

## Manuscript Details

<b>Manuscript number</b>	FUNECO_2018_132_R1
<b>Title</b>	Spatial patterns of fungal endophytes in a subtropical montane rainforest of northern Taiwan
<b>Article type</b>	Research Paper

### Abstract

Fungal endophytes of plants are ubiquitous and important to host plant health. Wood-inhabiting and foliar endophyte communities from multiple tree hosts were sampled at multiple spatial scales across Fushan forest dynamics plot in northern Taiwan, using culture-free, community DNA amplicon sequencing methods. Fungal endophyte communities were distinct between leaves and wood, but the mycobiomes were highly variable across and within tree species. Despite this, host tree species was an important predictor of mycobiome community-composition. Within a single common tree species, “core” mycobiomes were characterized using co-occurrence analysis. The spatial cooccurrence patterns of these few species of fungal endophytes appear to explain the strong host effect. For wood endophytes, a consistent core mycobiome coexisted with the host across the extent of the study. For leaf endophytes, the core fungi resembled a more dynamic, “gradient” model of the core microbiome, changing across the topography and distance of the study.

**Keywords** Fungal Endophytes, Microbial ecology, ITS, Plant-fungal interactions, Fushan Subtropical Forest Dynamics Plot, Mycobiome, Core microbiome, PCNM/dbMEM spatial analysis

**Corresponding Author** Daniel Thomas

**Order of Authors** Daniel Thomas, B. A. Roy, Huei-Mei Hsieh, Roo Vandegrift, Yu-Ming Ju

**Suggested reviewers** Anthony Amend, Takashi Osono, Derek Persoh, A. Elizabeth Arnold

## Submission Files Included in this PDF

### File Name [File Type]

dthomas\_coverletter.docx [Cover Letter]

reviewer\_responses.docx [Response to Reviewers]

DanThomas\_revised\_body.docx [Manuscript File]

Fig1.eps [Figure]

Fig2.tif [Figure]

Fig3.eps [Figure]

Fig4.eps [Figure]

Fig5.eps [Figure]

Fig6.eps [Figure]

Fig7.eps [Figure]

Fig8.eps [Figure]

SuppFig1.eps [Figure]

SuppFig2.eps [Figure]

SuppFig3.eps [Figure]

SuppFig4.eps [Figure]

Statistical\_notebook.pdf [e-Component]

Bioinformatic\_notebook.pdf [e-Component]

## Submission Files Not Included in this PDF

### File Name [File Type]

Table1.xlsx [Table]

Table2.xlsx [Table]

To view all the submission files, including those not included in the PDF, click on the manuscript title on your EVISE Homepage, then click 'Download zip file'.

## Research Data Related to this Submission

### Data set <http://dx.doi.org/10.17632/czvrfcj377.2>

Wood endophyte sequence library for: "Rolling a mycobiome down a hill: endophytes in the Taiwanese Cloud Forest"

This is the original wood endophyte library. The title of the manuscript has been changed, but sequence data is unchanged.

### Data set <https://doi.org/10.6084/m9.figshare.3208252.v1>

Data from: Spatial ecology in the Xylariaceae: combining traditional collection and next-generation sequence-based microbial survey techniques.

Leaf read library. These are the original Illumina MiSeq and other datafiles associated with the paper: Spatial ecology in the Xylariaceae: combining traditional collection and next-generation sequence-based microbial survey techniques.

December 7, 2017

To the editors and staff of Fungal Ecology,

I submit to you our manuscript, “Rolling a mycobiome down a hill: endophytes in the Taiwanese Cloud Forest.”

It is a biogeographical study of the mycobiome of aerial tissues (leaves and wood) of various tree species, in the typhoon-battered hills of northern Taiwan. It also examines the concept of a “core” microbiome. I entered into the project expecting to debunk concepts of the “core” mycobiome in natural settings. Instead, I finished the project believing that we may have identified a cadre of core fungi for one host tree species, with some interesting spatial patterns. The environmental data available explained very little of the patterns we observed, leaving us to examine topography and dispersal limitations as possible drivers of community assembly of endophytes, though of course further work is needed to confirm this.

I hope you enjoy. Please contact me with any questions.

Best,  
Dan Thomas  
Institute of Ecology and Evolution  
University of Oregon  
[dthomas@uoregon.edu](mailto:dthomas@uoregon.edu)

**Reviewer #1: I was excited to read the MS by Thomas and colleagues, which offers novel insights and scales of consideration for fungal foliar and wood endophyte communities. The most novel contribution of the paper, in my opinion, is the detection of landscape-scale variation of community dispersion. This is a system and topic of interest to Fungal Ecology's readership.**

**I have a few suggestions for how the paper might be edited for clarity, and a couple of statistical points that the authors may wish to consider in a revision: mostly the potentially confounding factors of topography, distance and host identity. I also feel that the authors spent much of the paper reporting their negative results from the whole community analyses. Negative results are fine, but with 13 figures, I suggest condensing these to the most critical and moving most to supplements.**

Removed.

**I was unable to access the source code for this paper, and I did not see where/if the data were deposited to the SRA or another sequence repository.**

The information for the read libraries was supplied in the data statement, the link to which is the final page in the submission pdf. The reads are deposited in two different repositories:

1) as suggested by Elsevier, the wood reads are stored with Mendelley data. However, we double checked, and found a typo in the wood reads doi. The doi listed in the data statement for the wood endophyte reads is <<http://dx.doi.org/10.17632/czvrfcj377.3>>. This is **incorrect**. The actual doi for the wood endophyte reads is:

<<http://dx.doi.org/10.17632/czvrfcj377.2>>

This will be corrected.

2) Leaf reads are held by figshare, listed correctly at the time in the data statement. The doi again is:

<https://doi.org/10.6084/m9.figshare.3208252.v1>

**In the future please submit your paper with line numbers to facilitate reviews.**

From the author instructions, <<https://www.elsevier.com/journals/fungal-ecology/1754-5048/guide-for-authors#4001>> which at the time of initial submission said the following:

*“Manuscripts should be double-spaced throughout. Do not include line numbering in the manuscript.”*

**First paragraph is very long. Could use a topic sentence.**

Paragraph has been shortened. The first sentence is intended to be a topic sentence.

**Field Methods: So host tree was not explicitly considered in the sampling scheme?**

Host was not explicitly considered for both statistical and logistical concerns; it was entirely unknown before sampling. We undertook too large a study in too biodiverse a site to sample in a way that could

balance hosts evenly. When ranked by abundance, tree species at the Fusan FDP follow a typical rank-abundance curve, with a small number of highly abundant tree species and a large number of relatively rare trees. Some hosts exist only in small clusters on the landscape, or are confined to unusual habitat. Attempting to balance our host numbers beyond a few abundant tree species would have resulted in an extremely contrived, clustered sampling pattern, aggravating the concerns about bias due to autocorrelation in the microbial communities.

We also needed a sufficient diversity of hosts to make our co-occurrence analyses robust. We needed enough other hosts, even if smaller numbers, to tell if a microbe was tending to co-occur with the hosts for which we had sufficient sample sizes to spatial analyses. The numerous other hosts with small sample sizes picked up in our study allowed us some extra statistical confidence to say that the core microbes we detected in *Helicia* were indeed tending to co-occur with *Helicia* more often than they seemed to be in other hosts.

### **Are the hosts spatially autocorrelated?**

As I'm sure Reviewer #1 is aware, the term autocorrelation is sufficiently broad to require further explanation of this question. Vegetative community does show strong spatial patterning, as is easily seen in fig. 1 but not quantitatively analyzed in our manuscript. A quick review of the individual tree species distributions on the plot also reveals obvious spatial patterns at several scales. Much of this patterning is discernible by the eye to be due to external ecological condition, and some excellent analyses by Su et al. support this<sup>2</sup>, creating "spatial dependence" in the tree distributions. Some of the patterning could be due to "autogenic" or "internal" autocorrelation.

We would like to know a bit more about which type or scale of autocorrelation in which Reviewer #1 is most interested, but we can imagine they would be interested a classical univariate spatial analysis of individual tree species (Moran's weof tree density, Nearest-Neighbor, kernel-density-estimate or Kriging-type analysis), or a multivariate, tree community approach using tools similar to the spatial analyses used in this paper (PCNMs and mantel tests of community dissimilarity). Further spatial analysis of spatial patterns of trees at Fusan would be a complex but very worthwhile project. It is unfortunately beyond the scope of this paper, and we believe the data from the tree census at this time are still under the supervision of Su et al. who are working on various projects with this data, though access to this data may be available upon request.

We believe we understand Reviewer 1's reasons for asking this question, however. It is important to ask whether any spatial patterns observed in the fungi are results of induced spatial dependence on their tree hosts and the habitat that these trees prefer,. The result of this would be lots of error due to autocorrelation, potentially causing false positives in our ecological analyses that are not due to the fungi but simply where the hosts tend to aggregate. To address this question we focused most of our analyses on the fungi of just one host.

Any remaining error due to autocorrelation that may have resulted from a clustered sampling scheme is then quantifiable in the spatial analyses. It is quantified in our study with the Mantel tests and the smaller scale PCNMs. In the case of multivariate Mantel tests, if assumptions are met, internal autocorrelation is often considered to be represented by the range of comparisons that are above the x-axis in the Mantel correlograms. In the case of PCNM/dbMEM analyses, the eigenvectors with smallest wavelengths are considered strong candidates for describing internal or autogenic autocorrelation, especially if they are not correlated with environmental data (Borcard et al 2011). Other remaining eigenvectors are orthogonal to these, and can be considered for the purposes of ecological analysis

independent of internal autocorrelation. Thus if we restrict our ecological observations to patterns that are larger in scale than these eigenvectors, we hope that we are not simply describing patterns due to internal autocorrelation. Both our mantel tests and our PCNM/dbMEM analyses point to ecological patterns of interest in our fungal community at a scale of at least ~150-200 m, and mantel tests indicate that positive autocorrelation effects are nearly non-existent by 150 m in both plants and wood.

**Bioinformatics: Please explain what variance stabilization is.**

We agree with Reviewer 1 that this is a complex topic that many readers may want to know more about. However, it is also now a fairly common term in microbial ecology when dealing with second-generation sequencer technologies. We're not sure that we should take the extensive space within the manuscript to explain, especially given that it is generally suggested by the reviewers that we shorten this manuscript. A brief explanation has been added, and citations are given for readers who want to know more.

**In the Bioinformatics section it's stated that the OTUs were converted to presence/absence. But then it's stated that data used BC distance index and were Hellinger transformed... Can you please clarify which data were used for which analyses**

All read abundances of an observation of a fungal OTU in a sample were simplified to presence/absence, before any other analyses were performed. So analyses that involved the community matrices of endophytes – which is **all** of the analyses reported in this manuscript - were performed using presence/absence type community matrices, not with read abundances. This was due to our finding that differences in abundances of reads did not represent “actual” abundances in our positive control (see discussion here: < <https://doi.org/10.1101/184960> >). We have added clarification in the text.

BC dissimilarity was used wherever a dissimilarity measure was required for a statistical test or for visualizations. Clarification has been added.

Hellinger transformations of community data were applied only when required by the assumption of a statistical test. This was only when applying redundancy analysis (RDA) to our community matrices indicated in the text, and not elsewhere.

These transformations are not mutually exclusive, obviously. In several tests all three were applied to the data. More clarification has been added in-text, hopefully it helps.

**-Trends in numbers of observations parallel patterns in OTU diversity (Fig. 4); if a class of fungi contained a large diversity of OTUs, it also tended to be observed often throughout the study site.**

**This sentence is awkward and nothing about "occupancy" is shown in figure 4.**

The sentence was removed.

**-For all of the PERmanova analyses, this is a measure of both dispersion and "location" so comparing categories with different sample sizes will almost always lead to significant differences because higher N samples will have greater dispersion. I suggest using a type III sum of squares (package(cars) can handle this), or check your dispersion using betadisper in vegan.**

We are aware of the possibility of confusing centroid-location and dispersion differences with PERMANOVA and other non-parametric, multivariate tests in the ANOVA family. The sample design is highly unbalanced (for reasons listed above), and dispersions were very different (see scripts, section “NMS/PERMANOVA Ordinations, Host”). As shown in the manuscript, all 2D NMS solutions estimate separate locations of centroids/dispersion in dissimilarity space among several hosts, with the most extreme example being *Cyathea*-associated leaf endophytes. In general, we believe that while effect size ( $R^2$ ) from the PERMANOVA model for host effects is probably exaggerated by differing dispersions, the p-value resulting from the pseudo-F comparisons for host-effects are not unrealistic, given the separation of groups shown in all NMS solutions.

In the initial PERMANOVAs are each single factor tests. In the case of host, host is the single factor, host species are levels. we did not nest them. we are uncertain how to apply a different type of sum of squares for the treatments or error in a “simple” ANOVA. My (admittedly poor) understanding is that alternate SS methods are generally applied when dealing with multifactorial experimental setups – is the reviewer asking that we treat sample size as a second factor in the ANOVA? we would appreciate further instructions on this. Even if we could follow this recommendation on a classical univariate ANOVA, we have not heard of alternative sum-of-squares methods developed for in the non-euclidean (Bray-Curtis) space of dissimilarity-based analyses such as PERMANOVA<sup>3</sup>. The car package has some support for alternate SS in MANOVA, but we can find no tools for PERMANOVA. The R car package is for general linear models, not for dissimilarity-based pseudo-linear models like PERMANOVA . We are not qualified to propose our own improvements to the PERMANOVA algorithms, or to make modifications to the existing code in the vegan package. This is perhaps something to bring up with Anderson, Oksanen, et al... So again, further instruction would be helpful.

Beta Dispersion was checked in the original scripts supplied with the initial manuscript, but the reviewer notes that they were not able to access to the supplementary material. In addition, even had they been available for Reviewer 1, the figures/results in these scripts were insufficient or absent in many places. This was due to some personal events that included a stolen computer and lost backups. we have made further efforts to reconstruct the original notebooks from the history of git commits. From this all original code was recovered, but some large gaps in figure outputs remain lost. As such, we rebuilt the scripts from ground up with figures. This was the main effort that took place during this round of revisions.

These rebuilt scripts include our process for observing multivariate dispersion in the microbe community dissimilarity matrix. Special detail is given to wood samples.

As with all microbiome studies, much of the art is in finding the signals in the massive noise. Here, much of the paper can be thought of as an exploration and denoising of this initial signal from host detected by the initial PERMANOVA/NMS tests. Taken alone, this signal of host-effects is, as reviewer #1 notes, not enough to write home about. Given the shortcomings of the PERMANOVA test, we dig further into the reasons for this signal, by decomposing the community matrix two into different categories of fungi, core and non-core. Plant microbiomes are incredibly, inordinately variable but when we do this binning, some order emerges. This subset of more “loyal” fungi that are identified through co-occurrence networks present themselves as the reason for the host-effects that are observed (shown in the BC comparisons in the last figure in the manuscript). They also show some interesting spatial patterning, unlike the chaotic microbial community at large.

**Better yet, add all of these components to your RDA model and calculate partitioned variance explained when all of these factors are considered together.**

A variation partitioning analysis was conducted for the environmental, host, and spatial (PCNM/dbMEM eigenvectors) matrices as explanatory variables for the fungal community matrix. This is, effectively, an RDA model (or rather several, compared). This was in the original manuscript, reviewed by reviewer 1. So further clarification from Reviewer 1 is needed for me to really answer this comment.

However, we should note here that we have decided to remove the language from the main manuscript around the variation partitioning, as it is mostly a negative (inconclusive) result, adding unnecessary complexity to the manuscript. Code and results for this analysis are still available in the first scripts.

In general the comments from reviewers have made me realize the PERMANOVA tests are over-emphasized in the manuscript – they are a minor part of the overall analysis. They were intended to show that a strong signal from host effects seems to be present, but that we must dig deeper into the spatial aspects of the sampling and behavior of “core” mycobiome to understand what may be behind this signal. As such, we have attempted to de-emphasize these test results.

**-It took me a couple of readings to understand the main point: that although community composition is not particularly spatially patterned, community dispersion (heterogeneity) is. This very interesting and novel result, unfortunately, seems to be a bit buried in the long list and figures of negative or weak results when the entire dataset was considered. I suggest highlighting this result more prominently and earlier in the paper. Perhaps also consider reporting this in more straightforward language (like heterogeneity). Couching this in terms of a "gradient" of "core" microbes requires some mental gymnastics.**

we believe that Reviewer #1 has missed the point that we've tried to make with this paper, or disagrees with it. If the former, I've tried to simplify and clarify with the revised manuscript presented. Unfortunately, we cannot discard the language and theory around the discussion of core microbiomes. The initial motivation and hypothesis for the study was the null hypothesis that core microbiomes do not exist in nature. Indeed, we began the study attempting to debunk the concept of a core microbiome and was then convinced that perhaps this is indeed a useful vocabulary for describing what we had observed. Though I've perhaps failed to convey this, the narrative is fairly simple.

I'll summarize here and we hope that the revisions to the manuscript serve to also clarify the message. In addition, we have discarded other results, especially results that concern the importance of the hilltop, that may be distracting from this central narrative.

We sampled endophytic fungi from numerous host trees. We found:

- 1) Mycobiomes are incredibly chaotic and different, even within single hosts species and on such a small scale as the boundaries of the Fushan FDP.
- 2) Nevertheless, fungal communities are differentiated somewhat by host. To discern the reasons for this host effect, we focus all following analyses on the most numerous tree species, *Helicia formosana*.
- 3) We identify a very small subset of fungi, which we call “core” fungi, that co-occur more often than expected by chance with their host, among thousands of other fungi that do not. This is the definition

that Reviewer #1 notes requiring mental gymnastics, we hope we have clarified the language around this well enough so that it no longer does.

4) We explore the patterns of these core fungi. The initial signal of host effect noted above seems well explained by the presence of these fungi. There are also interesting differences in spatial patterns between wood and leaf “core”.

**Two things you might want to address:**

**1) All of the spatial patterns in fungal communities appear to correlate with vegetation (composition, heterogeneity etc.) Given the reasonably strong host effect found, is it possible that surrounding vegetation is shaping fungal community structure even when host is held constant? Could it be that "core" fungi aren't actually specific to host, but instead to an ecotype?**

Reviewer 1 may have misinterpreted some of our efforts and results concerning this. Vegetative community and our other composite environmental variable were not really useful in predicting differences in fungal communities. We hope that the new scripts clarify things a bit, language has been clarified in the manuscript itself, and here is a verbose explanation:

Some very small amount of variance in our wood and leaf endophyte communities was explained by our environmental variables when all tree host species were considered. Though statistically significant, the effect size of the correlations mentioned above are very weak in both our original and newer analysis, ranging from  $R^2 = 0.02$  to  $0.09$ , to the point that they might be considered a negative result. It is difficult to tell whether this is due to coarse resolution of our environmental data ( $20m \times 20m$  grain), or whether these results are “real”, indicating that endophyte communities are not well predicted by environmental conditions at this spatial scale.

In addition, the environmental variables (vegetative and topographic), are highly correlated among themselves, and spatial patterns are not able to be taken into account with a test like a simple PERMANOVA. So we cannot easily interpret these small effects observed by the environmental PERMANOVA models. For this reason a partitioning of variance analysis was conducted, though it was vastly underpowered and we have removed it from the manuscript body as suggested. Code for it remains in the original scripts. Regardless, when we then focus on a single host, *Helicia formosana*, we lose all correlations between our environmental data and our fungal community.

We were unable to detect clear ecological drivers behind the endophyte community at large, but we are not surprised by this. We expect the microbiomes of trees to be complex in the mathematical sense, with many non-linear patterns, influenced at multiple scales in time and space, with chaotic behaviours in abundances and locations over time. There may simply be no order or manner of predicting patterns of most of the plant microbiome on this spatial scale and at a single point in time - virtually all spatial scales on the planet have been shown to be important to understanding microbiomes. Here we examine environmental predictors on a scale of ~ $700m$  or less, with most of our statistical power for detecting differences in microbial community lying in comparisons of communities at distances far less than this. However, the focus of this paper was on the detection and ecological description of possible “core” fungi, at the sampled scale. And for these fungi we uncover some spatial patterning that can be described relatively simply.

We did not supply an analysis of composition, in the sense of how different taxonomic groups vary among samples with distance and ecological conditions. Instead, our analysis were largely based upon

community (dis)similarity. All community dissimilarity measures were conducted at approximately the species level, for which ITS is probably best suited. As with most environmental sequencing studies, very many of our OTUs were not identified with high confidence, which chokes our ability to conduct higher-than-genus-level taxonomic analyses, or even to do extremely robust comparisons based on species identifications. We do present a picture of the ratios of the classes present in fungi that have been identified (Fig. 3), but without accompanying ecological analyses. Higher level taxonomic analyses in environmental sequencing efforts are probably better done with a region of the LSU. This direction of inquiry also seemed unlikely to be interesting, given the low amounts of variation explained by environmental variables at the species-level.

**2) Local patterns of diversity and richness will have a strong impact on community heterogeneity. To what extent do your heterogeneity results reflect richness gradients?**

The reviewer requested more information on the process of variance stabilization above. A discussion of this is required to answer this question.

Environmental sampling using culture-free, direct-sequencing technologies such as were employed in this study are generally considered to be very incomplete sampling of microbiomes. To give an example, in the case of the leaves, three leaves were taken from the host tree of at each site, from among thousands of leaves present in the canopy. From each of these three leaves, a single 1 cm<sup>2</sup> piece of leaf lamina was removed, from the several hundred of square centimeters of leaf tissue present in these leaves. Column purification of DNA followed, representing a sampling event of the available fungal genetic material. ITS-region primers were then used to bind to a subset of the fungal DNA present during PCR, representing another sampling event. In the illumina flow cell, the lawn of sticky oligonucleotides binds to the ITS-region amplicons in a manner that is ideally random, representing yet another sampling of the population. Thus at several points this process is nested, with multiple sub-sampling events of the microbiome of the leaves occurring.

Any and probably all of the steps in this process introduce large bias into the relative numbers of fungal species amplified, and into the relative abundances among these fungal species within any given sample. Unfortunately, these errors due to sampling are not simple to model – some samples receive deep coverage, others do not, and their read “abundances” do not correlate necessarily with “actual” abundance, from whatever upstream step we can consider “actual” (DNA? PCR product?). In the same way, the number of species recovered from a sample does not scale linearly with its original diversity. Number of species recovered in a sample is more a function of depth of coverage (which as mentioned doesn’t correlate well with “actual” upstream abundance) and also varies directly with ITS copy number, ease of DNA extraction, etc. of each of the species of endophytic fungi.

Variance stabilization attempts to model some of these sources of bias and allow comparisons among samples and species within samples. Here we use a method (“DeSeq2”) initially developed for RNAseq data, to make quantifiable comparisons of expression possible among genes and gene pathways possible despite these biases. But this and other methods can only claim to make *relative* differences in community composition among samples (or expression of mRNA sequences) meaningful.

All of this to say, that as we understand Reviewer #1, they would like some comparison of the original diversity of fungi within the samples, and unfortunately this cannot be honestly recovered from the sample prep and bioinformatic pipeline. we would not trust any patterns in species richness from any

stage of these pipelines, not due to any intellectual flaw in them but simply because these data and methods cannot produce that kind of data without large danger of false positives.

For more discussion of some of the above issues, and a list of useful sources, we have a preprinted discussion at <<https://doi.org/10.1101/184960>>.

**The figures are interesting, but in my opinion there are too many and they can be simplified. I'd suggest just removing the outliers from the NMDS plots and stating as much in the caption (no need to present the outliers), or just present a discontinuous scale.**

I've reduced the total number of figures and simplified some of the remaining, multi-paneled figures. Some are moved to supplementary information.

**I'm guessing that the average Fungal Ecology reader won't be familiar with the presentation of the PCNM vectors. Please explain this in the figure legend (what do the size of the bubbles represent, what are the values?).**

Language around interpretation of PCNM vectors has been added.

**Reviewer #2: Thank you for the opportunity to review the paper by Thomas et al. It's generally a nicely done study that places endophytes in the context of spatial and environmental factors, hosts, and tissue types (wood, leaves). Generally the work is solid and the stats top-quality. Most of my suggestions are cosmetic.**

**- Please try to avoid 'nouns stacked up as modifiers' like this: leaf fungal endophyte amplicon library preparations**

We removed the examples of this we could find.

**- Can the description of the FDP be shortened given the existence of previously published information?**

Section has been shortened.

**- Where possible avoid single-sentence paragraphs (as in results)**

We removed the examples of this we could find.

**- The results regarding 'Environmental effects on endophyte community composition' are difficult for me to understand/read.**

We've tried to condense and clarify, We hoped this helped.

**- In general the authors do a nice job of not over-inferring causality from the observational study, but occasionally that still comes through (e.g., environmental effects on). I suggest softening the language here and throughout, perhaps to match that used in the text of the preceding paragraph (Host species is the strongest predictor'...).**

Agreed, changed.

