Using a co-occurrance network

Jim Brunner

June 20, 2017

1 Network Building

We are looking at creating ecological networks of a microbiome. Right now, I have built two networks, in the form of graphs $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1)$ and $\mathcal{G}_2 = (\mathcal{V}_2, \mathcal{E}_2)$. In both graphs, vertex are labeled by taxa name. I'm going to conflate the vertices and their label. The edge sets are defined by co-incidence and co-occurence, respectively. Given a set of samples with abundancs of organisms, we map the abundances to discrete levels, as proportions of the maximum abundance *of that organism.* Precisely, let the samples be s_i and raw abundance of organism j in sample i be r_{ji}. I map the abundances according to

$$a(r_{ji}) = \begin{cases} \lfloor \left(\frac{r_{ji}}{\max_{s_k}(r_{jk})}\right) n \rfloor + 1 & \frac{r_{ji}}{\max_{s_k}(r_{jk})} \geq m \\ 0 & \frac{r_{ji}}{\max_{s_k}(r_{jk})} < m \end{cases}$$

where m is some minimum. Then, I create weighted edges between vertices (where 0 weight means no edge exists) where the weights of edges in $\mathcal{E}_1$ are

$$w_{jk}^1 = \frac{\|\{i : a(r_{ji}) = a(r_{ki}) \neq 0\}\|}{S}$$

where S is the total number of samples. That is, we count the propotion of samples in which the two taxa appear at the same discritized level.

The second network accounts for random coincidence of taxa in a sample, following [2]. It begins with $\mathcal{G}_1$, and compares to a null model N. The null model is defined in the following way.

$$A_j = \sum_{s_i} \mathbf{1}_{a(r_{ji}) \neq 0}$$

and

$$S_i^l = \sum_{v_j} \mathbf{1}_{a(r_{ji}) = l}$$

Then, N assumes that if

$$X_{jil} \sim binom\left(A_j, \frac{S_i^l}{\sum_{il} S_i^l}\right)$$

then $P(a_{ji}^N = l) = 1 - P(X_{jil} = 0)$. This allows us to calculate the probability of co-incidence of taxa under the null model. Let w_{jk}^N be

$$w_{jk}^N = \|\{i : a_{ji}^N = a_{ki}^N \neq 0\}\|$$

This is the similar to the co-incidence model but now randomized. Then,

$$P(w_{jk}^N = K) = \sum_{\{A \subset \mathcal{V}_2 : |A| = k\}} \prod_{l \in A} a_{jl} a_{kl} \prod_{l \notin A} (1 - a_{jl} a_{kl})$$

Ideally, we would then define $\mathcal{E}_2$ by the weights

$$w_{jk}^2 = \begin{cases} 1 & P(w_{jk}^N \geq w_{jk}^1) \geq t \\ 0 & P(w_{jk}^N < w_{jk}^1) > t \end{cases}$$

However, that probability is intractible to compute. Instead, we take

$$\widetilde{P}(w_{jk}^N = K) = \sum_{l=0}^{i} \binom{N_1}{l}\binom{N_2}{K-l} p_1^j p_2^{K-l}(1-p_1)^{N-l}(1-p_2)^{N-K+l}$$

where

$$p_1 = p_a - \left(\frac{N_2}{N_1}\frac{N(\mu-\sigma^2)-\mu^2}{N^2}\right)^{1/2}$$

and

$$p_2 = p_a - \left(\frac{N_1}{N_2}\frac{N(\mu-\sigma^2)-\mu^2}{N^2}\right)^{1/2}$$

Finally, N_1, N_2 are to ensure that $p_1, p_2 \in [0,1]$. It turns out we need:

$$\frac{\mu N(1-p_a) - N\sigma^2}{N(1-p_a)-\sigma^2} \leq N_2 \leq \frac{\mu^2}{\mu - \sigma^2}$$

with μ the mean of the real distrubution, σ^2 the variance, and $p_a = \frac{1}{S}\sum_i a_{ji}a_{ki}$. So, we take

$$w_{jk}^2 = \begin{cases} 1 & \widetilde{P}(w_{jk}^N \geq w_{jk}^1) \geq t \\ 0 & \widetilde{P}(w_{jk}^N < w_{jk}^1) > t \end{cases}$$

2 Network Analysis

We can cluster - community clustering, spectral clustering, comparing to random graphs. Comparison of clusters to grouping by sample type.

3 Using the network to analyze a sample

The question now is what can we do these networks?

