

3.5. Comparisons with Bayesian tensor factorization methods. We have not included additional empirical comparisons with Bayesian tensor factorization methods for two reasons. First, the code implementing PARAFAC methods, e.g. the Matlab implementation available at https://github.com/david-dunson/WASP/tree/master/parafac/code is not usable in its current form. The code does not have any comments explaining use of the functions, and has hard coded dimensions whose meaning we could not understand. We have tried using this code, and have been unsuccessful in our attempt. Second, as we point out in the text of the paper, the advantage of clique loglinear models over PARAFAC type methods is that clique loglinear models are easily interpretable, while PARAFAC models are not easily interpretable: they do not embed interactions in the usual way, and for this reason the higher order interactions they correspond to need to be inferred from the posterior samples. This is not a straightforward
process for an applied researcher who needs to understand the models they are using in their analysis.

4. Simulations. To verify the performance of the proposed method, a synthetic experiment was created with a known bacterial independence structure. This constitutes our ground truth. A number of 2,273 bacterial genomes from the National Center for Biotechnology Information (NCBI) GenBank database were obtained from the complete set of genomes. These complete genomes are considered to be very high quality by GenBank, and are deemed to have a final DNA sequence for their respective genomic sequences (chromosomes and/or plasmids).

![Input matrix with a predefined independence structure among the genomes (columns) based on their sequencing reads (rows) and column totals (last row).](image)

A binary matrix that dictates the reads (rows) that are shared by each genome (columns) was generated. Figure 6 illustrates such a matrix. In this matrix, rows represent sequencing reads, columns represent known bacterial genomes, and cells contain a 1 when a read maps to a given genome, and 0 otherwise. The last row of the matrix contains the number of reads that match a given genome, and we use this total match count as the starting point to create a independence among a set of genomes.

A simulation script was developed in Python (version 2.7.10) to facilitate the creation of random binary matrices. The script takes as input the total number of genomes to include in the experiment, the maximum clique size,
the total number of reads, and a percentage of the genomes to leave out of
the connections.

The script will then randomly select the genomes to make connections
based on shared reads and construct cliques of size two to the maximum
number specified in the parameters. Genomes are essentially binary vectors
where each location is a read, and a 1 or 0 at that location marks a read as
mapping to the genome. Creating linked genomes is a matter of establishing
binary vectors that share the same amount of reads at similar positions.
Genomes with identical sequences will have identical read catalogs repre-
sented by identical binary vectors. As output the simulation script creates
the binary matrix that represents the known independence graph and is used
as input to the software – see Section 6.

5. Wound microbiome. The wound microbiome experiment investigated
the bacterial population involved in chronic wound healing in elderly
diabetic patients. The microbial community in the wound bed, wound edge,
and peripheral healthy skin was compared at two time points.

Samples were collected from the wound bed and wound edge, as well as
swab samples from the peripheral healthy skin of 10 patients. These sam-
ple were collected twice: once for the patient’s first visit before treating and
redressing the wound, and in the second visit one week later. A total of 50
samples were collected. They were sequenced using the standard Illumina
protocol for 16S rRNA gene sequencing, using the Illumina MiSeq instru-
ment. After processing by the company (Second Genome), the investigators
shared with us a table of already calculated OTU abundances. We did not
make the jump from 16S to OTUs; it was already done and we just used
the data. We have a copy of the report from the company but are not yet
authorized to release it. What we can say is that an OTU is a cluster of
reads with a similar 16S-gene sequence. Usually the identity threshold is
96-97% or higher. The basic idea is that similar bacteria will have similar
16S-gene sequences, and they can be identified by grouping them together
given a threshold: 96-97% agreement usually gives resolution at the genera
level, 98-99% usually gives approximate resolution at the species level. It is
unclear if OTUs are useful at this time for finer levels such as strains. Thus,
for present purposes, it is enough to observe that the 16S rRNA sequencing
data was converted to operational taxonomic units (OTU) and analyzed to
identify the biologically significant OTUs.

5.1. OTU analysis. The analysis goal was to identify the OTUs that
are significant between patients whose wound healed (“healer”) versus those
patients that did not heal (“non-healer”) in the context of the location of
the wound: the wound bed, wound edge, or peripheral healthy skin.