**- In general the results are often described more in terms of statistics than biological inference. The authors are terrific biologists, so I would encourage them to read this from the biological perspective and, using their good judgment to not over-infer/overstate biological stories (they're great about this), try to help the average endophyte biologist get to the biology presented here.**

Agreed. This paper is heavy on statistics and light on biology. Unfortunately, we have to be honest about the limitations of the techniques. This paper is a report of results from a high-throughput ITS survey of fungal endophytes. With this kind of data, most ecological conclusions must be drawn from broad, community-dissimilarity-based patterns. With medium and large community data, conclusions often have to be gleaned using tools that are not familiar for many biologists – eigenanalyses of various sorts, and complex spatial tools. Other, simpler and more classical statistical techniques are typically very inappropriate. Another of the main drawbacks to this kind of survey is that inevitably many species remain anonymous. And often these kind of data are massively noisy and rarely produce a simple message. The result of all this is a type of study that can be very unsatisfying to most mycologists. Most of us would prefer to be counting mushrooms/stromata, culturing our organisms, etc. But there are some important and basic questions concerning fungi that are best addressed through these means. We should note that another report from this data is in-progress that focuses on patterns of

Xylariaceae, on both the forest floor and in the canopy – the authors are pursuing more organisms-focused questions from this dataset! But here we are using this dataset to vet some broad theoretical ecological questions.

**- The summary comparison could be the first paragraph of the discussion instead of appended on the results.**

It is reviewed/summarized in the first section of the discussion. Hopefully the redundancy isn't too much.

**- I'm not a huge fan of the trend to use 'catchy titles with colons' when a scientifically descriptive title might be more effective.**

Title is changed.

**- I suggest being a little careful in the discussion to not over-focus on the peculiarities of the FDP or this study in a way that really limits the scope of inference. The hill may be important indeed, but in the absence of a control or other context, does that broaden the study as written? Consider retooling here to help others appreciate and apply your work to improving their own. I also think that getting into neutral factors may be a bit far abroad for the study as presented here.**

Hilltop references have mostly been removed. We would prefer to keep in the mention of neutral processes. Discussion of neutral processes is inevitable when environmental data falls short of explaining spatial patterns, and we have added language to be straightforward about our inability to test such ideas. When we point out the importance of southwestern valley as the only region of the plot where a core may be establishing in leaves. this is an implicit reference to neutral processes. Regardless of the niches of our core microbes, we suggest that perhaps it is merely protection from change that is allowing the microbiome of a host to begin developing some local structure. It is useful to at least mention this, and to be open about the theoretical framework being referenced. We would love to devote more space to developing these concepts, but the discussion is probably already too lengthy.

**- I wonder if it would be useful for the nested diagram in Fig 1 to be proportional to otu or other measures.**

There may be a little confusion on figure numbers. Fig. 1 in both the revised and original manuscript is a physical map. Some clarification is needed here.

**- In the upper panel of Fig 5, any reason not to remove the outlier and re-analyze? Ditto with the odd host in Fig 6? (Recentering partially addresses this, but the analysis may be more informative if repeated w/ and w/o that species, with supplements used to good effect).**

As mentioned elsewhere, we had to rebuild the bioinformatics pipeline, due to a loss of data. While redoing this we modified my contaminant removal process, using slightly more stringent method. This process resulted in cleaner NMS graphic in both cases, hopefully resolving this issue.

**- Interesting that the core mycobiome contains lots of common genera that show up all the time in endophyte studies. Maybe a comment on this?**

As mentioned, we had to rebuild the bioinformatic and statistical pipelines. This time, the high-throughput taxonomic assignments become much less certain using UTAX, so we manually blasted the core OTUs against UNITE. This gave me finer control over the quality cutoffs for the match, and showed me that most of the assignments of the core OTUs were very low confidence, so we had to discard them. Many of the genera were removed from the taxonomy table. *Phyllostica* remained, however, so the reviewer's comment is still pertinent. Specifics are in the scripts, and some language has been added to the conclusion.

**- Throughout, the language could be a little more linear. Nested clauses/etc. are sometimes a bit more prevalent than necessary.**

We've tried to improve this (but we can't help it, at least one of the authors often thinks, - and speaks - in a very nested manner).

**- Where possible I think figures could be combined and/or set up in a way to more clearly emphasize the key, generalizable results of the paper. Supplements could be used effectively to focus the paper more.**

I've moved several figures to supplements, and removed others.

**- Overall the paper is nice, and I think the text would benefit from a 'step-back-and-read' to be sure the authors' key points are clear.**

Agreed, we have struggled with the process of interpreting the really large and noisy dataset into a single cohesive story without oversimplifying, but also without confusing the reader. So we erred on the side of confusing the reader. In these revisions we have simplified manuscript as best as we can, though it remains a statistically heavy paper. we hope the situation is somewhat improved by these edits.

1  
2  
3 Spatial patterns of fungal endophytes in a subtropical montane rainforest of  
4 northern Taiwan  
5

6 *L add authors names  
7 addresses etc*

8 Abstract

9  
10 Fungal endophytes of plants are ubiquitous and important to host plant health. Wood-inhabiting and foliar  
11 endophyte communities from multiple tree hosts were sampled at multiple spatial scales across *Fushan forest*  
12 dynamics plot in northern Taiwan, using culture-free, community DNA amplicon sequencing methods. Fungal  
13 endophyte communities were distinct between leaves and wood, but the mycobiomes were highly variable  
14 across and within tree species. Despite this, host tree species was an important predictor of mycobiome  
15 community-composition. Within a single common tree species, “core” mycobiomes were characterized using co-  
16 occurrence analysis. The spatial cooccurrence patterns of these few species of fungal endophytes appear to  
17 explain the strong host effect. For wood endophytes, a consistent core mycobiome coexisted with the host  
18 across the extent of the study. For leaf endophytes, the core fungi resembled a more dynamic, “gradient” model  
19 of the core microbiome, changing across the topography and distance of the study.  
20  
21  
22  
23  
24  
25  
26  
27

28  
29  
30  
31  
32  
33  
34  
35  
36  
37  
38  
39  
40  
41  
42  
43  
44  
45  
46  
47  
48  
49  
50  
51  
52  
53  
54  
55  
56

Keywords:

Fungal Endophytes, Microbial ecology, ITS, Plant-fungal interactions, Fushan Subtropical Forest Dynamics Plot, Mycobiome, Core microbiome, PCNM/dbMEM spatial analysis

Introduction

Microbial community assembly and geographic patterns in microbes remain poorly understood, despite nearly a century of discussion (Baas-Becking, 1934; De Wit and Bouvier, 2006; Green and Bohannan, 2006; Hanson et al., 2012; Martiny et al., 2006; Nemergut et al., 2013; Peay et al., 2010). Rich microbial communities appear to be associated with all multicellular organisms (Hoffman and Arnold, 2010; Rosenberg et al., 2010). Host-associated microbes present additional complexity in modeling microbial community assembly, and raise questions concerning fidelity of host-microbe interactions. *Microbial community assemblage as an academic topic is important and fascinating, but this topic also takes on a new urgency in the discussion of plant microbiomes and plant health in a changing planet (Woodward et al., 2012).*

*order  
by  
data  
tree  
(alphabetical)  
reference  
list*

57  
58 The potential importance of microbes in adding ecological functions to their hosts (Johnson and  
59 Versalovic, 2012; Rodriguez et al., 2009; Woodward et al., 2012) has led some to suggest that multicellular  
60 organisms may host core microbiomes (Hamady and Knight, 2009; Shade and Handelsman, 2012;  
61 Vandenkoornhuyse et al., 2015), which are subsets of important and consistent microbial partners. Initial  
62 explorations of plant core microbiomes have been highly controlled (Edwards et al., 2015; Lundberg et al.,  
63 2012). Studies of plant-associated microbiomes in natural settings have rarely been framed in terms of core  
64 microbiomes (Bodenhausen et al., 2013; Higgins et al., 2014; Kembel and Mueller, 2014; Kim et al., 2012;  
65 Zimmerman and Vitousek, 2012, 2012). This is not a coincidence: outside of experimental settings, the prospect  
66 of detecting a cadre of microorganisms absolutely loyal to their host in the face of a complex and dynamic  
67 natural environment is daunting. This definition of the core microbiome, known as either a “substantial” or  
68 “minimal” core (Hamady and Knight, 2009) may be useful when carefully applied to long-studied symbioses such  
69 as ruminant gut communities (Liggenstoffer et al., 2010) or mycorrhizal relationships (Malloch et al., 1980; Van  
70 Der Heijden and Horton, 2009). This definition may not always serve for describing the other numerous and  
71 labyrinthine microbe-host interactions that occur between hosts and microbes. However, other definitions of core  
72 microbiomes exist that may be more useful for ecologically modeling microbiomes (Hamady and Knight, 2009).  
73  
74

75 Fungal endophytes, or fungi that live internally in plant tissues without causing disease symptoms (Wilson, 1995), are an important component of the plant microbiome. They have been shown to be widespread and important to plant health (Arnold et al., 2003; Mejía et al., 2008, 2007; Porras-Alfaro and Bayman, 2011; Rodriguez et al., 2009). The endophytic compartment in which they reside is a distinct ecological space, in the sense that very different communities of microbes are observed outside vs. inside plant tissues (Bodenhausen et al., 2013; Lundberg et al., 2012; Santamaría and Bayman, 2005), at least partly due to host-microbe preferences (Oldroyd, 2013; Schulz et al., 1999; Venkateshwaran et al., 2013). Plant organs have been shown to host distinct communities of endophytes (Bodenhausen et al., 2013; Edwards et al., 2015; Peršoh, 2013; Tateno et al., 2015). Endophyte communities are also influenced by environmental conditions (Arnold and Herre, 2003; Carroll and Carroll, 1978; Zimmerman and Vitousek, 2012), in spite of presumed buffering from environmental stresses by host tissues. Fungal communities are also subject to spatial processes such as dispersal limitation (Higgins et al., 2014; Peay et al., 2010) at multiple scales (Mummey and Rillig, 2008; Norros et al., 2012; Tedersoo et al., 2014). Fungal endophytes, therefore, make ideal systems for studying the interplay of host-microbe interactions, environmental influences, and spatial patterning of both host and microbes in natural  
112

113  
114 settings.  
115  
116

117 It must be acknowledged that plant hosts exert strong influence on community membership of their  
118 endophyte community. However, we hypothesized that even the most faithful fungal associates will uncouple  
119 from their hosts with changing environmental conditions and dispersal constraints. We predicted, on the scale of  
120 the present study, that plant mycobiomes resemble "gradient" core microbiomes (Hamady and Knight, 2009).  
121 Under this model, microbiomes can totally change across a landscape, with host-interactions mitigating but  
122 ultimately not preventing environmentally- and spatially-driven changes in the microbiome. In other words, we  
123 hypothesized that a persistent "core" of microbes shared among all members of a plant species does not truly  
124 exist in nature, on any meaningful scale. To test this, we compared community composition between wood and  
125 leaf fungal endophytes in multiple species of plant host across the landscape of the Fushan Forest Dynamics  
126 plot. We examined patterns in the total detected endophyte community of several plant hosts. Delving further  
127 within these data, we focused on a single tree host and the spatial patterns of its most strongly associated  
128 endophytic fungi.  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
162  
163  
164  
165  
166  
167  
168

## Materials and methods

Background/Site **J5** Sampling occurred in summer of 2013 at Fushan forest, in Northeastern Taiwan ( $24^{\circ} 45' 40''$   
N,  $121^{\circ} 33' 28''$  E), which hosts a 25-ha Smithsonian-associated Forest Dynamics Plot (FDP) (Losos and Leigh,  
2004; Su et al., 2007). Fushan is a humid subtropical old-growth montane site that receives 4.27 m of rain each  
year. Most of this precipitation falls during rainy, cool winters, though a significant fraction of this rain is due to  
typhoons, the main agent of disturbance in this system, during warm summer months. The flora is diverse,  
characterized by many evergreen broadleaf tree species and a diverse understory of lianas, ferns, tree ferns,  
and other herbs, graminoids, and shrubs. Vegetative communities can be broadly categorized into four  
community types described by dominant tree species combinations (Fig. 1). Topography is highly variable, with a  
maximum elevation of 733 m above sea level at an approximately central hilltop within the FDP, and a minimum  
of 600 m, though the present study sampled areas only as low as 650 m. The complex topography of Fushan  
has been summarized by classification of each 20 m x 20 m quadrant of the FDP into one of seven habitat  
types, based on aspect, slope, convexity, and elevation (Figs.1 and 2), which are found to influence vegetative  
communities (Su et al., 2010).

## 169 170 171 172 Field methods

173  
174 Fushan FDP was divided into 9 sub-plots, and subplots were sampled using a nested logarithmic scheme  
175 intended to detect dispersal limitation and community turnover (Rodrigues et al., 2013) (Fig. 2). Sampling of  
176 each set of nested points was undertaken in random order. Once sampling of a single set of nested squares had  
177 begun, all points within that set of nested points were sampled prior to beginning another. Six out of nine sets of  
178 nested squares were sampled, due to time constraints. For each sampling point, we located the tree with the  
179 largest DBH with canopy above the point and collected the three lowest "healthy" appearing leaves that were  
180 safely reachable. Leaves and accompanying woody stems were obtained using a 3m collapsible pole pruner.  
181 Only  
182 Identification of host-tree was ~~supplied~~<sup>determined from</sup> survey data from ongoing ecological research at Fushan FDP (Su et  
183 al., 2007). All plant material was carried to a nearby field station and stored at 4°C for no longer than 5 days  
184 before processing.

## 185 186 187 188 189 190 191 Lab methods

192  
193 Preparation and sequencing of Illumina libraries for leaves and wood were undertaken separately, with differing  
194 protocols. Protocols for leaf library preparations are given in Vandegrift (2016). Protocols for wood endophytes  
195 are given in detail in Thomas et al. (2017). Briefly, all leaves were washed and surface-sterilized, and woody  
196 stem material was debarked with a sterile scalpel and phloem and sapwood were harvested. DNA was extracted  
197 from both in separate library preparations and ITS region 1 was amplified using a fungal-specific primer set with  
198 illumina© tagged, barcoded primers. Positive, "mock community" controls were included in the wood-endophyte  
199 library, and pure-water negative controls were included in both libraries. Samples were multiplexed and  
200 sequenced in separate illumina© MiSeq sequencer runs.

## 201 202 203 204 205 206 207 208 209 210 211 212 Bioinformatics

213 Details of the bioinformatics pipeline are explained in Thomas et al. (2017). Full scripts are available in  
214 supplementary information, and as a jupyter notebook at:

215 <[https://nbviewer.jupyter.org/github/danchurch/taiwan\\_combined\\_biom/blob/master/combo\\_biome-revived.ipynb](https://nbviewer.jupyter.org/github/danchurch/taiwan_combined_biom/blob/master/combo_biome-revived.ipynb)>.

216 A bit more information is needed  
217 required. What primers were  
218 used? To what extent  
219 did they amplify tree  
220 DNA? i.e. what  
221 proportion of sequence  
222 was the wood then  
223 surface sterilized?  
224 Spores from bark could land on  
225 wood during bark removal

Just citing  
this is not the paper  
does not appear to  
have been published

Briefly, general bioinformatics protocols followed the USEARCH/UPARSE pipeline version 8.1 (Edgar, 2013) wherever possible. Libraries of leaf and wood fungal endophyte DNA were prepared separately, so to maximize comparability, the reads from both libraries were combined as early as possible in the bioinformatics pipeline, following merging of paired ends. Variance stabilization, the process of maximizing comparability among samples and studies while considering differences in sampling depth, was done using the DESeq2 and phyloseq packages in R (Love et al., 2014; McMurdie and Holmes, 2013), using leaf/wood as the design variable. Positive controls were used to calibrate OTU similarity radius and minimum cutoffs, which were subtracted from all observations to reduce error from index-misassignment and artificial splitting of OTUs. Large differences in abundances remained among positive control OTUs even after variance stabilization, so all statistical analyses were conducted with incidence (presence/absence)-transformed community matrices.

Initial taxonomic assignments were assigned to all possible reads in the USEARCH 8.1 UTAX algorithm against the UNITE fungal ITS database (Edgar, 2013; Kõljalg et al., 2013). This is a high-throughput method, used here to create an aggregate, “big picture” of the endophytes communities, summarized in Figure 3. Because of the low confidence of many of the high-throughput identifications, classes containing less than 1% of total OTUs were not included in this figure, but are available for inspection in scripts. Manual curation of taxonomic calls was then required for members of the core microbiomes. For higher confidence taxonomic calls, members of the wood and leaf core microbiome were assigned taxonomy using the BLAST (Altschul et al., 1990), against the UNITE database. See scripts for details of this process.

## Statistical methods

### Overview

Ecological patterns of the entire fungal community of leaves and wood of all hosts were examined first. Analyses were then focused on patterns in the mycobiome of the most commonly-sampled host tree, *Helicia formosana*. Finally, host-fungus co-occurrence patterns were used to define a core mycobiome that was also examined for ecological patterns. Statistical analysis was conducted in R Statistical Software, version 3.3.1 (R Core Team, 2017), with the vegan (Oksanen et al., 2017), phyloseq (McMurdie and Holmes, 2013), cooccur (Griffith et al., 2016), igraph (Csardi and Nepusz, 2006) and ecodist (Goslee and Urban, 2007) packages. The statistical pipeline is available as a jupyter notebook, viewable at:

281  
282 <[https://nbviewer.jupyter.org/github/danchurch/taiwan\\_combined\\_stats/blob/master/Csrev/Csrev.ipynb](https://nbviewer.jupyter.org/github/danchurch/taiwan_combined_stats/blob/master/Csrev/Csrev.ipynb)>.  
283  
284  
285

286 All read abundances in fungal community matrices were transformed to presence/absence (see  
287 bioinformatics, above), so all downstream community analyses were based on this incidence data. Tests and  
288 visualizations that required the use of dissimilarity included PERMANOVA and Mantel tests, NMS and Bray-  
289 Curtis map visualizations. In all cases endophyte community comparisons were conducted using Bray-Curtis  
290 distances (Bray and Curtis, 1957; McCune et al., 2002).  
291  
292  
293

294  
295  
296 Mycobiome of all hosts  
297  
298

299  
300 Dissimilarity of leaf and wood endophyte communities were as a first step modeled and visualized using non-  
301 parametric multivariate analysis of variance (NPMANOVA or PERMANOVA) (Anderson, 2017, 2001), and non-  
302 metric multidimensional scaling (NMS). Following initial comparisons of community composition, all analyses of  
303 wood and leaf endophytes were conducted separately, in parallel. Host and environmental variables of  
304 vegetative community and topography (Fig. 1) as predictors of endophyte communities were modeled  
305 individually using PERMANOVA. However, PERMANOVA tests as employed here are not sensitive to the  
306 complexities of spatial patterns on the variable landscape of Fushan FDP. For this reason spatially-explicit tests  
307 for autocorrelation and correlations with endophyte community composition and environmental data were used.  
308  
309  
310  
311  
312  
313  
314  
315

316 Spatial trends in endophyte communities were first explored using multivariate Mantel tests (Legendre  
317 and Fortin, 1989; Mantel, 1967) of community dissimilarity matrices against physical distance matrices, and  
318 visualized with Mantel multivariate correlograms. For greater resolution of spatial trends, distance-based Moran's  
319 eigenvector maps analysis, also known as Principal Components of Neighbor Matrices (PCNM) analysis, was  
320 conducted on our sampling scheme. Following the general statistical pipeline recommended by Legendre et al.  
321 (Borcard et al. 2011) Legendre and Legendre, 2012), endophyte community matrices were Hellinger-  
322 transformed (Legendre and Gallagher, 2001), and "regressed" using Redundancy analysis (RDA) (Buttigieg and  
323 Ramette, 2014; Legendre and Gallagher, 2001) against all eigenvectors ("PCNM vectors") resulting from dbMEM  
324 analysis. Stepwise model selection was then used to filter the ecologically informative eigenvectors (Oksanen et  
325 al., 2017).  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336

337  
338 Eigenanalysis of sampling schemes such as dbMEM/PCNM used here give a portrait of what patterns can be  
339 confidently tested by a particular sampling scheme. They are useful in that they can approximate classical  
340 linear spatial patterns, but also model other, more complex spatial patterns. These eigenvectors represent a  
341 range of spatial patterns that are periodic functions of wavelengths of varying size and direction. When  
342 combined with constrained direct gradient analyses such as Canonical Correspondence Analysis (CCA) or  
343 Redundancy analysis (RDA), eigenanalyses of sampling schemes become a very sensitive tool for detecting  
344 important spatial patterns in biological communities.  
345  
346

347 As a first filter, only PCNM vectors that are in some way correlated with changes in the community  
348 matrix are considered ecologically meaningful and are retained. Following this, if a correlation is found with these  
349 PCNM vectors with either between differences in fungal community or with patterns in environmental data, we  
350 can infer that these environmental or community differences behave on the landscape somewhat like the PCNM  
351 vectors. This is especially true if a large amount of variance in the community matrix or environmental data is  
352 explained by a PCNM (i.e.  $R^2$  is high). The wavelengths and shapes of PCNM vectors are often visualized  
353 using “bubble” plots, 2-D scatter plot with proportionally sized and colored symbols (Borcard et al., 2011).  
354 Ecological patterns of interest detected in spatial analysis were further visualized by mapping Bray-Curtis  
355 distance of all wood or leaf samples from a single point of interest on maps of the reserve and in NMS  
356 ordinations.  
357  
358

### 359 Mycobiome of a single host, *Helicia formosana*

360 To exclude variation in fungal communities resulting from differing host tree species, above analyses were  
361 repeated for the fungal endophytes of a single host tree, *Helicia formosana* [cut & Hems]. This was the host  
362 tree for which the most samples (leaves, n=31; wood n=22) were available. Bray-Curtis dissimilarity values  
363 resulting from comparison were then plotted onto a map of Fushan FDP.  
364  
365

### 366 Core fungi of *Helicia formosana*

367 To test for the presence of a core mycobiome, co-occurrence analysis was conducted on the all-host, all-  
368 endophyte species-using a pairwise, probabilistic model (Griffith et al., 2016; Veech, 2013). Core mycobiomes of  
369 hosts were defined as the subset of fungi that showed strong co-occurrence associations with a host. Strong  
370  
371

393  
394 associations were defined as those with probabilities under null models of random association corrected to a  
395 Benjamini-Hochberg false discovery rate (FDR) of 0.05 or less. Fungal OTUs found to be strongly cooccurring  
396 with *H. formosana* were used to populate a species composition matrix of just these core species as columns,  
397 with rows of just sites where *H. formosana* was sampled. Patterns of this subset of core fungi were then  
398 visualized by first calculating Bray-Curtis dissimilarity distance of each sample (row) of this subsetted “core  
399 matrix” from an idealized core mycobiome row that contained all members of the core fungi. These values were  
400 then mapped on the Fushan FDP plot.  
401  
402  
403  
404  
405

## 406 Results

### 407

#### 408 Mycobiome of all hosts

409  
410 After variance-stabilization, in the leaf endophyte library 1302 OTUs were detected, and 2025 OTUs in the  
411 wood library. Both leaf and wood samples were dominated by Ascomycota (Fig. 3), but a larger proportion of  
412 remaining reads in wood OTUs matched to Basidiomycota (10% of leaf OTUs vs. 17% of wood OTUs) than in leaves.  
413 This larger percentage of OTUs identified to Basidiomycota was due mostly to a larger percentage of  
414 OTUs identified to Tremellomycetes (<1% of leaf OTUs vs. 5.6% of wood OTUs). Within Ascomycota, both leaf  
415 and wood samples contained high percentages of OTUs identified to Sordariomycetes (24% of leaf OTUs vs.  
416 16% of wood OTUs), Dothideomycetes (20% of leaf OTUs vs. 27% of wood OTUs), and Eurotiomycetes (7% of  
417 leaf OTUs vs. 14% of wood OTUs). Significant observations of Lecanoromycete fungi also occurred, especially  
418 in leaves (%18.5 of leaf OTUs vs. %4 of wood OTUs). At the class level 16% of leaf OTUs were unidentified,  
419 compared to 10% of wood.