First, assesing GOTTCHA reads. A single GOTTCHA read would produce a network with each connected component complete. I think the co-incidence network is more appropriate. We want to assess the probability that you see this set together. We have the probability that you see any vertex pair (estimated) as the edge weights of $\mathcal{G}_1$. Precisely, the edge weights are

$$w_{jk}^1 = P(j \,\&\, k \in S_i^l)$$

where S_i^l is sample i at discrete abundance level l. It might be useful to have the directed weight graph where

$$w_{jk}^3 = P(j \in S_i^l | k \in S_i^l)$$

but that wouldn't be hard, because then

$$w_{jk}^3 = S\frac{w_{jk}^1}{\|\{i : a(r_{ki}) > 0\}\|}$$

Anyway, let's start with the simplest case of one abundance level. Assume GOTTCHA found taxa $a, b, c, ..., n$. Maybe the first thing we would want is

$$P(a|b,c,d,...,n),\ P(b|a,c,d,...,n),\ etc$$

Clearly, we can see directly $P(a|b)$, etc. We can also get a bound for triplets (assuming $P(c,b) \neq 0$):

$$P(a|b,c) \leq \frac{\min_{(i,j)\subset\{a,b,c\}}(P(i,j))}{P(b,c)}$$

but we can't do any better than triplets explicitly, because we don't have any sort of independence (conditional or otherwise) and because our network is not acyclic.

We can probably learn something from asking about the connectivity of the induced subgraph. Notice that if it isn't complete, then one of the $P(a,b)$ is 0 above.

What does the connectivity of the induced subgraph of $\mathcal{G}_2$ tell us? If it is connected, that's good. We could use that network to identify vertices that are connected to many of the vertices in the induced subgraph - this might indicate that node should be in the sample.

I guess $\mathcal{G}_2$ is a markov random field [1]. This might give us a way to calculate the probability you see a group taxa (and maybe others). The main idea of a MRF is that nodes are conditionally independent of nodes they aren't neighbors of (conditioned on ones they are neighbors of). If c are the (maximal) cliques of the graph (complete subgraphs), then the probability of configuration $\boldsymbol{x}$ is

$$P(\boldsymbol{x}) = \frac{1}{Z}\prod_{c} \psi_c(x_c)$$

where Z is a normalizing constant and ψ_c are potential functions I don't know how to come up with yet. Anyway, if we have a sample that contains (maybe as a subset) s, we can calculate something. Let C_s be the cliques represented in s.

$$P(\boldsymbol{s}) = \frac{1}{Z}\sum_{\{\boldsymbol{x}:s\subset\boldsymbol{x}\}}\prod_{c\in C_s}\psi_c(\boldsymbol{x})$$

And we can ignore cliques not represented in s. We probably have to change Z. I guess we can also use neighbor pairs instead of maximal cliques. Either way we have to figure out what ψ_c are.

3.1 Determining ψ_c

Before figuring out ψ_c, it should be noted that we have a choice of configuration space of the network. We can use a binary $\{1,0\}^N$ space to denote presence or absence, or we can choose a continuous space to include abundances. To begin, I will only consider presence & absence.

Snowshoe hairs and Canadian Lynx, CRNT

Now I'm very tempted to think about how this approach relates to mass action dynamical systems. I'm going to think about my very favorite model.

$$\dot{\boldsymbol{x}} = \kappa_1 x \begin{pmatrix}1\\0\end{pmatrix} + \kappa_2 xy \begin{pmatrix}-1\\1\end{pmatrix} + \kappa_3 y \begin{pmatrix}0\\-1\end{pmatrix}$$

Well. Yes. But there is only one maximal clique so $P(\boldsymbol{x}) = \psi(\boldsymbol{x})$ and that's just whatever it is. We know that it evolves according to the stochastic mass action equations. In fact, the result that the stationary distribution is a product of poissons for a complex balanced system can be thought of as fitting into this framework. I wonder if you could use this theory to re-prove that? Recall that the stationary distribution for a complex balanced system with equilibrium $\boldsymbol{c}$ is

$$\pi(\boldsymbol{x}) = \frac{1}{Z}\prod_{i=1}^{d}\frac{c_i^{x_i}}{x_i!}$$

so here we have that

$$\psi_c = \prod_{i\in c}\frac{c_i^{x_i}}{x_i!}$$

or taking cliques to be just singletons

$$\psi_i = \frac{c_i^{x_i}}{x_i!}$$

I can't think of any way to arrive upon that directly, and so prove the result from this direction.

It seems reasonable to use a pairwise RMF. It also seems reasonable to take a log linear function. My initial guess is

$$\psi_{s\sim t}(y_s, y_t) = \exp(\boldsymbol{\theta}\cdot\mathbf{1}_{y_s=i,y_t=j})$$

where $\theta = (0, \log(P(s)), \log(P(t)), \log({}^1\!/\!{}_2(P(s|t) + P(t|s))))$. But that's a guess. It makes sense we have only unconnected nodes. It also also penalizes leaving out nodes that are connected to the nodes we do have.

References

[1] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective.* The MIT Press, 2012.

[2] Heiko Mller and Francesco Mancuso. Identification and analysis of co-occurrence networks with netcutter. *PLOS ONE*, 3(9):1–16, 09 2008.