# Load data
otu <- read.csv('otuTable_counts.csv', header=T, row.names=1)
samp <- read.csv('otuTable_samples.csv', header=T, row.names=1, stringsAsFactors=FALSE)

otu.mat <- as.matrix(otu)

source("https://bioconductor.org/biocLite.R")
biocLite()
biocLite('phyloseq')
library("phyloseq")

OTU <- otu_table(otu.mat, taxa_are_rows = TRUE)

# Phyloseq
physeq<-phyloseq(OTU)
sampledata<-sample_data(samp)
physeq1<-merge_phyloseq(physeq,sampledata)

biocLite('edgeR')

# EdgeR Analysis
library(edgeR)
Diagnosis<-get_variable(physeq1, "type")
Location<-get_variable(physeq1, "location")
design<-model.matrix(~Diagnosis + Location)
x<-as(otu_table(physeq1), "matrix")+1L
x<-DGEList(counts=x, group=Diagnosis)

# Calculate norm factors and estimate dispersion
x<-calcNormFactors(x, method="RLE")
x<-estimateGLMCommonDisp(x, design)

# Model fitting
fit <- glmFit(x, design)
lrt <- glmLRT(fit)
5.2. *OTU results.* 185 OTUs were identified as being significant between “healer” and “non-healer” patients. Table 2 contains the results. Figure 7 contains the intersection between the resulting lists from Table 2.

<table>
<thead>
<tr>
<th>Wound Location</th>
<th>Number of Significant OTUs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bed</td>
<td>144</td>
</tr>
<tr>
<td>Edge</td>
<td>100</td>
</tr>
<tr>
<td>Peripheral Healthy</td>
<td>75</td>
</tr>
</tbody>
</table>

**Figure 7. Intersection between the significant OTUs from the wound bed, wound edge, and peripheral healthy skin.**

To examine the dependencies among the three wound locations, we ran the significant OTUs to create an independence graph using the proposed method. At the same time, we ran a principal component analysis (PCA) on the samples, and also performed hierarchical clustering to see if we could identify groups of samples that were linked together by their respective OTUs. Figure 8 contains the dendrogram plot for the hierarchical clustering (panel A), the PCA plot (panel B), and the independence graph (panel C).
In all figures, the patient effect is the strongest differentiating factor as samples from a given patient tend to cluster together in hierarchical clustering (panel "A") and PCA (panel "B"). The independence graph creating by the model is also in agreement with the plots as samples from a given patient, or a patient condition (healer or non-healer), are connected in a clique.

6. **Software availability.** The software for this project is open source software, available under the GNU General Public License, Version 3. The software is developed in R, version 3.2.3, and can be obtained at the following GitHub repository:

   https://github.com/camilo-v/Clique_Log_Linear

**References.**


---

**DEPARTMENT OF STATISTICS, UNIVERSITY OF WASHINGTON, Box 354322, Seattle, Washington, 98195 USA**

E-MAIL: adobra@uw.edu

**DEPARTMENT OF DERMATOLOGY AND CUTANEOUS SURGERY, AND DEPARTMENT OF MICROBIOLOGY AND IMMUNOLOGY, MILLER SCHOOL OF MEDICINE, UNIVERSITY OF MIAMI, RMSB ROOM 2089, 1600 NW 10TH AVE, MIAMI, FL 33136**

E-MAIL: d.ajdic@med.miami.edu

**DEPARTMENT OF STATISTICS, AND DEPARTMENT OF FOOD SCIENCE AND TECHNOLOGY, 340 HARDIN HALL NORTH WING, LINCOLN, NE 68583**

E-MAIL: jclarke3@unl.edu

**DEPARTMENT OF COMPUTING AND INFORMATION SCIENCES, FLORIDA INTERNATIONAL UNIVERSITY, 11200 SW 8TH STREET, ECS 354, MIAMI, FL 33199**

E-MAIL: hobbes182@gmail.com

**SCHOOL OF COMPUTING AND INFORMATION SCIENCES, DEPARTMENT OF STATISTICS, UNIVERSITY OF NEBRASKA-LINCOLN, 340 HARDIN HALL NORTH WING, LINCOLN, NE 68583**

E-MAIL: bclarke3@unl.edu