420 Host species was the strongest single predictor of similarity within both leaf (PERMANOVA,  $F(33, 89) =$

421 1.95,  $p < 0.01$ ,  $R^2 = 0.42$ , permutations = 10000). Wood endophyte communities were also most strongly  
422 predicted by host (PERMANOVA,  $F(29, 61) = 1.49$ ,  $p < 0.01$ ,  $R^2 = 0.42$ , permutations = 10000), see NMS  
423 visualizations (Fig. 4). Vegetative community is a weak predictor of similarity in both leaf (PERMANOVA,  $F(3,$   
424 119) = 2.04,  $p < 0.01$ ,  $R^2 = .05$ , permutations = 10000) and wood endophyte community (PERMANOVA,  $F(3, 87)$   
425 = 1.71,  $p < 0.01$ ,  $R^2 = 0.055$ , permutations = 10000). Micro-topographic conditions were also weak predictors of  
426 similarity in both leaf (PERMANOVA,  $F(6, 116) = 1.23$ ,  $p < 0.01$ ,  $R^2 = 0.06$ , permutations = 10000) and wood

449  
450 endophyte community (PERMANOVA,  $F(6,84) = 1.36$ ,  $p < 0.01$ ,  $R^2 = 0.09$ , permutations = 10000).  
451  
452 PERMANOVA tests cannot easily account for complex spatial patterns, some environmental correlations were  
453 uncovered in more sensitive spatial analyses summarized below.  
454

455  
456 Wood endophyte community displayed a weak pattern of community-turnover/distance-decay over the  
457 entire study area (global Mantel's  $r = 0.10$ ,  $p < 0.01$ ) (Supp Fig. 1). Leaf communities displayed no global  
458 distance decay relationship (global Mantel's  $r = -0.01$ ,  $p = 0.57$ ), but displayed local negative autocorrelation in  
459 comparisons of samples approximately 200 meters apart (Mantel correlogram, local Mantel's  $r = -0.10$ ,  $p < 0.05$ ).  
460 In both wood and leaf endophyte communities of all hosts, Mantel's  $r$  approaches zero in both leaf and wood  
461 samples at comparisons around 150 m apart, indicating that positive autocorrelation is ~~is~~ undetected beyond this.  
462  
463  
464  
465  
466  
467  
468

469 For leaves, our sampling scheme yielded 7 biologically significant PCNM vectors, explaining a combined  
470 total of 7.1% of endophyte community variation (Redundancy analysis, constrained inertia = 0.06, Unconstrained  
471 inertia = 0.82,  $F(7,117) = 1.25$ ,  $P < 0.01$ ) (Supp. Fig. 2). Of these, the smallest scale vectors are ~~were~~ uncorrelated  
472 with environmental data, and probably indicative of endogenous autocorrelation (Borcard et al., 2011), with  
473 wavelengths up to 50m. Two mid-range (300m) vectors represented north-south and east-west surface trends,  
474 unexplained by environmental data, each predicting approximately 1 % of fungal species variation. Two mid-  
475 range leaf PCNM vectors of interest correlate with environmental variables, most strongly with the presence of  
476 the steep habitat zone present mostly on the central hill of the plot (Linear model/multiple regression, adj-  
477  $R^2=0.23$ ,  $F(9,113)=4.95$ ,  $p < 0.01$ ). The largest scale (500 m) PCNM vector ran in a NE-SW direction, resulting  
478 in a partial contrast between the southwest valley and the rest of the plot. It predicted 1.2% of leaf endophyte  
479 community variation, and was most strongly explained by the presence of the two vegetation communities  
480 dominated by the tree *Helicia formosana* (Linear model/multiple regression, adj- $R^2=0.33$ ,  $F(9,113)=7.68$ ,  $p <$   
481 0.01).  
482  
483  
484

485 For wood, our sampling scheme also yielded 5 biologically significant PCNM vectors, explaining 7.5% of  
486 variation (Redundancy analysis, constrained inertia = 0.07, Unconstrained inertia = 0.80,  $F(5,85) = 1.38$ ,  $P <$   
487 0.01). Of these, the smallest scale vectors are ~~were~~ probably indicative of endogenous autocorrelation, at a  
488 wavelength of 70 m or less. The three other PCNM vectors are large- to mid-range, with lengths from 300m to  
489 500m. They correlated with environmental variables, and generally point to a contrast between upland and  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504

505  
506 lowland habitat (Supp. Fig. 3). For more detailed results and interpretation of all-host PCNM results, see scripts.  
507  
508  
509  
510

indicate  
location of  
host

## 511 Mycobiome of a single host, *Helicia formosana*

512  
513 When the community of endophytes was constrained to one host tree species, environmental variables were not  
514 found to directly explain any variance in endophyte community structure (PERMANOVA tests,  
515 permutations=10000. Leaf community ~ topography:  $F(4,26) = 0.93$ ,  $R^2 = 0.12$ ,  $p = 0.66$ . Leaf community ~  
516 vegetative community:  $F(3,27) = 1.07$ ,  $R^2 = 0.11$ ,  $p = 0.27$ . Wood community ~ topography:  $F(4,17) = 1.06$ ,  $R^2$   
517 = 0.20,  $p = 0.20$ . Wood community ~ Vegetative community:  $F(3,18) = 1.08$ ,  $R^2 = 0.15$ ,  $p = 0.19$ .).

518  
519  
520  
521  
522  
523  
524 Leaf endophyte community yielded ~~3~~<sup>three</sup> ecologically significant PCNMs explaining 13% of community  
525 variation (RDA, leaves: constrained inertia = 0.011, Unconstrained inertia = 0.67,  $F(2,19) = 1.32$ ,  $P < 0.01$ ).  
526  
527 Wood endophyte community yielded ~~2~~<sup>two</sup> ecologically significant PCNMs explaining 12% of community variation  
528 (RDA, leaves: constrained inertia = 0.10, Unconstrained inertia = 0.70,  $F(1,29) = 1.78$ ,  $P < 0.01$ ). As with the  
529 non-spatial PERMANOVA model of *Helicia* endophytes, these PCNMs also did not correlate with any available  
530 environmental data, in either wood or leaves. However, both wood and leaf communities showed ecologically  
531 meaningful PCNM vectors that center on the southwestern valley of the FDP as a place of difference in the  
532  
533 *Helicia* mycobiome (Fig. 5). This pattern of dissimilarity was particularly pronounced in the *Helicia* leaf endophyte  
534 community when visualized with Bray-Curtis comparisons, (Fig. 6).  
535  
536  
537  
538  
539

## 540 Core fungi of *Helicia formosana*

541 Out of 1302 possible fungal OTUs observed in leaves of all hosts, 426 OTUs were found in *Helicia* leaves. Of  
542 these 12 showed strong patterns of co-occurrence with *Helicia formosana*. Out of 2025 possible fungal OTUs  
543 observed in woody tissue of all hosts, 731 OTUs were found in *Helicia* wood (Supp. Fig. 4). Of these 7 OTUs  
544 showed strong patterns of co-occurrence with *Helicia formosana* (Supp. Fig. 4). These fungi were considered  
545 members of the *H. formosana* core mycobiome (Table 1). Visual inspection of patterns of dissimilarity show that  
546 only leaves within the southern valley of the plot contained relatively high proportions of core fungi (Fig. 7).  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560

Wood samples retained most of their core fungi consistently throughout the plot (Fig. 7).

## Summary comparison

The above analysis compared patterns of community dissimilarity at several levels (Table 2). Wood and leaf endophytes of all host-trees showed a similar pattern, very high levels of dissimilarity among all samples (all-host leaf endophyte mean BC=0.90, sd=0.09; wood endophyte mean BC=0.87, sd=0.07). *Helicia formosana* samples show a lower average level of dissimilarity (leaf mean BC=0.80, sd =0.11; wood mean BC = 0.80, sd=0.06). This variation can then be partitioned into two groups: (1) non-core fungi, which show a high average level of dissimilarity (leaf mean BC=0.89, sd =0.08 ; wood mean BC = 0.83, sd=0.06), and (2) core fungi, which show a lower mean BC (leaf mean BC=0.38, sd =0.17; wood mean BC = 0.39, sd=0.17).

## Discussion

You have written the discussion in the present tense, but I think mostly it should be in the past tense

Relate this to published literature  
See Parkitt et al. (2010)  
Fungal Ecology

### Patterns observed

But could this be because some were contaminants if you bark removed was not sufficiently ascropic - discuss

Contrary to our predictions, we found evidence for a consistent core of fungi in the wood of *Helicia formosana*.

Closer to our predictions, we found in leaves a subset of fungi that cooccurred with *H. formosana* in just one area of the plot (Fig. 7). In leaves, these core fungi are most consistently present in the southern valley, and are often completely missing in other areas of the study. In wood, they are more "loyal", and coexist more reliably with *H. formosana* throughout the plot. This may perhaps be due to the high rate of turnover in leaves, which are flushed mostly sterile (Arnold and Herre, 2003), and are shed within 1 to several years, in contrast with the longer lifespan of woody tissues. Applying terminology proposed by Hamady and Knight (2009), core woody endophytes here may be best described by the "minimal" core model: they are few in number among a large and highly variable microbiome, but are consistently present throughout the study. In contrast, leaf endophytes may be described better by "gradient" or "subpopulation" core models, where a core group of associated microbes may establish with a particular host, but whose presence is highly conditional on space and environment.

When all host trees were compared, the average dissimilarity between any two trees was extremely high, (Fig. 8, Table 2). Samples were slightly become somewhat more similar on average when constrained to a single host, for wood and leaves, seemingly a result of the strong effects of host (Figs. 4 and 8). The taxonomic core of fungi

617  
618 behaves differently from the fungal microbiome at large. Removing these fungi from consideration brings the  
619 mycobiome of their host, *H. formosana*, nearly back to the high background levels of dissimilarity among  
620 samples of the entire study, indicating that these may be the species through which host effects are manifested  
621 (Fig. 8).  
622  
623  
624

## 625 626 Fungal species 627

628 Most of the candidate core fungal species of the tree *Helicia formosana* were unidentified (Table 1), often even  
629 to the phylum level. This makes specific ecological interpretation of the individual core fungi difficult or  
630 impossible. It also speaks to an urgent and daunting challenge facing mycologists today: the need for more high-  
631 quality accessions of lesser known fungi in fungal barcode databases, to keep pace with the ongoing technical  
632 revolution of high-throughput sequencer technologies.  
633  
634

635 Within the candidates for core fungi of *H. formosana*, *Phyllosticta* were present as least three  
636 endophytes OTUs in leaves. This included *P. capitalensis*, which is a well-known endophyte of tropical woody  
637 plants (Baayen et al., 2002), that may have been transported throughout the world via nursery trade (George  
638 Carroll, pers. com.) Small differences in genotype can determine a fungal species' tendency to a pathogenic or  
639 endophytic-mutualistic lifestyle (Carroll, 1988; Rodriguez and Redman, 2008). This has apparently resulted in  
640 genera that contain numerous examples of both pathogenic<sup>s</sup> and endophytic<sup>es</sup> within genera such as  
641 *Colletotrichum* (Rojas et al., 2010), *Mycosphaerella* (Crous, 1998; Lau et al., 2013), and *Phyllosticta* (Glienke et  
642 al., 2011). These differences imply that radically different ecological effects can result from small changes in a  
643 fungal genome. These three *Phyllosticta* species, though closely related, may have very different interactions  
644 with their host, *H. formosana*, which cannot be understood without further investigation.  
645  
646

## 647 The value of the taxonomic Core 648

649 Some have argued that the concept of a taxonomic core should be augmented or replaced by a *functional* core  
650 (Lemanceau et al., 2017; Lloyd-Price et al., 2016). This is an important and complementary approach to  
651 understanding microbe and host interactions, and does not contradict taxonomic delineation of cores. On either  
652 side, the concept of a taxonomic core is useful for understanding the assembly rules of microbial communities.  
653  
654

673  
674  
675  
676  
677  
678  
679  
680  
681  
682  
683  
684  
685  
686  
687  
688  
689

This is stuff I like

What extreme spatial scale?

extreme of spatial scales, the search for a taxonomic core is probably meaningless – cosmopolitan species such as humans will likely show complete turnover microbiomes throughout their global range (The Human Microbiome Project Consortium, 2012). Extremely localized searches for core species of a host are nonsensical.

Why are we suddenly talking about humans?

At well-defined, intermediate spatial scales such as examined here, a search for a taxonomic core retains its usefulness to microbial ecology: patterns such as those observed here would be extremely difficult to quantify from a purely functional perspective. On some level, disagreement over taxonomic vs. functional cores is another example of a false, reductionist-versus-holistic dichotomy (Fang and Casadevall, 2011). Discarding either approach results in a loss of information.

690  
691  
692  
693  
694  
695  
696  
697  
698  
699  
700  
701  
702  
703  
704  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728

The presence of a core taxonomic group of microbes in a host might be considered a kind of stabilization or structuring of a portion of a host's microbiome, possibly as a result of interactions among hosts and select microbes. When defining core microbiomes as we have here, it may be important to consider the different organs of hosts as very different refugia for microbes: here the woody tissues appeared to host a more consistent assemblage of core fungi. Similarly, the leaves *Helicia formosana* trees in the more sheltered southwestern valley held more a consistent microbial core than in more exposed areas of the plot.

We are limited here in our ability to examine the importance of neutral spatial processes, given our coarse environmental data. However, these patterns suggest that even strong biological interactions between microbe and host can be disrupted or prevented. This disruption could stem from neutral processes such as obstacles to dispersal from topography, or environmental changes such as fierce summer storm events resetting community assembly in leaves. As such, it may be that for a consistent taxonomic core to develop in a plant microbiome, either local habitat or more persistent host tissue may need to provide some measure of stability from change. High rates of dispersal and disturbance can disrupt the tendency to local structure in communities and gene pools (Cadotte, 2006; Vellend, 2010; Wright, 1940). A parallel logic may apply for taxonomic microbiomes of large hosts.

What do you mean?

## Acknowledgements

water  
Ken West  
contaminants

This project was jointly funded by the USA National Science Foundation and the National Science Council in Taiwan, under the 2013 East Asia and Pacific Summer Institutes program (EAPSI program 12-498). Additional funds for laboratory work were contributed by the Cascade Mycological Society, and the Mycological Society of America. We give our immense thanks to Dr. Sheng-Hsin Su and colleagues in the 2013 Fushan tree census crew, for supporting our field work. The Fushan Forest Dynamics Plot is a collaborative project of the Taiwan Forestry Research Institute, Taiwan Forestry Bureau, and National Taiwan University, and was funded by the Council of Agriculture and National Science Council in Taiwan.

- Altschul, S.F., Gish, W., Miller, W., Myers, E.W., Lipman, D.J., 1990. Basic local alignment search tool. *Journal of Molecular Biology* 215, 403–410. [https://doi.org/10.1016/S0022-2836\(05\)80360-2](https://doi.org/10.1016/S0022-2836(05)80360-2)
- Anderson, M.J., 2017. Permutational Multivariate Analysis of Variance (PERMANOVA), in: Wiley StatsRef: Statistics Reference Online. American Cancer Society, pp. 1–15. <https://doi.org/10.1002/9781118445112.stat07841>
- Anderson, M.J., 2001. A new method for non-parametric multivariate analysis of variance. *Austral Ecology* 26, 32–46. <https://doi.org/10.1111/j.1442-9993.2001.01070.pp.x>
- Arnold, A.E., Herre, E.A., 2003. Canopy cover and leaf age affect colonization by tropical fungal endophytes: Ecological pattern and process in *Theobroma cacao* (Malvaceae). *Mycologia* 95, 388–398. <https://doi.org/10.1080/15572536.2004.11833083>
- Arnold, A.E., Mejía, L.C., Kyllo, D., Rojas, E.I., Maynard, Z., Robbins, N., Herre, E.A., 2003. Fungal endophytes limit pathogen damage in a tropical tree. *PNAS* 100, 15649–15654. <https://doi.org/10.1073/pnas.2533483100>
- Baas-Becking, L.G.M., 1934. Geobiologie; of inleiding tot de milieukunde. WP Van Stockum & Zoon NV.
- Baayen, R.P., Bonants, P.J.M., Verkley, G., Carroll, G.C., van der Aa, H.A., de Weerdt, M., van Brouwershaven, I.R., Schutte, G.C., Maccheroni, W., de Blanco, C.G., Azevedo, J.L., 2002. Nonpathogenic Isolates of the Citrus Black Spot Fungus, *Guignardia citricarpa*, Identified as a Cosmopolitan Endophyte of Woody Plants, *G. mangiferae* (*Phyllosticta capitalensis*). *Phytopathology* 92, 464–477. <https://doi.org/10.1094/PHYTO.2002.92.5.464>
- Bodenhausen, N., Horton, M.W., Bergelson, J., 2013. Bacterial Communities Associated with the Leaves and the Roots of *Arabidopsis thaliana*. *PLOS ONE* 8, e56329. <https://doi.org/10.1371/journal.pone.0056329>
- Borcard, D., Gillet, F., Legendre, P., 2011. Spatial analysis of ecological data, in: Numerical Ecology with R. Springer, pp. 227–292.
- Bray, J.R., Curtis, J.T., 1957. An ordination of the upland forest communities of southern Wisconsin. *Ecological monographs* 27, 325–349.
- Buttigieg, P.L., Ramette, A., 2014. A guide to statistical analysis in microbial ecology: a community-focused, living review of multivariate data analyses. *FEMS microbiology ecology* 90, 543–550.
- Cadotte, M.W., 2006. Dispersal and Species Diversity: A Meta-Analysis. *The American Naturalist* 167, 913–924. <https://doi.org/10.1086/504850>
- Carroll, G., 1988. Fungal endophytes in stems and leaves: from latent pathogen to mutualistic symbiont. *Ecology* 69, 2–9.

- 785  
786 Carroll, G.C., Carroll, F.E., 1978. Studies on the incidence of coniferous needle endophytes in the  
787 Pacific Northwest. Canadian Journal of Botany 56, 3034–3043.  
788 Crous, P.W., 1998. *Mycosphaerella* spp. and their anamorphs associated with leaf spot diseases of  
789 Eucalyptus. American Phytopathological Society (APS Press).  
790 Csardi, G., Nepusz, T., 2006. The igraph software package for complex network research. InterJournal,  
791 Complex Systems 1695, 1–9.  
792 De Wit, R., Bouvier, T., 2006. ‘Everything is everywhere, but, the environment selects’: what did Baas  
793 Becking and Beijerinck really say? Environmental Microbiology 8, 755–758.  
794 https://doi.org/10.1111/j.1462-2920.2006.01017.x  
795 Edgar, R.C., 2013. UPARSE: highly accurate OTU sequences from microbial amplicon reads. Nature  
796 Methods 10, 996–998. https://doi.org/10.1038/nmeth.2604  
797 Edwards, J., Johnson, C., Santos-Medellín, C., Lurie, E., Podishetty, N.K., Bhatnagar, S., Eisen, J.A.,  
798 Sundaresan, V., 2015. Structure, variation, and assembly of the root-associated microbiomes of  
800 rice. PNAS 112, E911–E920. https://doi.org/10.1073/pnas.1414592112  
801 Fang, F.C., Casadevall, A., 2011. Reductionistic and Holistic Science. Infect Immun 79, 1401–1404.  
802 https://doi.org/10.1128/IAI.01343-10  
803 Glienke, C., Pereira, O.L., Stringari, D., Fabris, J., Kava-Cordeiro, V., Galli-Terasawa, L., Cunningham,  
804 J., Shivas, R.G., Groenewald, J.Z., Crous, P.W., 2011. Endophytic and pathogenic Phyllosticta  
805 species, with reference to those associated with Citrus Black Spot. Persoonia 26, 47–56.  
806 https://doi.org/10.3767/003158511X569169  
807 Goslee, S.C., Urban, D.L., 2007. The ecodist package for dissimilarity-based analysis of ecological  
808 data. Journal of Statistical Software 22, 1–19.  
809 Green, J., Bohannan, B.J.M., 2006. Spatial scaling of microbial biodiversity. Trends in Ecology &  
810 Evolution 21, 501–507. https://doi.org/10.1016/j.tree.2006.06.012  
811 Griffith, D.M., Veech, J.A., Marsh, C.J., 2016. Cooccur: probabilistic species co-occurrence analysis in  
812 R. J Stat Softw 69, 1–17.  
813 Hamady, M., Knight, R., 2009. Microbial community profiling for human microbiome projects: Tools,  
814 techniques, and challenges. Genome Res. https://doi.org/10.1101/gr.085464.108  
815 Hanson, C.A., Fuhrman, J.A., Horner-Devine, M.C., Martiny, J.B.H., 2012. Beyond biogeographic  
816 patterns: processes shaping the microbial landscape. Nature Reviews Microbiology 10, 497–  
817 506. https://doi.org/10.1038/nrmicro2795  
818 Higgins, K.L., Arnold, A.E., Coley, P.D., Kursar, T.A., 2014. Communities of fungal endophytes in  
819 tropical forest grasses: highly diverse host- and habitat generalists characterized by strong  
820 spatial structure. Fungal Ecology 8, 1–11. https://doi.org/10.1016/j.funeco.2013.12.005  
821 Hoffman, M.T., Arnold, A.E., 2010. Diverse Bacteria Inhabit Living Hyphae of Phylogenetically  
822 Diverse Fungal Endophytes. Appl. Environ. Microbiol. 76, 4063–4075.  
823 https://doi.org/10.1128/AEM.02928-09  
824 Johnson, C.L., Versalovic, J., 2012. The Human Microbiome and Its Potential Importance to Pediatrics.  
825 Pediatrics peds.2011-2736. https://doi.org/10.1542/peds.2011-2736  
826 Kembel, S.W., Mueller, R.C., 2014. Plant traits and taxonomy drive host associations in tropical  
827 phyllosphere fungal communities. Botany 92, 303–311.  
828 Kim, M., Singh, D., Lai-Hoe, A., Go, R., Rahim, R.A., A.n, A., Chun, J., Adams, J.M., 2012.  
829 Distinctive Phyllosphere Bacterial Communities in Tropical Trees. Microb Ecol 63, 674–681.  
830 https://doi.org/10.1007/s00248-011-9953-1  
831 Kõlalg, U., Nilsson, R.H., Abarenkov, K., Tedersoo, L., Taylor, A.F.S., Bahram, M., Bates, S.T.,  
832 Bruns, T.D., Bengtsson-Palme, J., Callaghan, T.M., Douglas, B., Drenkhan, T., Eberhardt, U.,  
833 Dueñas, M., Grebenc, T., Griffith, G.W., Hartmann, M., Kirk, P.M., Kohout, P., Larsson, E.,  
834 Lindahl, B.D., Lücking, R., Martín, M.P., Matheny, P.B., Nguyen, N.H., Niskanen, T., Oja, J.,  
835 Peay, K.G., Peintner, U., Peterson, M., Põldmaa, K., Saag, L., Saar, I., Schüßler, A., Scott, J.A.,  
836  
837  
838  
839  
840

- 841  
842 Senés, C., Smith, M.E., Suija, A., Taylor, D.L., Telleria, M.T., Weiss, M., Larsson, K.-H.,  
843 2013. Towards a unified paradigm for sequence-based identification of fungi. *Mol Ecol* 22,  
844 5271–5277. <https://doi.org/10.1111/mec.12481>
- 845 Krings, M., Taylor, T.N., Hass, H., Kerp, H., Dotzler, N., Hermsen, E.J., 2007. Fungal endophytes in a  
846 400-million-yr-old land plant: infection pathways, spatial distribution, and host responses. *New*  
847 *Phytologist* 174, 648–657. <https://doi.org/10.1111/j.1469-8137.2007.02008.x>
- 848 Lau, M.K., Arnold, A.E., Johnson, N.C., 2013. Factors influencing communities of foliar fungal  
849 endophytes in riparian woody plants. *Fungal Ecology* 6, 365–378.  
850 <https://doi.org/10.1016/j.funeco.2013.06.003>
- 851 Legendre, P., Fortin, M.J., 1989. Spatial pattern and ecological analysis. *Vegetatio* 80, 107–138.
- 852 Legendre, P., Gallagher, E.D., 2001. Ecologically meaningful transformations for ordination of species  
853 data. *Oecologia* 129, 271–280.
- 854 Legendre, P., Legendre, L., 2012. Numerical Ecology, Third. ed, *Developments in Environmental*  
855 *Modelling*. Elsevier.
- 856 Lemanceau, P., Blouin, M., Muller, D., Moënne-Loccoz, Y., 2017. Let the core microbiota be  
857 functional. *Trends in plant science* 22, 583–595.
- 858 Liggenstoffer, A.S., Youssef, N.H., Couger, M.B., Elshahed, M.S., 2010. Phylogenetic diversity and  
859 community structure of anaerobic gut fungi (phylum Neocallimastigomycota) in ruminant and  
860 non-ruminant herbivores. *The ISME Journal* 4, 1225–1235.  
861 <https://doi.org/10.1038/ismej.2010.49>
- 862 Lloyd-Price, J., Abu-Ali, G., Huttenhower, C., 2016. The healthy human microbiome. *Genome*  
863 *medicine* 8, 51.
- 864 Losos, E., Leigh, E.G. (Eds.), 2004. Tropical Forest Diversity and Dynamism. University of Chicago  
865 Press.
- 866 Love, M.I., Huber, W., Anders, S., 2014. Moderated estimation of fold change and dispersion for  
867 RNA-seq data with DESeq2. *Genome Biology* 15, 550. <https://doi.org/10.1186/s13059-014-0550-8>
- 868 Lundberg, D.S., Lebeis, S.L., Paredes, S.H., Yourstone, S., Gehring, J., Malfatti, S., Tremblay, J.,  
869 Engelbrektson, A., Kunin, V., Rio, T.G. del, Edgar, R.C., Eickhorst, T., Ley, R.E., Hugenholtz,  
870 P., Tringe, S.G., Dangl, J.L., 2012. Defining the core *Arabidopsis thaliana* root microbiome.  
871 *Nature* 488, 86–90. <https://doi.org/10.1038/nature11237>
- 872 Malloch, D.W., Pirozynski, K.A., Raven, P.H., 1980. Ecological and evolutionary significance of  
873 mycorrhizal symbioses in vascular plants (A Review). *PNAS* 77, 2113–2118.  
874 <https://doi.org/10.1073/pnas.77.4.2113>
- 875 Mantel, N., 1967. The Detection of Disease Clustering and a Generalized Regression Approach. *Cancer*  
876 *Res* 27, 209–220.
- 877 Martiny, J.B.H., Bohannan, B.J.M., Brown, J.H., Colwell, R.K., Fuhrman, J.A., Green, J.L., Horner-  
878 Devine, M.C., Kane, M., Krumins, J.A., Kuske, C.R., Morin, P.J., Naeem, S., Øvreås, L.,  
879 Reysenbach, A.-L., Smith, V.H., Staley, J.T., 2006. Microbial biogeography: putting  
880 microorganisms on the map. *Nature Reviews Microbiology* 4, 102–112.  
881 <https://doi.org/10.1038/nrmicro1341>
- 882 McCune, B., Grace, J.B., Urban, D.L., 2002. Analysis of ecological communities. MjM software  
883 design Gleneden Beach, OR.
- 884 McMurdie, P.J., Holmes, S., 2013. phyloseq: An R Package for Reproducible Interactive Analysis and  
885 Graphics of Microbiome Census Data. *PLOS ONE* 8, e61217.  
886 <https://doi.org/10.1371/journal.pone.0061217>
- 887 Mejía, L.C., Kyllo, D.A., Rojas, E., Maynard, Z., Butler, A., Van Bael, S.A., Herre, E., 2007.  
888 Ecological Implications of Anti-Pathogen Effects of Tropical Fungal Endophytes and  
889 Mycorrhizae. *Ecology* 88, 550–558. <https://doi.org/10.1890/05-1606>
- 890  
891  
892  
893  
894  
895  
896

- 897  
898 Mejía, L.C., Rojas, E.I., Maynard, Z., Bael, S.V., Arnold, A.E., Hebar, P., Samuels, G.J., Robbins, N.,  
899 Herre, E.A., 2008. Endophytic fungi as biocontrol agents of *Theobroma cacao* pathogens.  
900 Biological Control, Special Issue: Endophytes 46, 4–14.  
901 https://doi.org/10.1016/j.biocontrol.2008.01.012  
902  
903 Mumme, D.L., Rillig, M.C., 2008. Spatial characterization of arbuscular mycorrhizal fungal  
904 molecular diversity at the submetre scale in a temperate grassland. FEMS Microbiol Ecol 64,  
905 260–270. https://doi.org/10.1111/j.1574-6941.2008.00475.x  
906 Nemergut, D.R., Schmidt, S.K., Fukami, T., O'Neill, S.P., Bilinski, T.M., Stanish, L.F., Knelman, J.E.,  
907 Darcy, J.L., Lynch, R.C., Wickey, P., Ferrenberg, S., 2013. Patterns and Processes of Microbial  
908 Community Assembly. *Microbiology and Molecular Biology Reviews* 77, 342–356.  
909 https://doi.org/10.1128/MMBR.00051-12  
910 Norros, V., Penttilä, R., Suominen, M., Ovaskainen, O., 2012. Dispersal may limit the occurrence of  
911 specialist wood decay fungi already at small spatial scales. *Oikos* 121, 961–974.  
912 https://doi.org/10.1111/j.1600-0706.2012.20052.x  
913 Oksanen, J., Guillaume Blanchet, F., Friendly, M., Kindt, R., Legendre, P., McGlinn, D., Minchin,  
914 P.R., O'Hara, R.B., Simpson, G.L., Solymos, P., 2017. M. Stevens MHH, Szoecs E, Wagner H  
915 (2017). vegan: Community Ecology Package. R package version 2.4-3.  
916 Oldroyd, G.E.D., 2013. Speak, friend, and enter: signalling systems that promote beneficial symbiotic  
917 associations in plants. *Nature Reviews Microbiology* 11, 252–263.  
918 https://doi.org/10.1038/nrmicro2990  
919 Peay, K.G., Garbelotto, M., Bruns, T.D., 2010. Evidence of dispersal limitation in soil microorganisms:  
920 Isolation reduces species richness on mycorrhizal tree islands. *Ecology* 91, 3631–3640.  
921 https://doi.org/10.1890/09-2237.1  
922 Peršoh, D., 2013. Factors shaping community structure of endophytic fungi—evidence from the *Pinus-*  
923 *Viscum*-system. *Fungal Diversity* 60, 55–69. https://doi.org/10.1007/s13225-013-0225-x  
924 Porras-Alfaro, A., Bayman, P., 2011. Hidden Fungi, Emergent Properties: Endophytes and  
925 Microbiomes. *Annual Review of Phytopathology* 49, 291–315. https://doi.org/10.1146/annurev-  
926 phyto-080508-081831  
927 R Core Team, 2017. R: A language and environment for statistical computing. R Foundation for  
928 Statistical Computing, Vienna, Austria.  
929 Redecker, D., Kodner, R., Graham, L.E., 2000. Glomalean Fungi from the Ordovician. *Science* 289,  
930 1920–1921. https://doi.org/10.1126/science.289.5486.1920  
931 Rodrigues, J.L.M., Pellizari, V.H., Mueller, R., Baek, K., Jesus, E. da C., Paula, F.S., Mirza, B.,  
932 Hamaoui, G.S., Tsai, S.M., Feigl, B., Tiedje, J.M., Bohannan, B.J.M., Nüsslein, K., 2013.  
933 Conversion of the Amazon rainforest to agriculture results in biotic homogenization of soil  
934 bacterial communities. *PNAS* 110, 988–993. https://doi.org/10.1073/pnas.1220608110  
935 Rodriguez, R., Redman, R., 2008. More than 400 million years of evolution and some plants still can't  
936 make it on their own: plant stress tolerance via fungal symbiosis. *J Exp Bot* 59, 1109–1114.  
937 https://doi.org/10.1093/jxb/erm342  
938 Rodriguez, R.J., White Jr, J.F., Arnold, A.E., Redman, R.S., 2009. Fungal endophytes: diversity and  
939 functional roles. *New Phytologist* 182, 314–330. https://doi.org/10.1111/j.1469-  
940 8137.2009.02773.x  
941 Rojas, E.I., Rehner, S.A., Samuels, G.J., Bael, S.A.V., Herre, E.A., Cannon, P., Chen, R., Pang, J.,  
942 Wang, R., Zhang, Y., Peng, Y.-Q., Sha, T., 2010. *Colletotrichum gloeosporioides* s.l. associated  
943 with *Theobroma cacao* and other plants in Panamá: multilocus phylogenies distinguish host-  
944 associated pathogens from asymptomatic endophytes. *Mycologia* 102, 1318–1338.  
945 https://doi.org/10.3852/09-244  
946 Rosenberg, E., Sharon, G., Atad, I., Zilber-Rosenberg, I., 2010. The evolution of animals and plants via  
947 symbiosis with microorganisms. *Environmental Microbiology Reports* 2, 500–506.  
948  
949  
950  
951  
952

- 953  
954 https://doi.org/10.1111/j.1758-2229.2010.00177.x  
955 Santamaría, J., Bayman, P., 2005. Fungal Epiphytes and Endophytes of Coffee Leaves (<Emphasis  
956 Type="Italic">Coffea arabica</Emphasis>). *Microb Ecol* 50, 1–8.  
957 https://doi.org/10.1007/s00248-004-0002-1  
958 Schulz, B., Römmert, A.-K., Dammann, U., Aust, H.-J., Strack, D., 1999. The endophyte-host  
959 interaction: a balanced antagonism? *Mycological Research* 103, 1275–1283.  
960 Shade, A., Handelsman, J., 2012. Beyond the Venn diagram: the hunt for a core microbiome.  
961 *Environmental Microbiology* 14, 4–12. https://doi.org/10.1111/j.1462-2920.2011.02585.x  
962 Stukenbrock, E.H., McDonald, B.A., 2008. The Origins of Plant Pathogens in Agro-Ecosystems.  
963 *Annual Review of Phytopathology* 46, 75–100.  
964 https://doi.org/10.1146/annurev.phyto.010708.154114  
965 Su, S., Hsieh, C., Chang-Yang, C., Lu, C., Guan, B.T., 2010. Micro-topographic differentiation of the  
966 tree species composition in a subtropical submontane rainforest in northeastern Taiwan. *Taiwan  
967 Journal of Forest Science* 25, 63–80.  
968 Su, S.-H., Chang-Yang, C.H., Lu, C.L., Tsui, C.C., Lin, T.T., Lin, C.L., Chiou, W.L., Kuan, L.H.,  
969 Chen, Z.S., Hsieh, C.F., 2007. Fushan subtropical forest dynamics plot: tree species  
970 characteristics and distribution patterns. *Taiwan Forestry Research Institute*.  
971 Tateno, O., Hirose, D., Osono, T., Takeda, H., 2015. Beech cupules share endophytic fungi with leaves  
972 and twigs. *Mycoscience* 56, 252–256. https://doi.org/10.1016/j.myc.2014.07.005  
973 Tedersoo, L., Bahram, M., Põlme, S., Kõljalg, U., Yorou, N.S., Wijesundera, R., Ruiz, L.V., Vasco-  
974 Palacios, A.M., Thu, P.Q., Suija, A., Smith, M.E., Sharp, C., Saluveer, E., Saitta, A., Rosas, M.,  
975 Riit, T., Ratkowsky, D., Pritsch, K., Põldmaa, K., Piepenbring, M., Phosri, C., Peterson, M.,  
976 Parts, K., Pärtel, K., Otsing, E., Nouhra, E., Njouonkou, A.L., Nilsson, R.H., Morgado, L.N.,  
977 Mayor, J., May, T.W., Majuakim, L., Lodge, D.J., Lee, S.S., Larsson, K.-H., Kohout, P.,  
978 Hosaka, K., Hiiesalu, I., Henkel, T.W., Harend, H., Guo, L., Greslebin, A., Grelet, G., Geml, J.,  
979 Gates, G., Dunstan, W., Dunk, C., Drenkhan, R., Dearnaley, J., Kesel, A.D., Dang, T., Chen,  
980 X., Buegger, F., Brearley, F.Q., Bonito, G., Anslan, S., Abell, S., Abarenkov, K., 2014. Global  
981 diversity and geography of soil fungi. *Science* 346, 1256688.  
982 https://doi.org/10.1126/science.1256688  
983 The Human Microbiome Project Consortium, 2012. Structure, Function and Diversity of the Healthy  
984 Human Microbiome. *Nature* 486, 207–214. https://doi.org/10.1038/nature11234  
985 Thomas, D., Bailes, G., Vandegrift, R., Roy, B.A., 2017. Understanding and mitigating some  
986 limitations of Illumina © MiSeq for environmental sequencing of fungi. *bioRxiv*.  
987 Van Der Heijden, M.G.A., Horton, T.R., 2009. Socialism in soil? The importance of mycorrhizal  
988 fungal networks for facilitation in natural ecosystems. *Journal of Ecology* 97, 1139–1150.  
989 https://doi.org/10.1111/j.1365-2745.2009.01570.x  
990 Vandegrift, A.W.R., 2016. Ecological Roles of Fungal Endophytes. University of Oregon.  
991 Vandenkoornhuyse, P., Quaiser, A., Duhamel, M., Le Van, A., Dufresne, A., 2015. The importance of  
992 the microbiome of the plant holobiont. *New Phytol* 206, 1196–1206.  
993 https://doi.org/10.1111/nph.13312  
994 Veech, J.A., 2013. A probabilistic model for analysing species co-occurrence. *Global Ecology and  
995 Biogeography* 22, 252–260. https://doi.org/10.1111/j.1466-8238.2012.00789.x  
996 Vellend, M., 2010. Conceptual Synthesis in Community Ecology. *The Quarterly Review of Biology*  
997 85, 183–206. https://doi.org/10.1086/652373  
998 Venkateshwaran, M., Volkening, J.D., Sussman, M.R., Ané, J.-M., 2013. Symbiosis and the social  
999 network of higher plants. *Current Opinion in Plant Biology, Growth and development* 16, 118–  
1000 127. https://doi.org/10.1016/j.pbi.2012.11.007  
1001 Wilson, D., 1995. Endophyte: The Evolution of a Term, and Clarification of Its Use and Definition.  
1002 *Oikos* 73, 274–276. https://doi.org/10.2307/3545919  
1003  
1004  
1005  
1006  
1007  
1008

- 1009  
1010 Woodward, C., Hansen, L., Beckwith, F., Redman, R.S., Rodriguez, R.J., 2012. Symbiogenics: An  
1011 Epigenetic Approach to Mitigating Impacts of Climate Change on Plants. HortScience 47, 699–  
1012 703.  
1013 Wright, S., 1940. Breeding Structure of Populations in Relation to Speciation. The American Naturalist  
1014 74, 232–248. <https://doi.org/10.1086/280891>  
1015 Zimmerman, N.B., Vitousek, P.M., 2012. Fungal endophyte communities reflect environmental  
1016 structuring across a Hawaiian landscape. PNAS 109, 13022–13027.  
1017 <https://doi.org/10.1073/pnas.1209872109>

1019  
1020 **Captions**

1021 (A)  
1022 Figure 1. Left: topographic map of the Fushan FDP with the four vegetation types as classified by Su et  
1023 al. (2007). Right: map of the habitat type, a composite classification based on microtopographic characteristics of  
1024 quadrats, defined by Su et al. (2010). The units of the coordinates and contours are in meters, with quadrats at  
1025 20x20m scale. Figures reproduced with permission from authors.

1026 (A)  
1027 Figure 2. Left - An overview of nested-squares, logarithmic sampling scheme Vandegrift (2016). Vertices of  
1028 squares are sample sites. Units are meters. Right: Perspective diagram of Fushan Forest Dynamics Plot (Su et  
1029 al., 2010). Figures reproduced with permission from authors.

1030 (C)  
1031 Figure 3. Class-level overview of taxonomic composition of wood and leaf libraries, from all trees sampled.  
1032 Proportions are out of total OTUs observed in each plant organ: 2025 OTUs in wood, 1302 in leaves.

1033 Figure 4. Non-metric multidimensional scaling diagram of endophyte communities, with all tree hosts that were  
1034 sampled at least 3 times.

1035 English  
1036 English  
1037 English  
1038 English  
1039 English  
1040 English  
1041 English  
1042 English  
1043 English  
1044 English  
1045 English  
1046 English  
1047 English  
1048 English  
1049 English  
1050 English  
1051 English  
1052 English  
1053 English  
1054 English  
1055 English  
1056 English  
1057 English  
1058 English  
1059 English  
1060 English  
1061 English  
1062 English  
1063 English  
1064 English  
Figure 5. Two PCNM vectors showing patterns of variation in endophyte communities of a single host-tree species, *Helicia formosana*. Size and color of bubble show differences in community predicted by this spatial pattern. For some significant subset of the total endophytes species sampled, sites with large black circles contain very different fungal species assemblages than sites with large white circles. Here, both leaf and wood endophyte communities display dissimilarity between the plot at large and the southern valley. Full arrays of ecologically significant PCNM vectors are provided in Supp Figs. 1 and 2.

Figure 6. Map of Bray-Curtis dissimilarity values over the Fushan FDP, resulting from comparisons between red circled point all other *Helicia formosana* samples. Dark blue points (BC=1) share no fungal species in common with the circled sample, and increase in similarity from yellow to green (BC=0).

Figure 7. Map of Bray-Curtis dissimilarity values over the Fushan FDP, resulting from comparisons between all *H. formosana* points and the core fungi of the *H. formosana*. Dark blue points (BC=1) contain no species from this set of core fungi, and increase in similarity from yellow to green (BC=0, 100% of core fungi present).

Figure 8. Distribution of Bray-Curtis dissimilarity among sample comparisons of all hosts, and of *Helicia formosana* only.

You need  
to get permission  
from the copyright  
holder who is  
probably the journal.  
Please send a  
copy of  
the paper.

1065  
1066  
1067  
1068

## Supplementary figures

Please remove from main text and put in supplement figures

1069 Supplemental Figure 1 . Mantel correlogram, showing standardized Mantel statistics of distance classes  
1070 between Bray-Curtis dissimilarity matrix from leaf and wood endophyte communities and distances among  
1071 sampling sites. Positive Mantel's r for a given set of comparisons indicates positive autocorrelation among a  
1072 comparisons of this distance. Black filled circles indicate statistically significant correlations, after correction for  
1073 multiple testing.

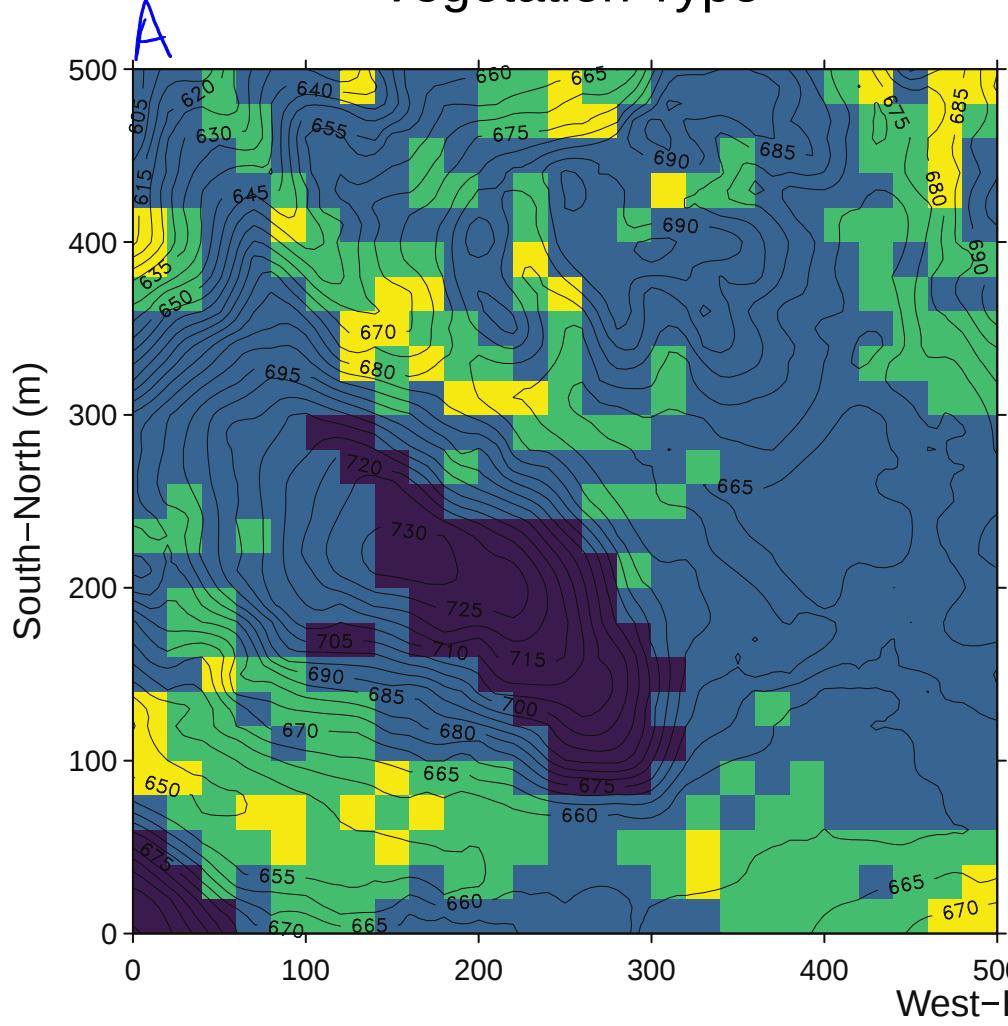
1074 Supplemental Figure 2. All-host leaf dbMEM analysis. All vectors shown here correlate to some degree with  
1075 spatial patterns found within leaf endophyte populations. See methods section for further details on interpretation  
1076 of dbMEM vectors and their interpretation.

1077 Supplemental Figure 3. All-host wood dbMEM analysis. All vectors shown here correlate to some degree with  
1078 spatial patterns found within leaf endophyte populations. See methods section for further details on interpretation  
1079 of dbMEM vectors and their interpretation.

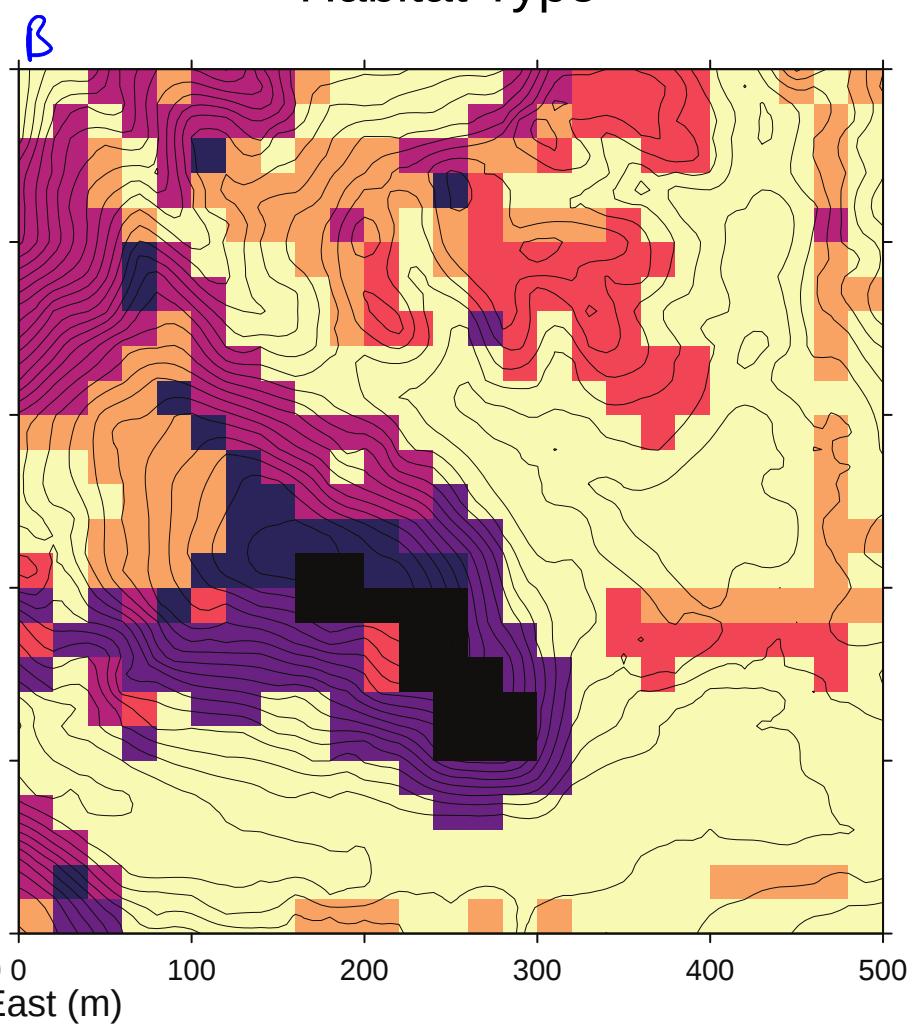
1080 Supplemental Figure 4. Co-occurrence networks of fungal endophytes and their tree hosts. Closest (first-degree)  
1081 associations were used to select fungi as candidates for the core microbiome of a host.  
1082  
1083  
1084  
1085  
1086  
1087  
1088  
1089  
1090  
1091  
1092  
1093  
1094  
1095  
1096  
1097  
1098  
1099  
1100  
1101  
1102  
1103  
1104  
1105  
1106  
1107  
1108  
1109  
1110  
1111  
1112  
1113  
1114  
1115  
1116  
1117  
1118  
1119  
1120

Below is some text which is part of manuscript  
and upload components

## Vegetation Type

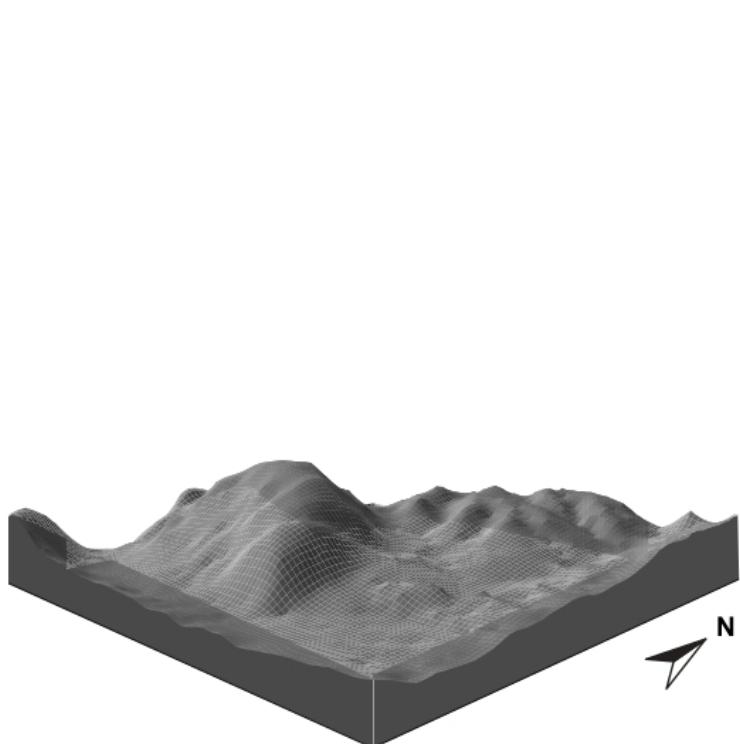
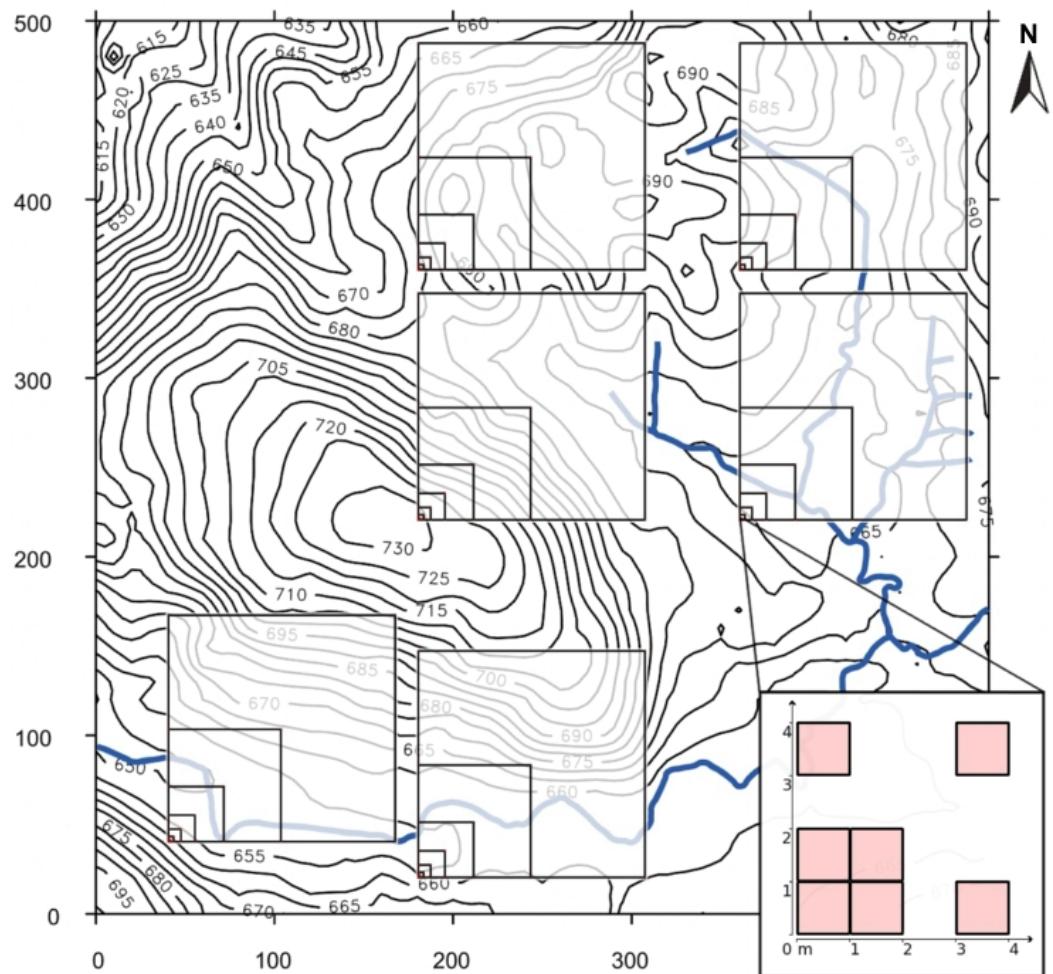


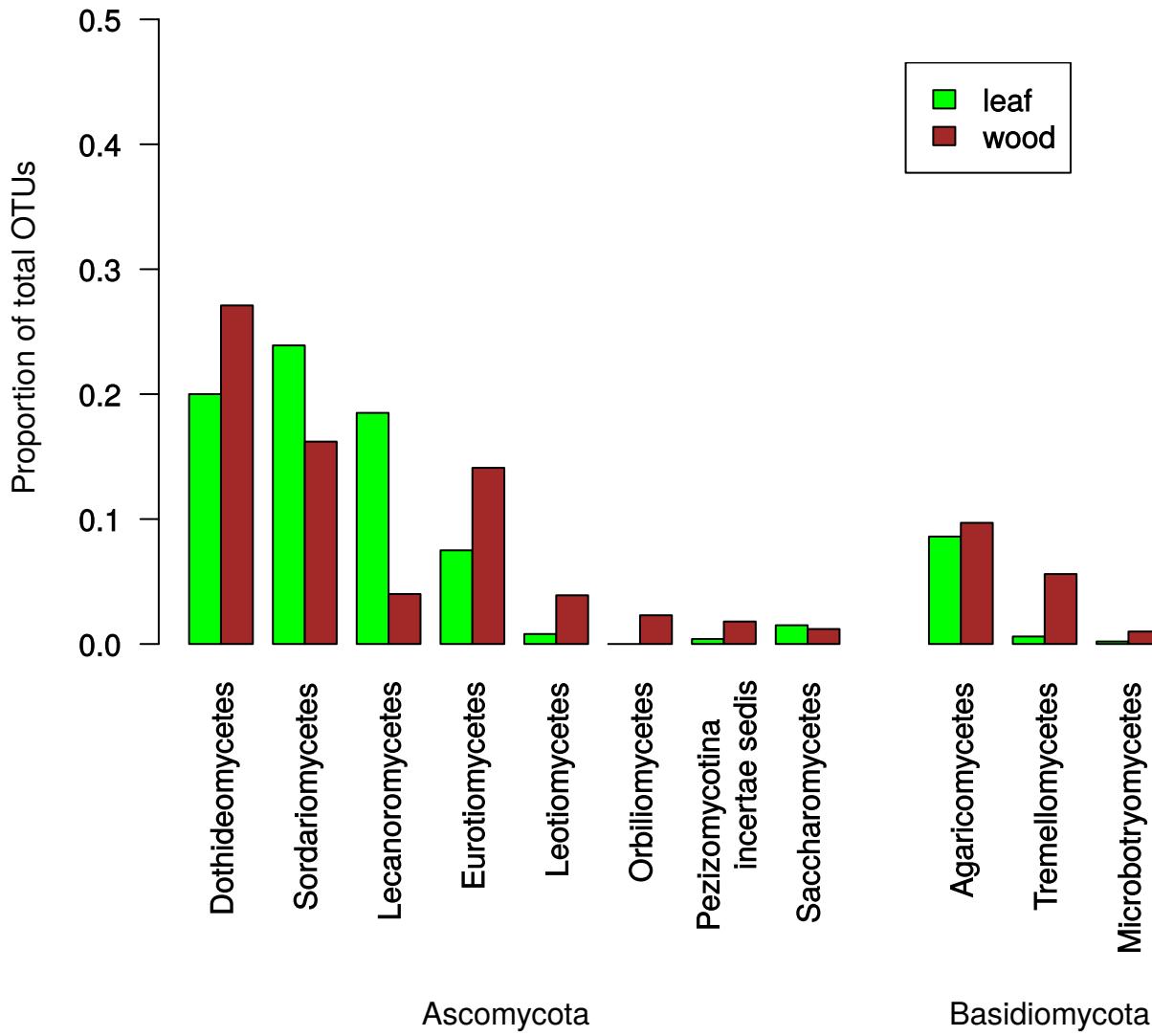
## Habitat Type



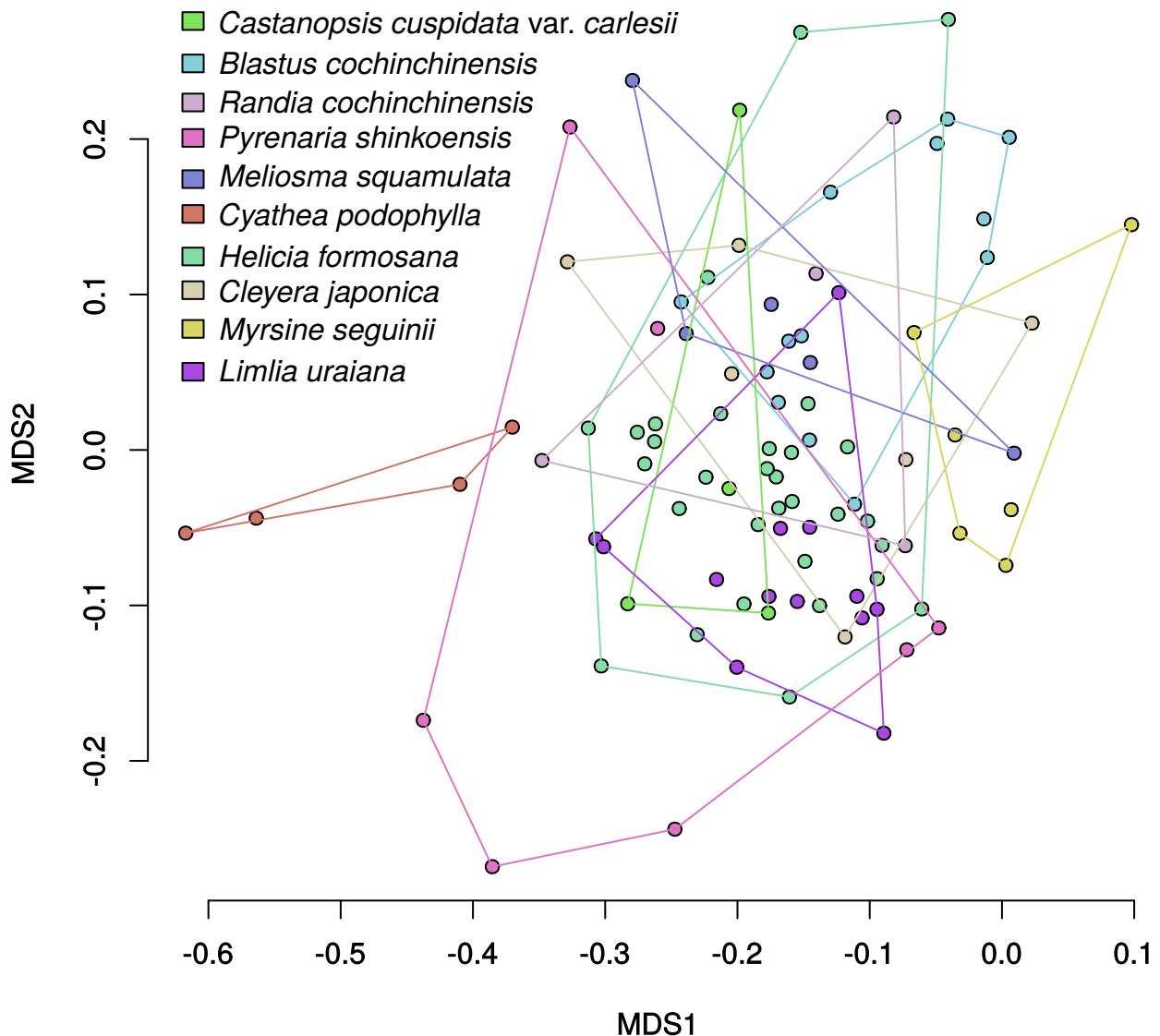
Myrsine sequinii type	Helicia formosana-Limlia uraiana type
Limlia uraiana type	Helicia formosana-Maes perlarius type

Type 1	Type 3	Type 5	Type 7
Type 2	Type 4	Type 6	

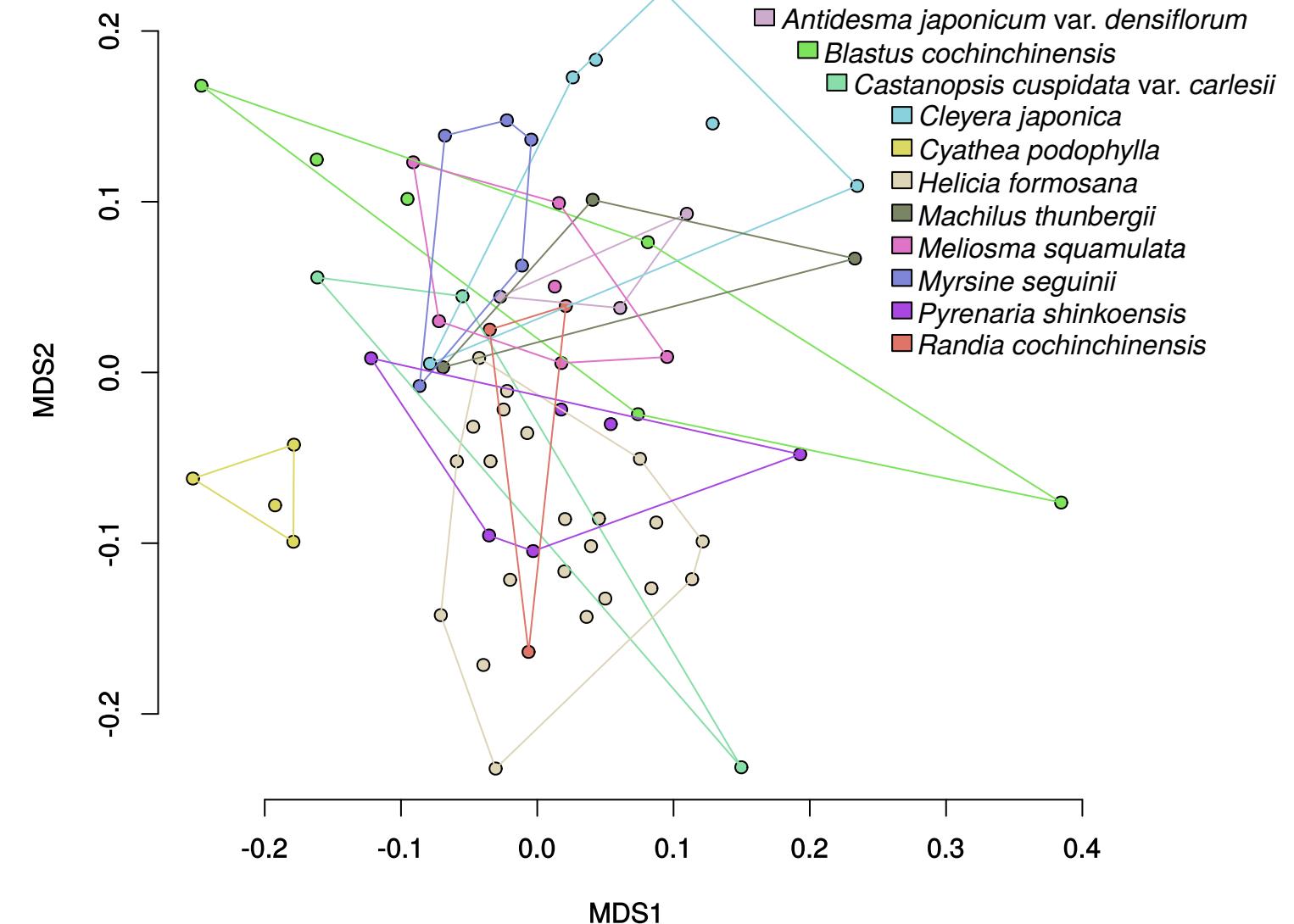




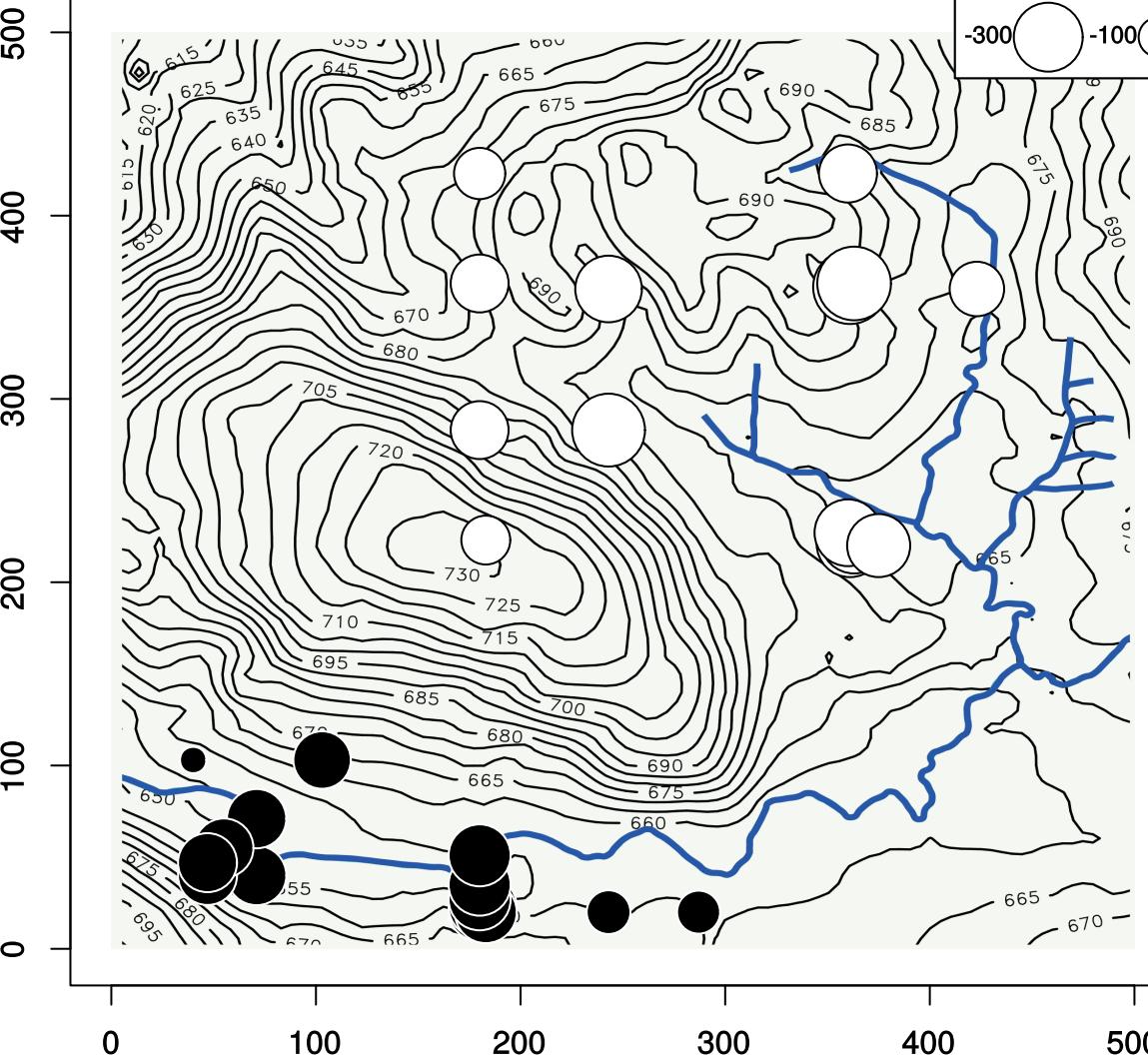
# Leaves



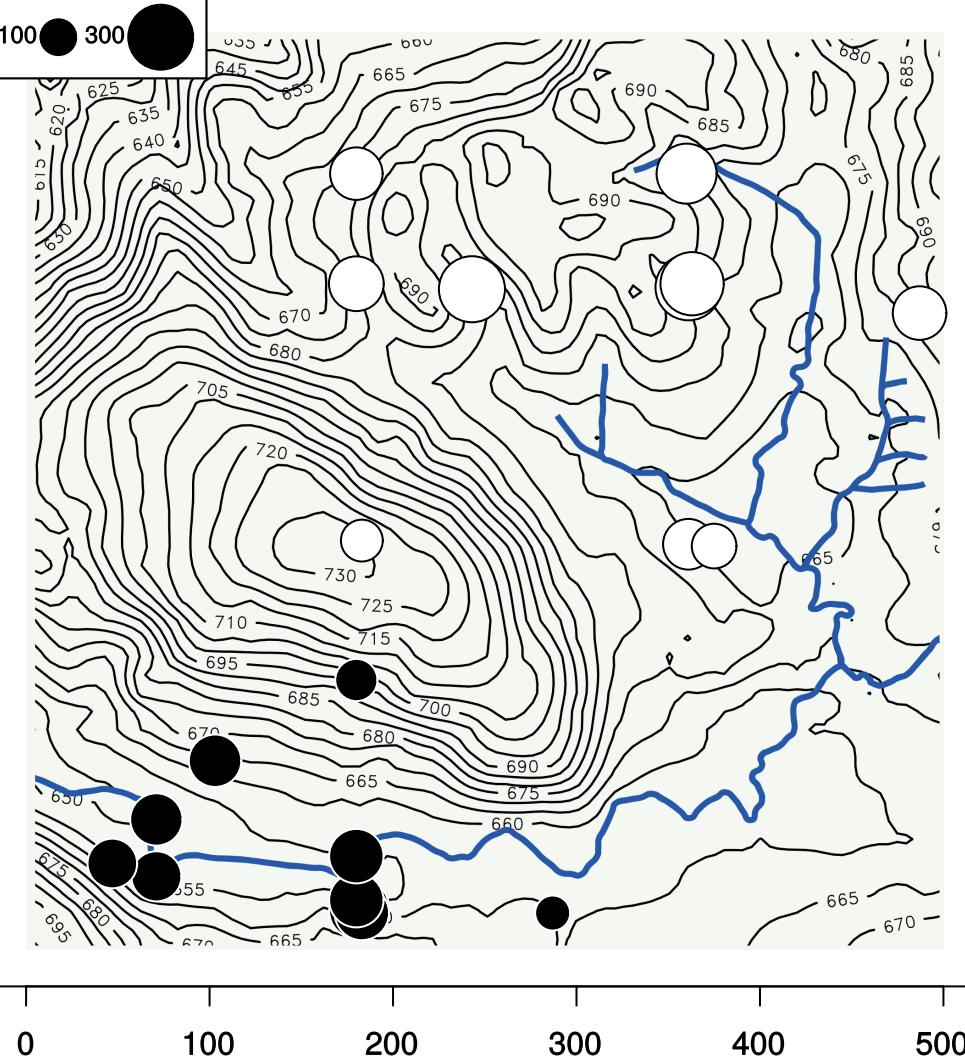
# Wood



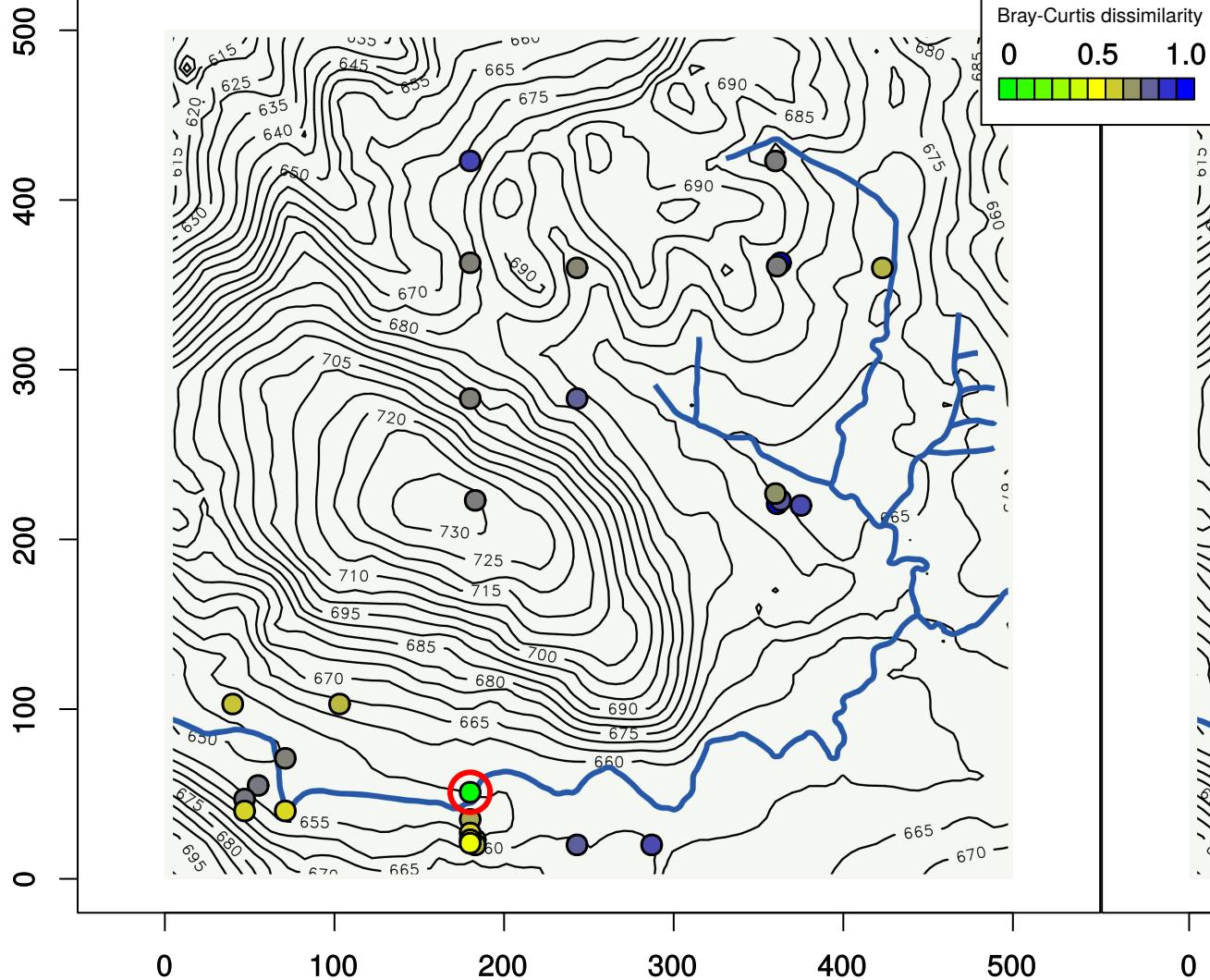
## Leaves



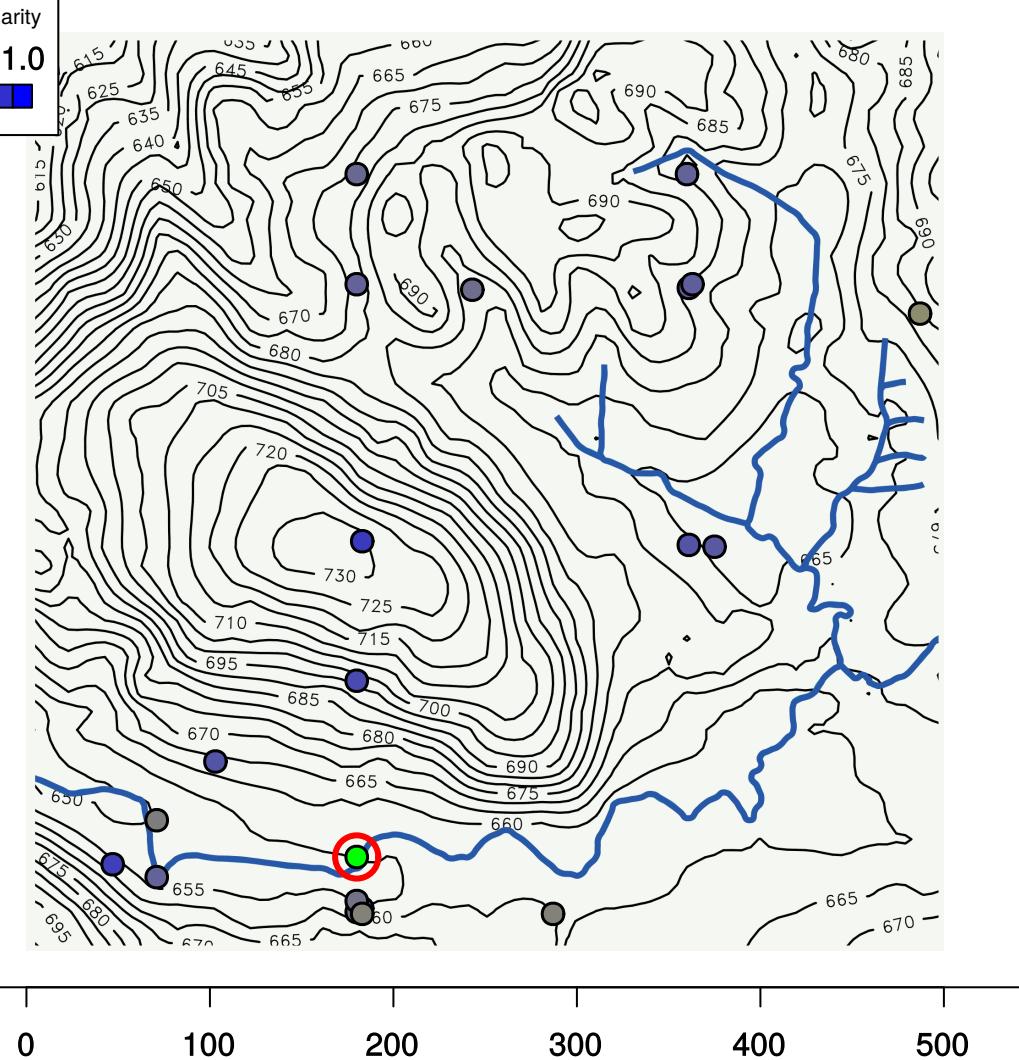
# Wood



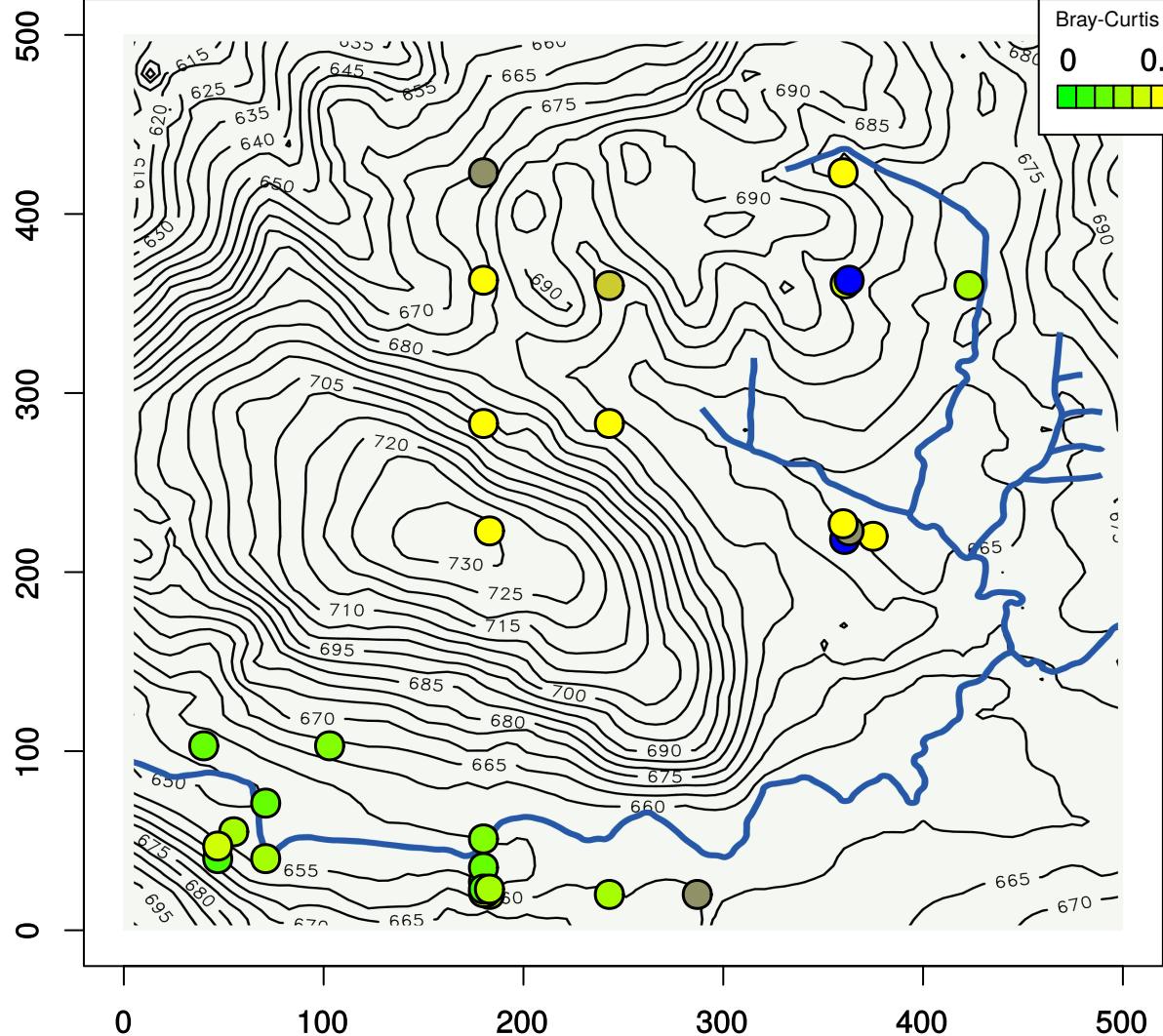
# Leaves



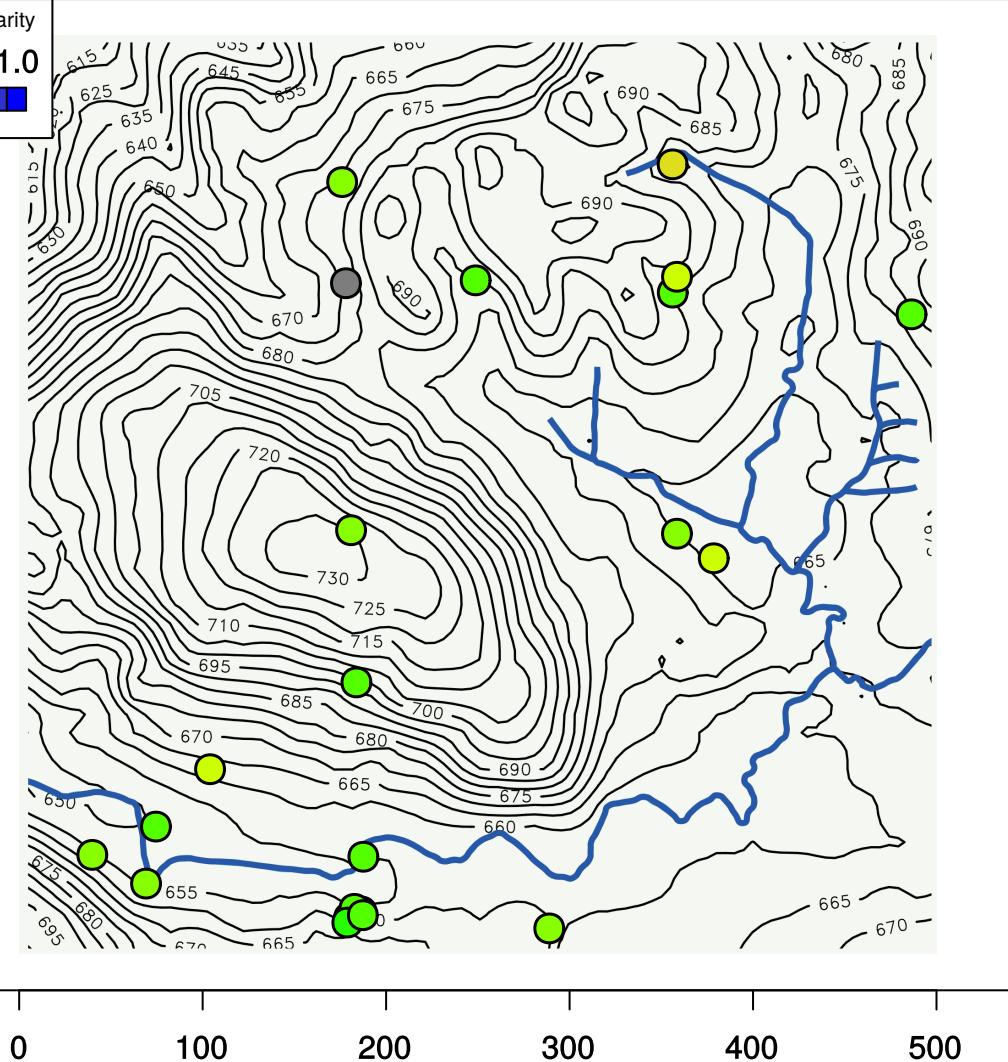
# Wood

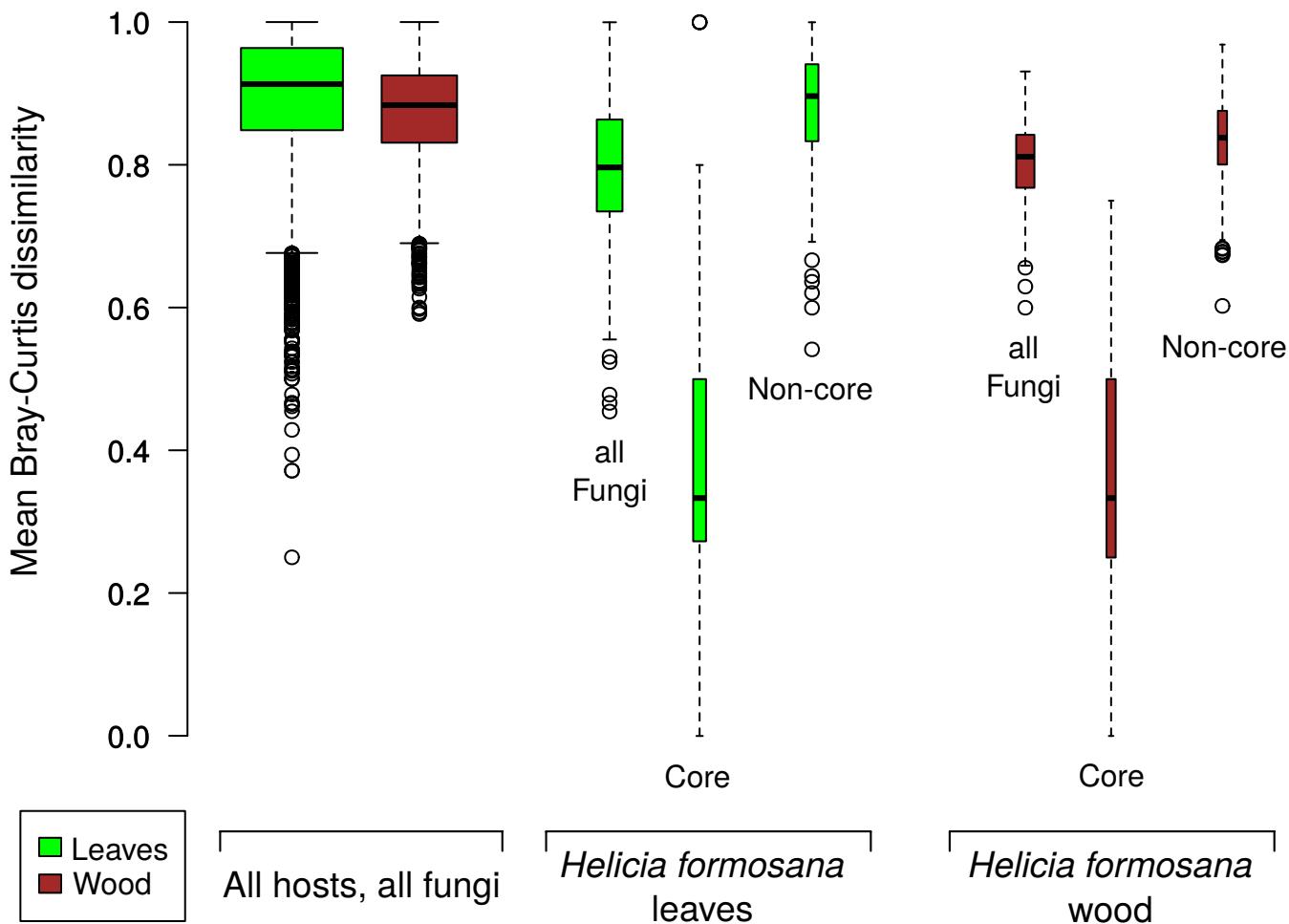


# Leaves

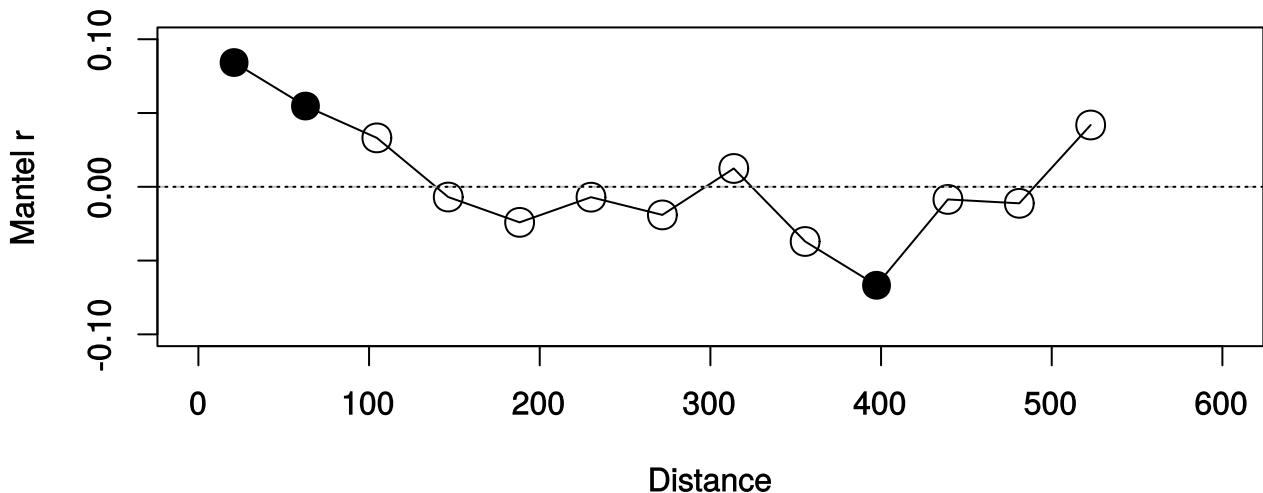


# Wood

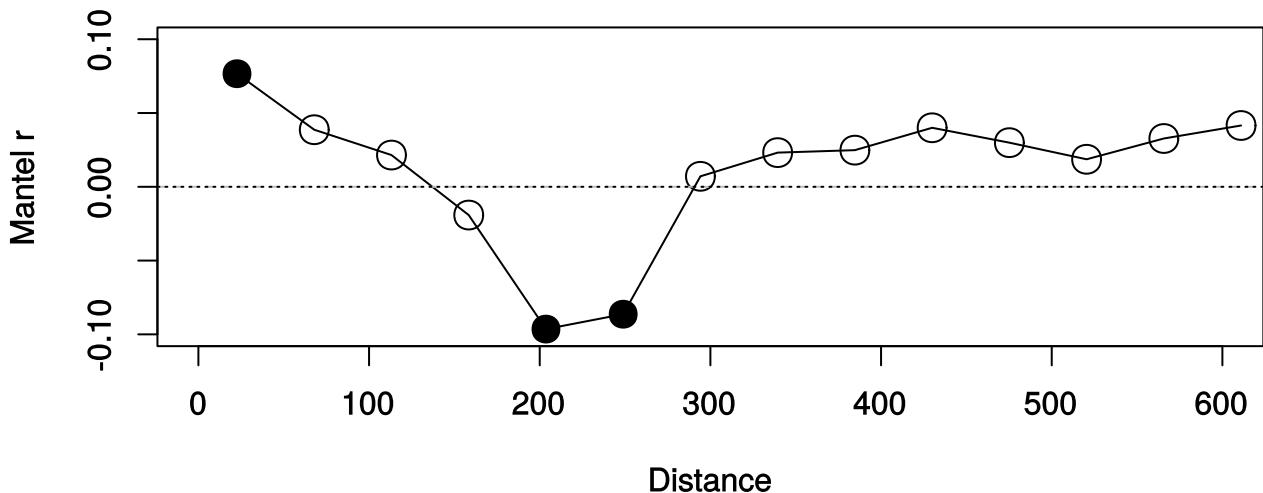




## Wood endophytes

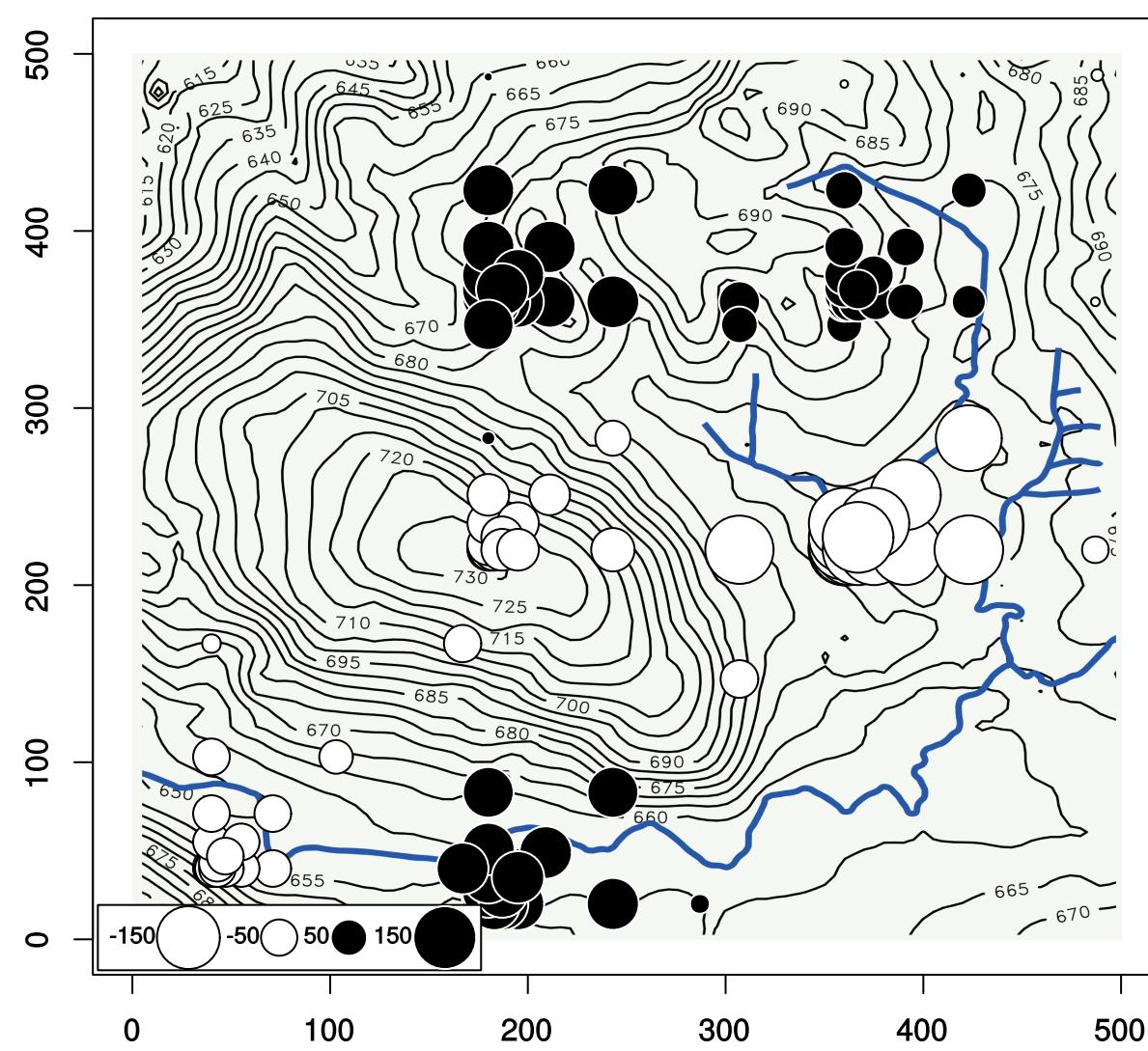


## Leaf endophytes

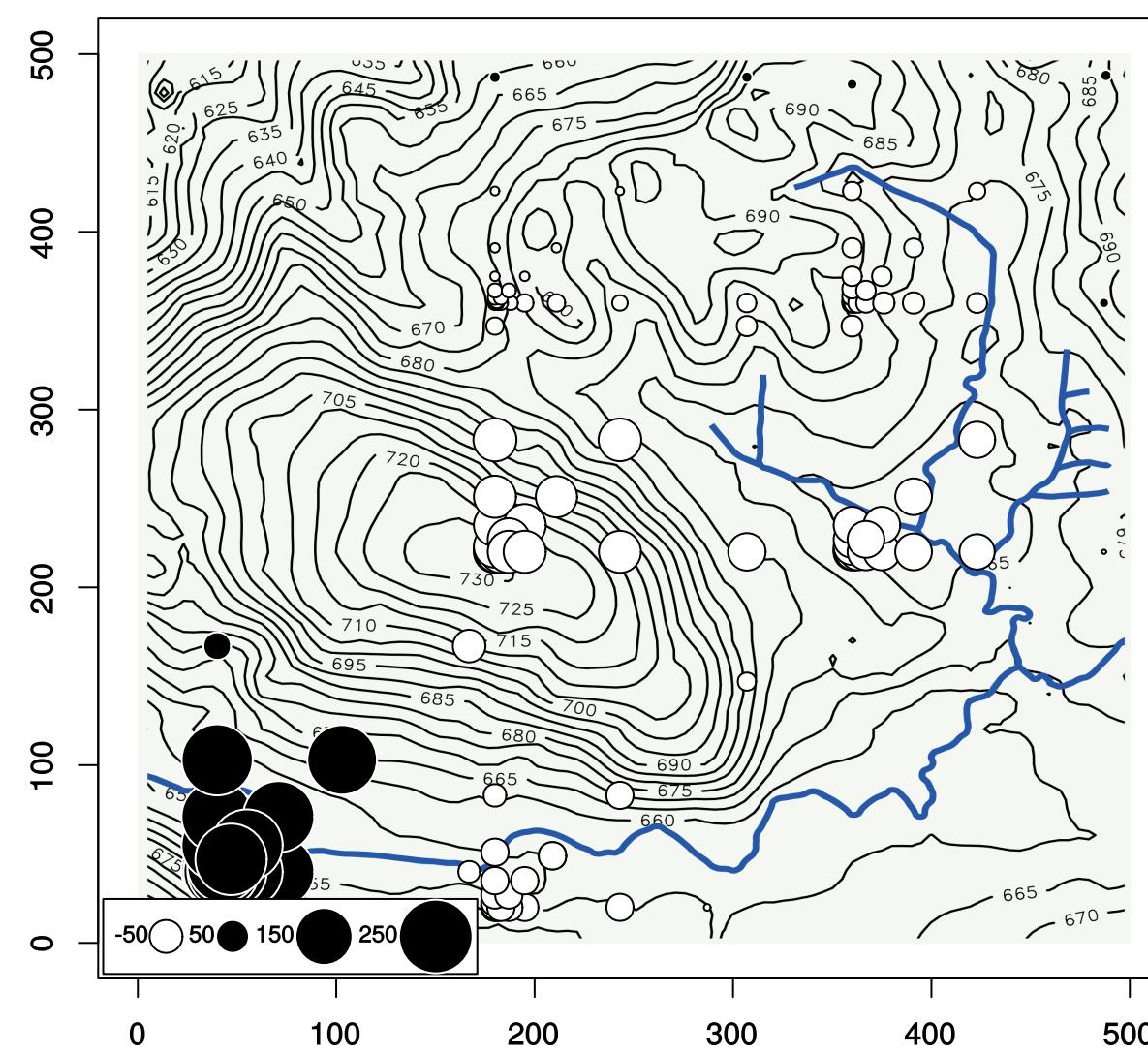


# All-host, leaf PCNM vectors

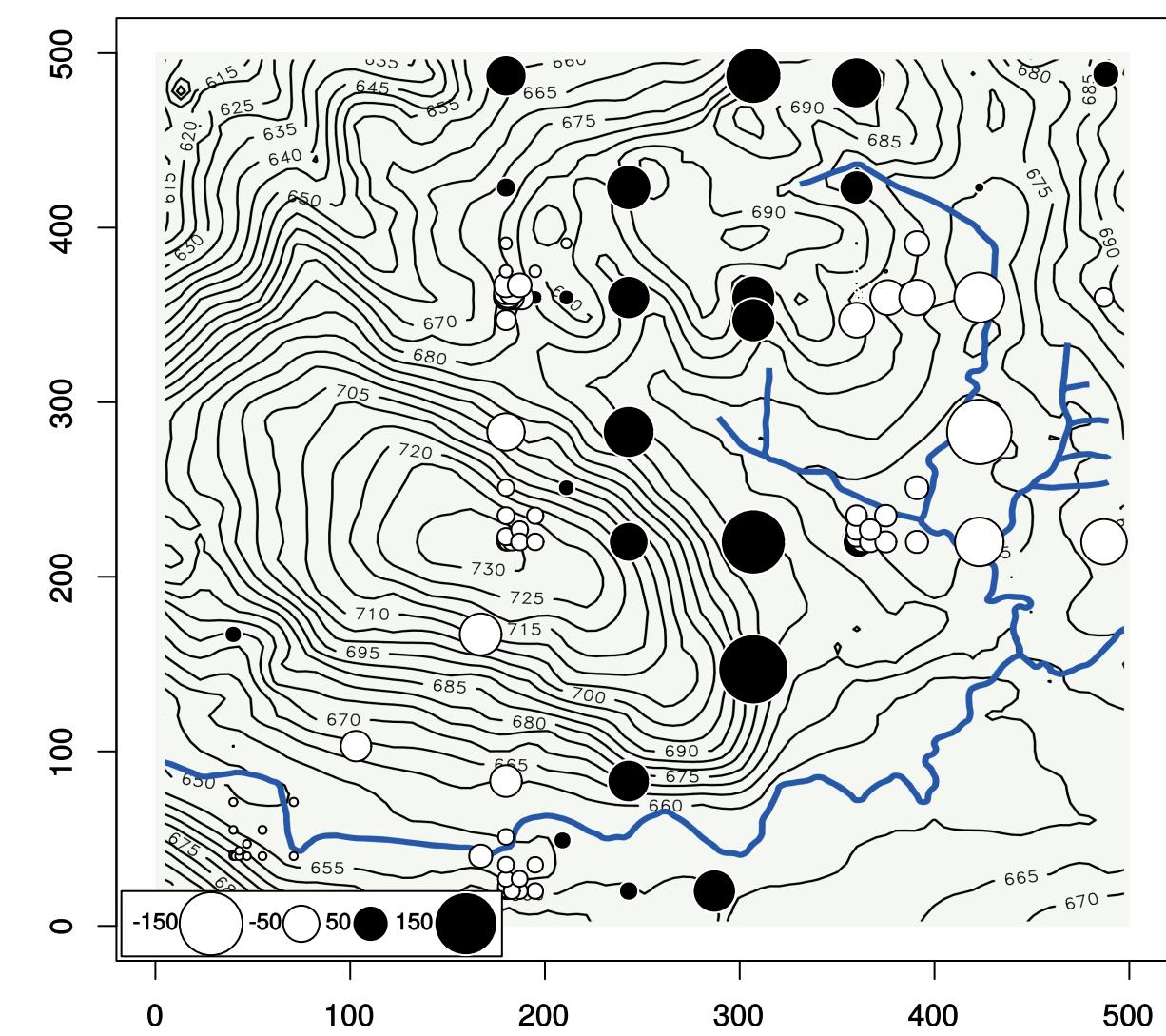
PCNM X3



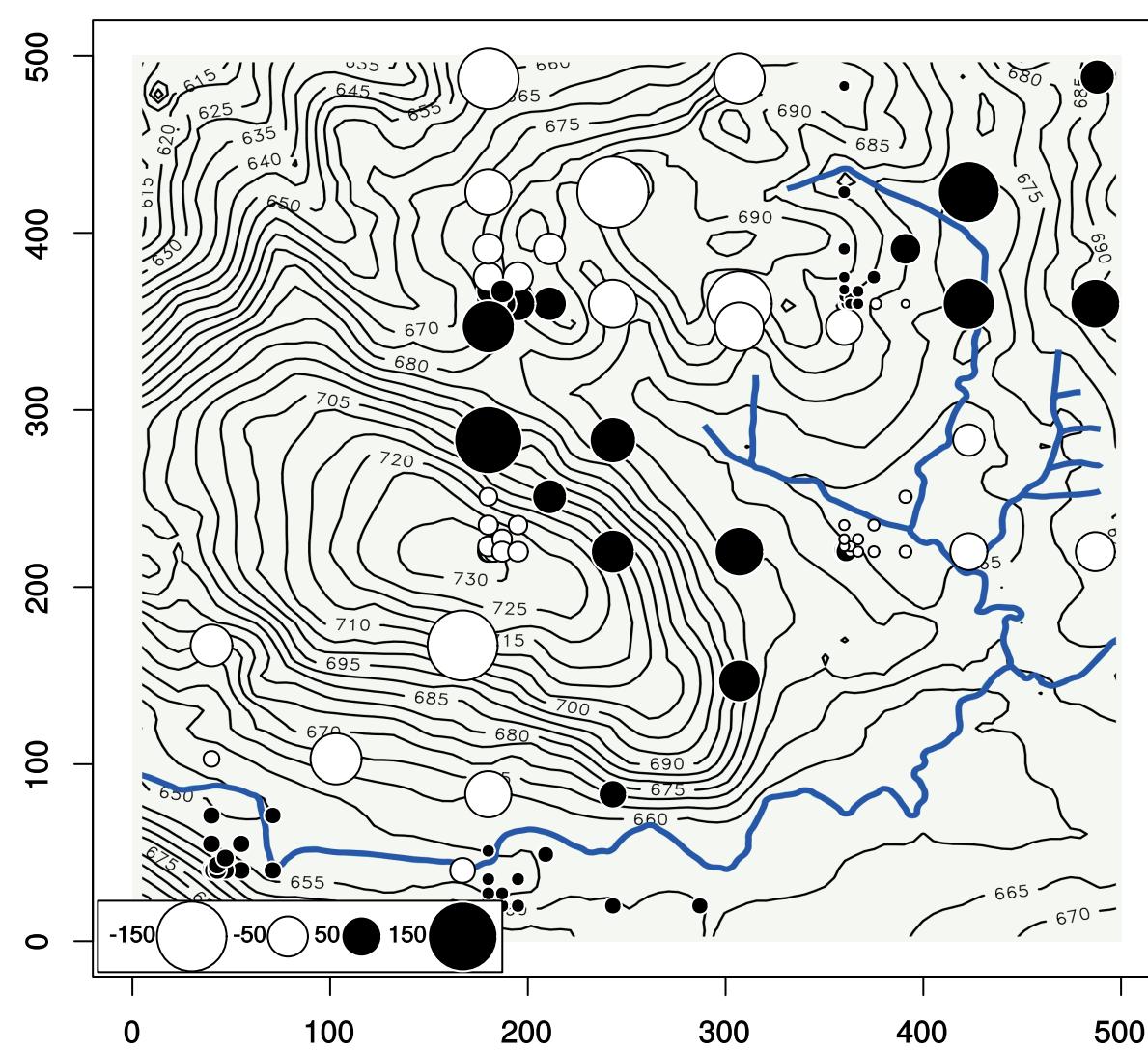
PCNM X5



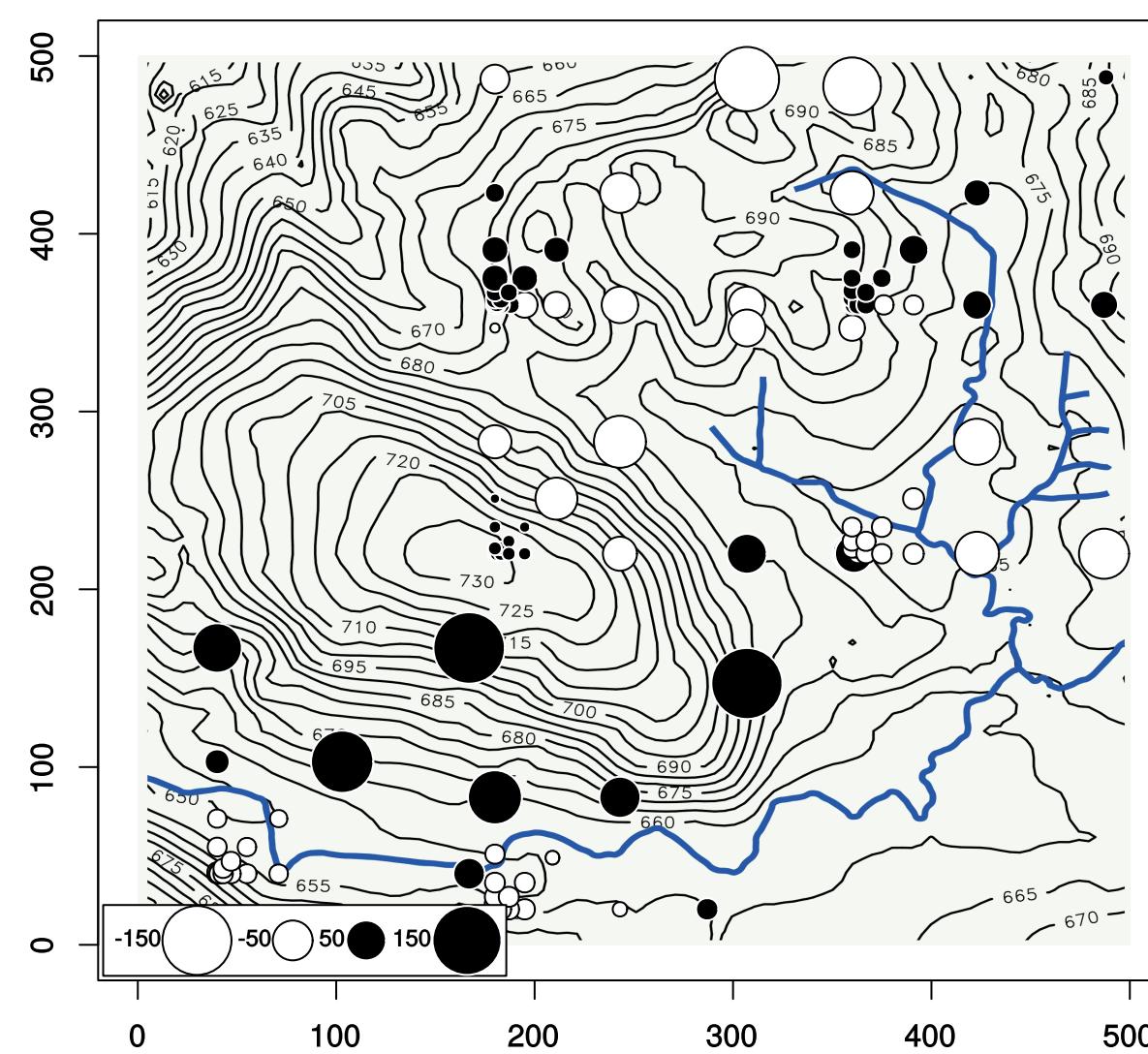
PCNM X7



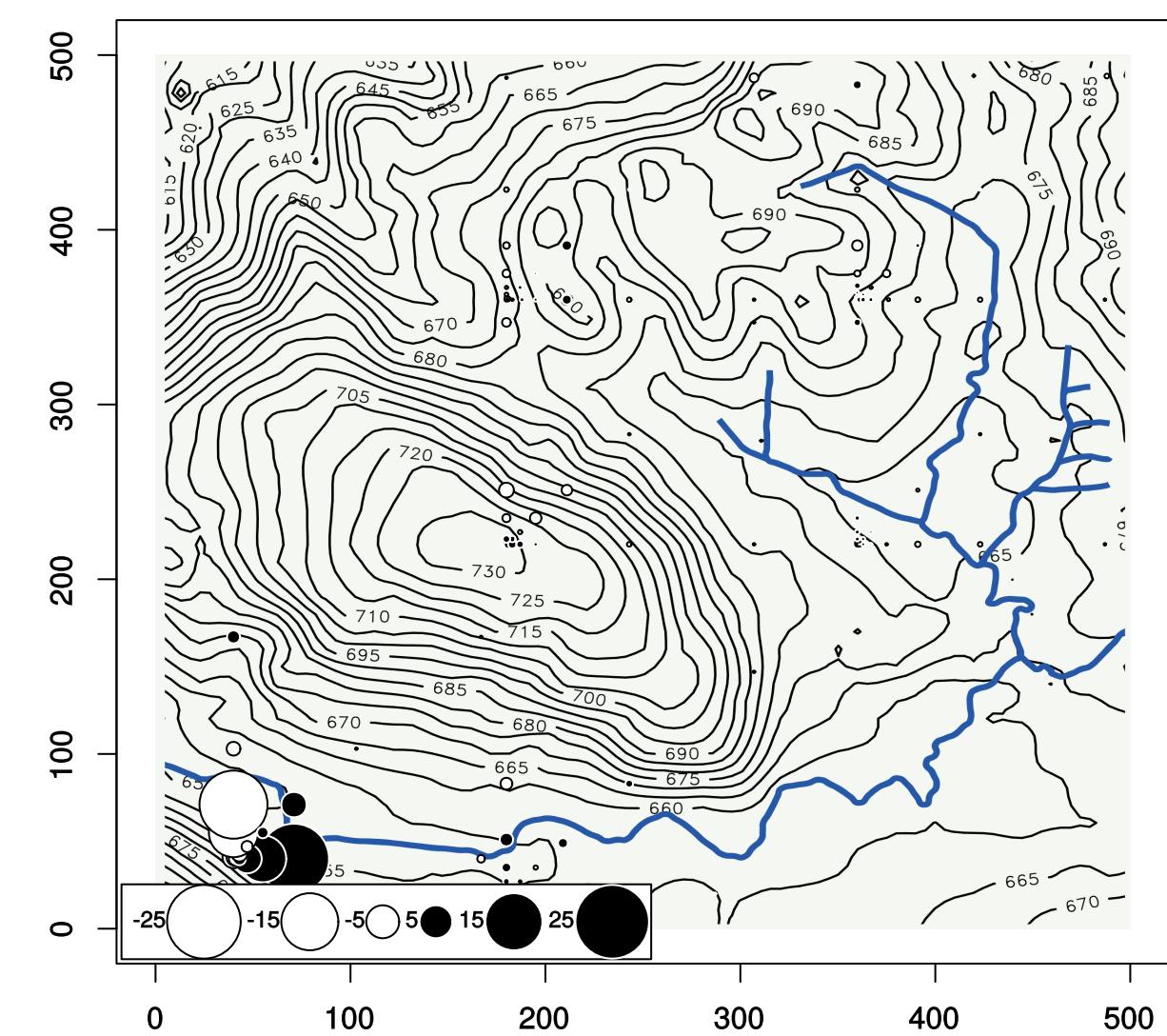
PCNM X8



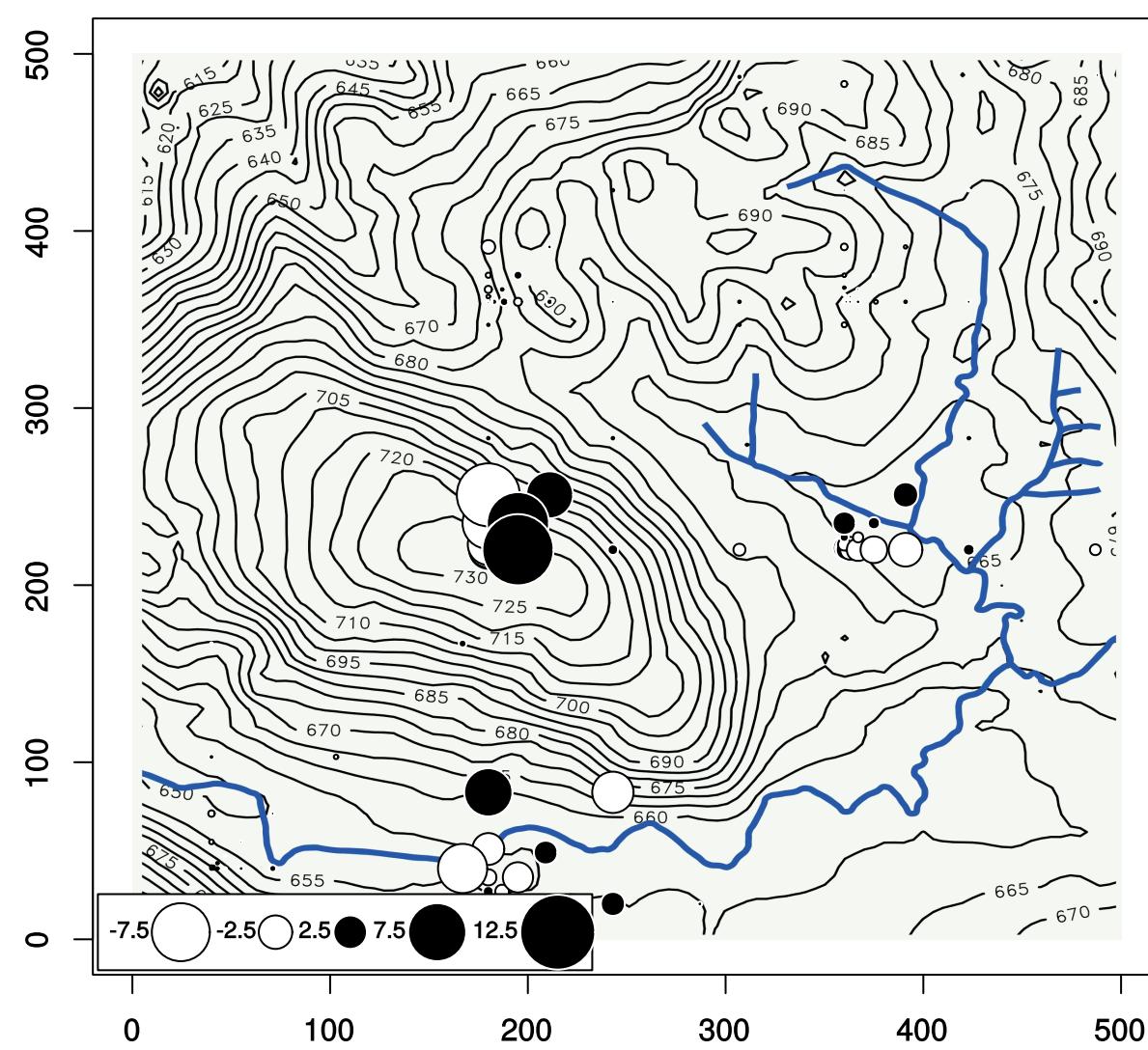
PCNM X9



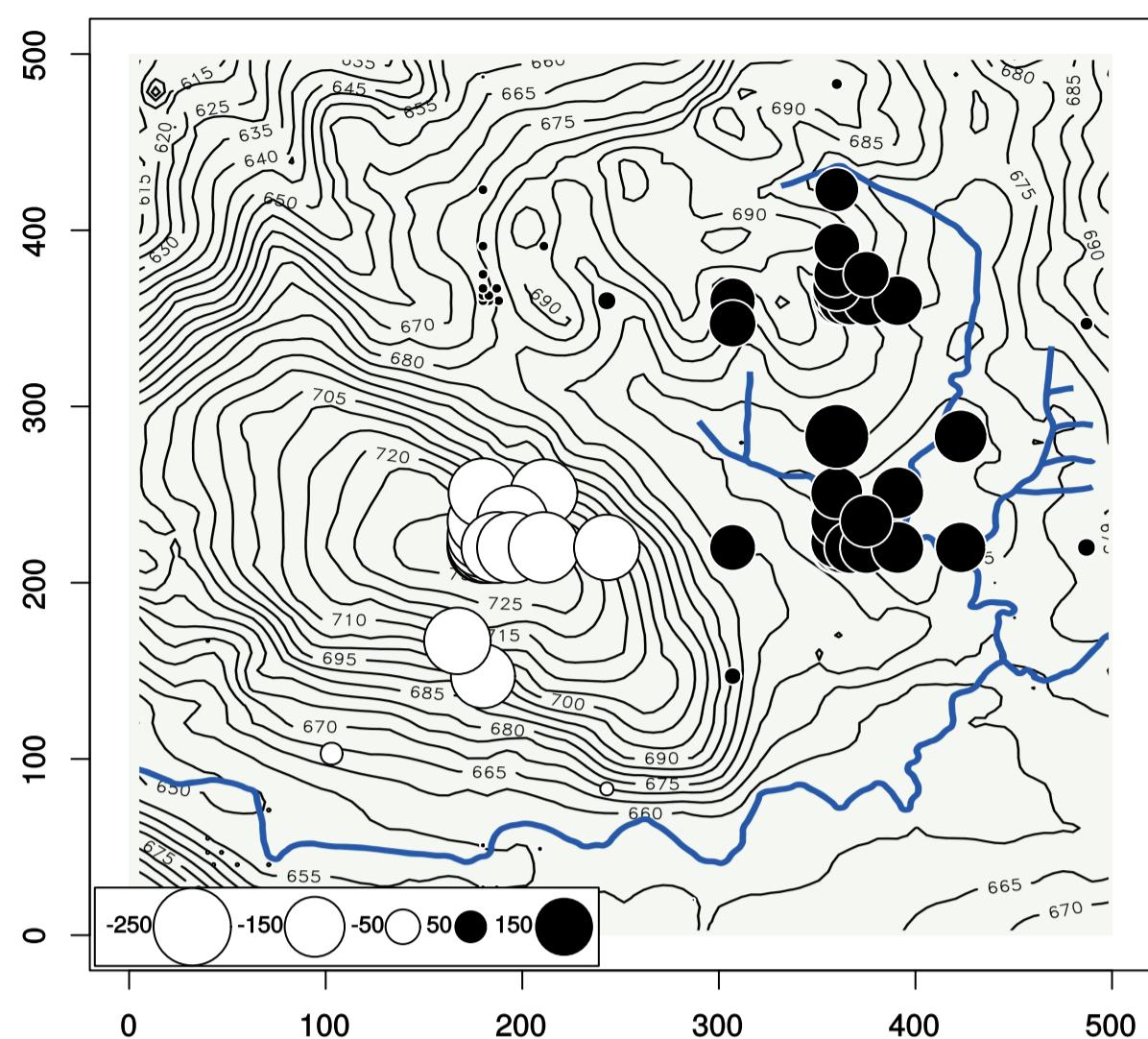
PCNM X29



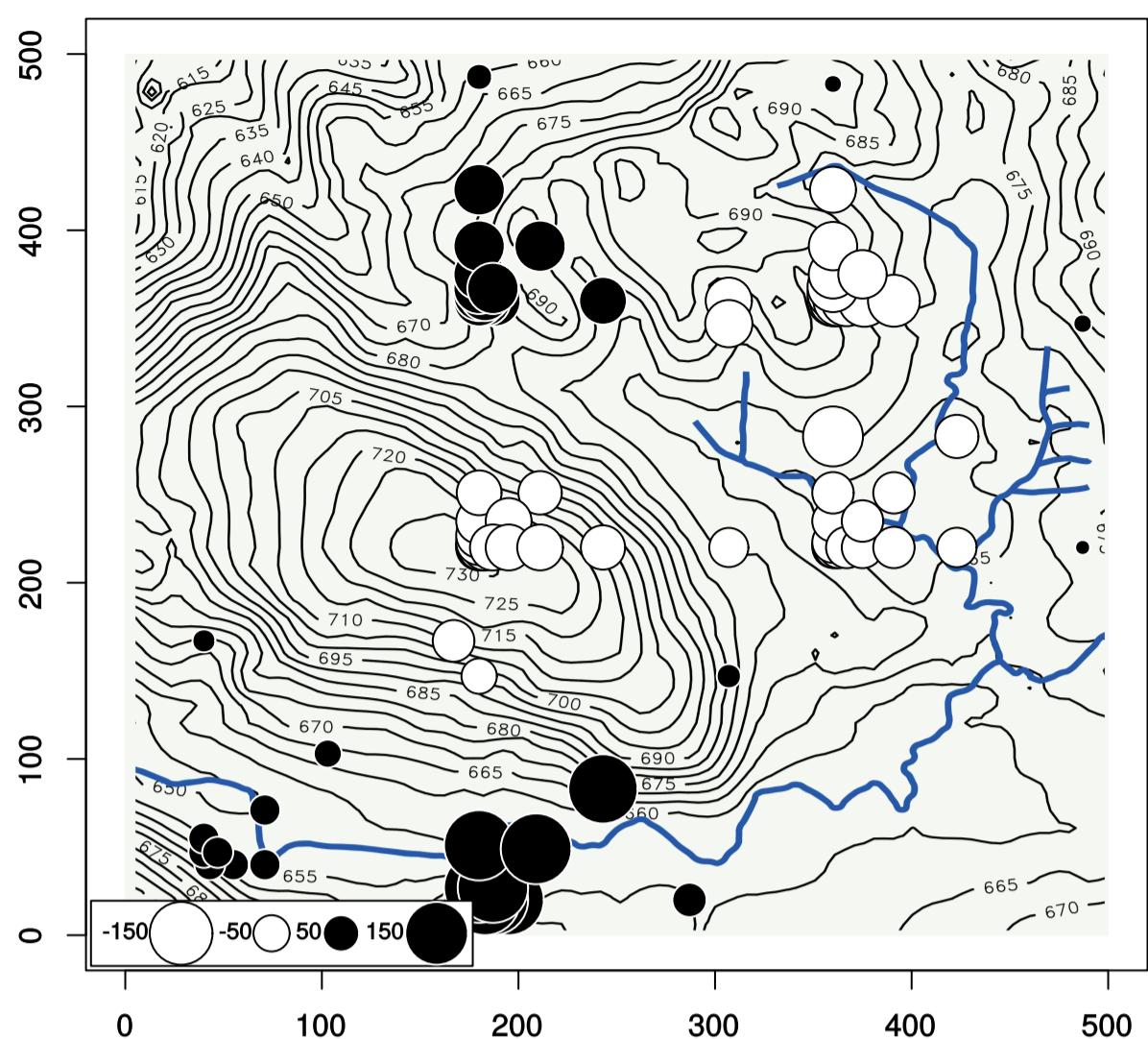
PCNM X34



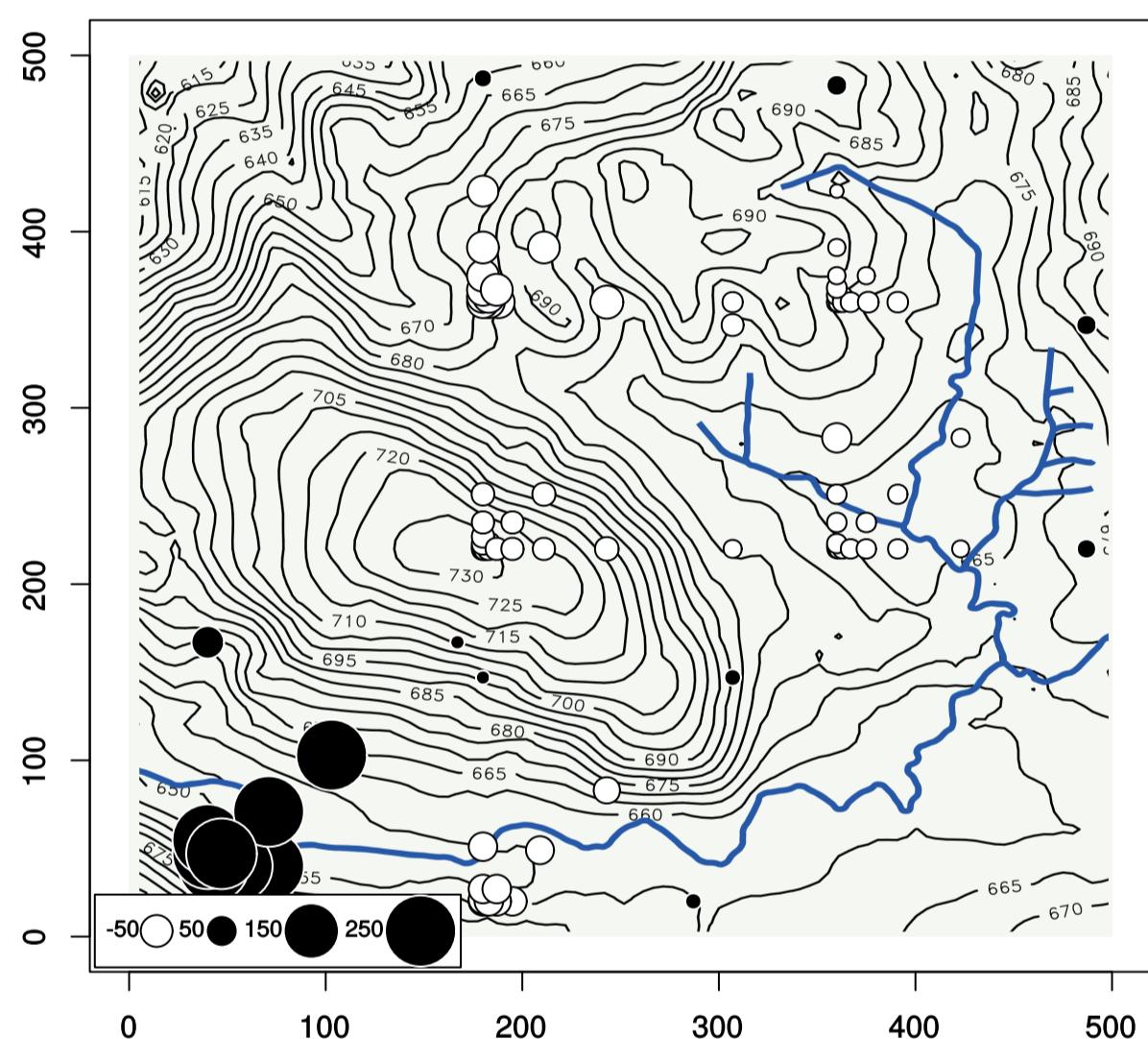
**PCNM X1**



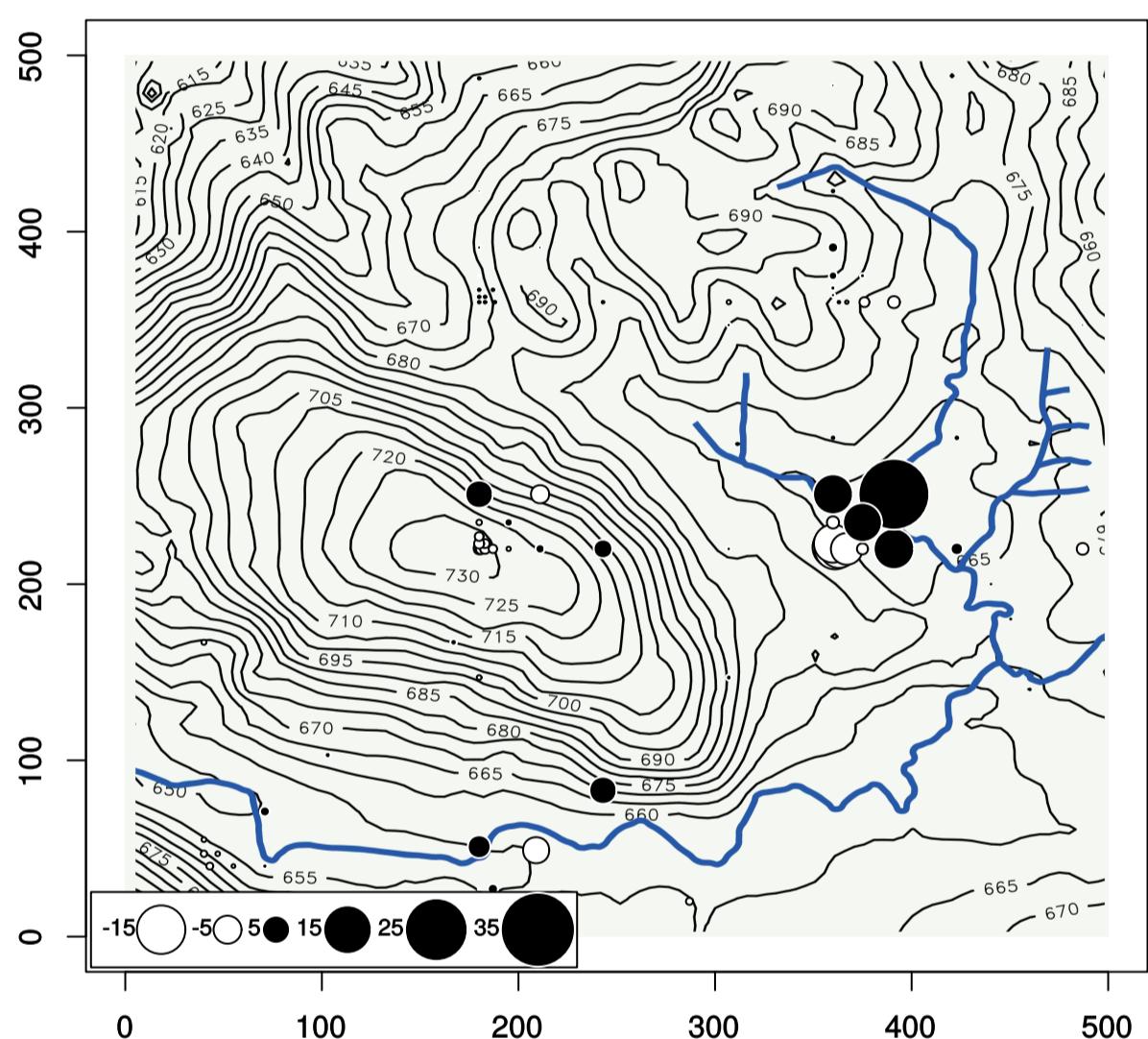
**PCNM X3**



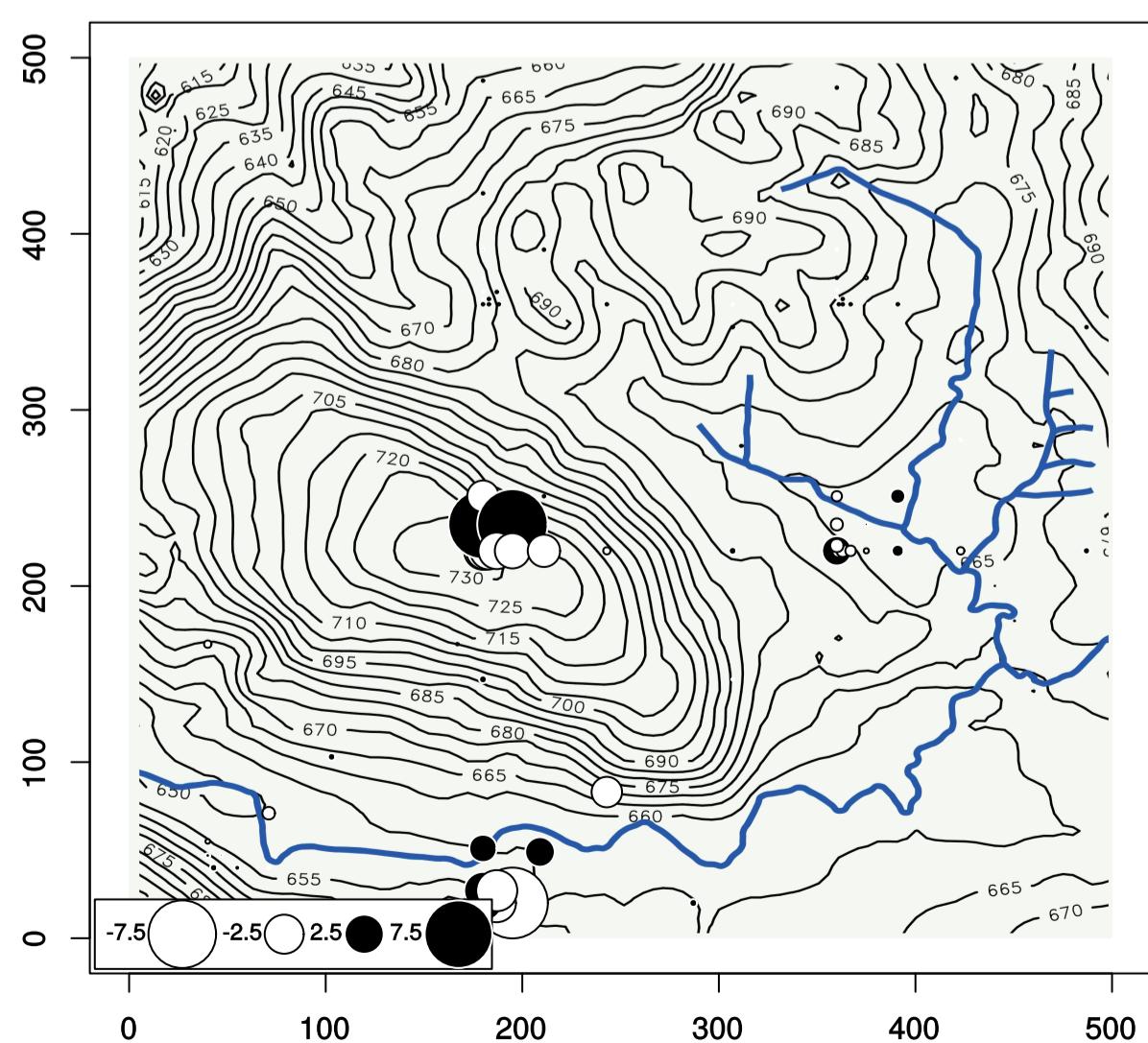
**PCNM X5**

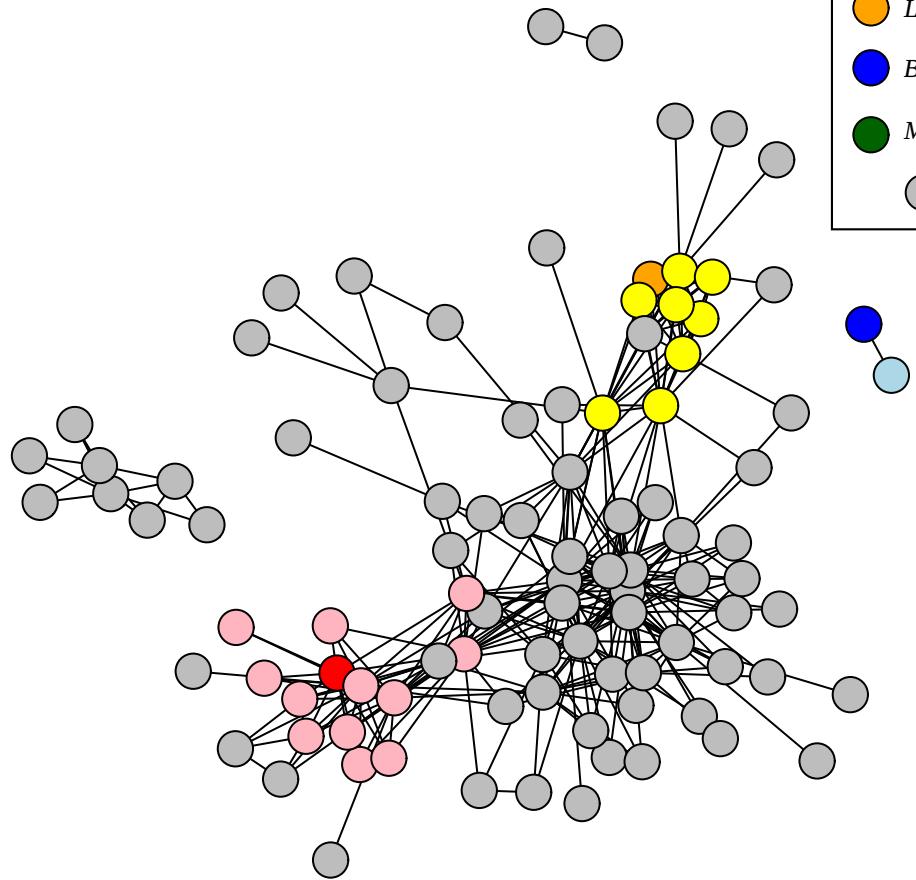
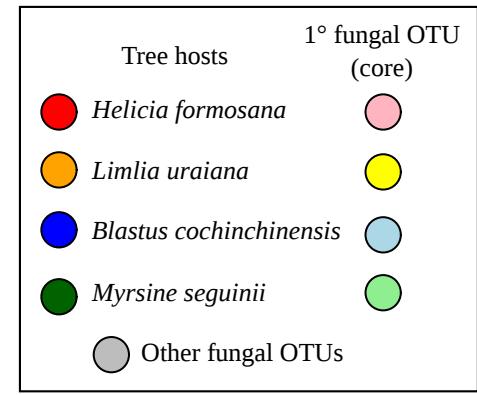


**PCNM X20**

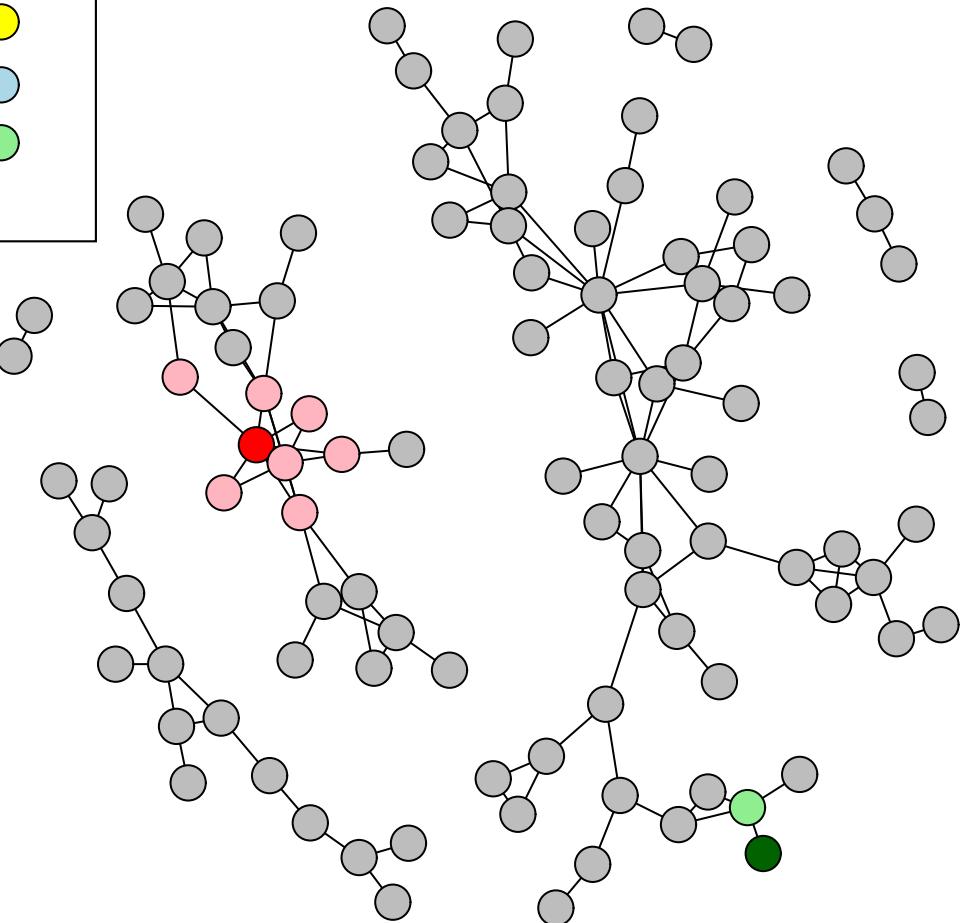


**PCNM X30**





Leaves



Wood

# CSrev

August 27, 2018

Okay, time to rebuild our old statistical pipeline as best as we can. We will keep only the essential portions, in the interests of time and clarity.

As of writing this, this analysis is being kept as a folder in the repo of the original analysis: [https://github.com/danchurch/taiwan\\_combined\\_stats/tree/master/CSrev](https://github.com/danchurch/taiwan_combined_stats/tree/master/CSrev). As long as it is there, it can be viewed with Jupyter nbviewer [here](#).

The original version of this [here](#). As long it remains in that repo, you can read it more easily with Jupyter nbviewer [here](#).

```
Section ??  
## Contents  
Section ??  
Section ?? - Section ??  
- Section ??  
- Section ?? - Section ?? - Section ??  
    Section ?? - Section ?? - Section ??  
    Section ?? - Section ??  
- Section ??  
    Section ?? - Section ?? - Section ??  
    Section ??  
    Section ?? - Section ?? - Section ?? - Section ??  
-Section ?? - Section ??  
-Section ?? - Section ??  
    Section ?? - Section ??  
- Section ?? - Section ?? - Section ?? - Section ?? - Section ?? - Section ??  
    Section ??  
- Section ??  
- Section ?? - Section ?? - Section ?? - Section ?? - Section ??  
    Section ??  
Importing biom table and setup
```

```
In [1]: library('phyloseq')  
        library('DESeq2')  
        library('vegan')  
        library('cooccur')  
        library('igraph')  
        library('ecodist')  
        library('ade4')  
        library('png')
```

```
library('randomcoloR')
library('VennDiagram')

Loading required package: S4Vectors
Loading required package: stats4
Loading required package: BiocGenerics
Loading required package: parallel

Attaching package: BiocGenerics

The following objects are masked from package:parallel:

clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
clusterExport, clusterMap, parApply, parCapply, parLapply,
parLapplyLB, parRapply, parSapply, parSapplyLB
```

The following objects are masked from package:stats:

```
IQR, mad, sd, var, xtabs
```

The following objects are masked from package:base:

```
anyDuplicated, append, as.data.frame, cbind, colMeans, colnames,
colSums, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
grepl, intersect, is.unsorted, lapply, lengths, Map, mapply, match,
mget, order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
rbind, Reduce, rowMeans, rownames, rowSums, sapply, setdiff, sort,
table, tapply, union, unique, unsplit, which, which.max, which.min
```

Attaching package: S4Vectors

The following object is masked from package:base:

```
expand.grid
```

Loading required package: IRanges

Attaching package: IRanges

The following object is masked from package:phyloseq:

```
distance
```

```
Loading required package: GenomicRanges
Loading required package: GenomeInfoDb
Loading required package: SummarizedExperiment
Loading required package: Biobase
```

```
Welcome to Bioconductor
```

```
Vignettes contain introductory material; view with  
'browseVignettes()'. To cite Bioconductor, see  
'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
Attaching package: Biobase
```

```
The following object is masked from package:phyloseq:
```

```
sampleNames
```

```
Loading required package: DelayedArray  
Loading required package: matrixStats
```

```
Attaching package: matrixStats
```

```
The following objects are masked from package:Biobase:
```

```
anyMissing, rowMedians
```

```
Attaching package: DelayedArray
```

```
The following objects are masked from package:matrixStats:
```

```
colMaxs, colMins, colRanges, rowMaxs, rowMins, rowRanges
```

```
The following object is masked from package:base:
```

```
apply
```

```
Loading required package: permute  
Loading required package: lattice  
This is vegan 2.5-2
```

```
Attaching package: igraph
```

```
The following object is masked from package:vegan:
```

```
diversity
```

```
The following object is masked from package:permute:
```

```
permute
```

```
The following object is masked from package:GenomicRanges:
```

```
union
```

The following object is masked from package:IRanges:

```
union
```

The following object is masked from package:S4Vectors:

```
union
```

The following objects are masked from package:BiocGenerics:

```
normalize, union
```

The following objects are masked from package:stats:

```
decompose, spectrum
```

The following object is masked from package:base:

```
union
```

Attaching package: ecodist

The following object is masked from package:vegan:

```
mantel
```

The following object is masked from package:SummarizedExperiment:

```
distance
```

The following object is masked from package:GenomicRanges:

```
distance
```

The following object is masked from package:IRanges:

```
distance
```

The following object is masked from package:phyloseq:

```
distance
```

Attaching package: ade4

```
The following object is masked from package:GenomicRanges:
```

```
score
```

```
The following object is masked from package:IRanges:
```

```
score
```

```
The following object is masked from package:BiocGenerics:
```

```
score
```

```
Loading required package: grid
```

```
Loading required package: futile.logger
```

Take a look at our raw biom table:

```
In [6]: biom95 <- import_biom('combo_otu_wMeta.biom', parseFunction=parse_taxonomy_greengenes)
      #save(biom95, file='biom95.rda')
```

```
In [6]: biom95
```

```
phyloseq-class experiment-level object
otu_table()    OTU Table:           [ 11588 taxa and 232 samples ]
sample_data()  Sample Data:        [ 232 samples by 11 sample variables ]
tax_table()    Taxonomy Table:     [ 11588 taxa by 7 taxonomic ranks ]
```

```
In [10]: sample_data(biom95)
```

	vegcom	stream_distance	Host_genus	Host_genus_species	Library	Fore
100leaf	2	25.97654	Helicia	Helicia_formosana	L	7
101leaf	3	18.36984	Helicia	Helicia_formosana	L	7
102leaf	3	21.3725	Cleyera	Cleyera_japonica	L	7
103leaf	3	11.08831	Helicia	Helicia_formosana	L	7
104leaf	3	1.409998	Helicia	Helicia_formosana	L	7
105leaf	3	22.46722	Limlia	Limlia_uriana	L	7
106leaf	2	82.49734	Helicia	Helicia_formosana	L	3
107leaf	1	64.85876	Blastus	Blastus_cochinchinensis	L	3
108leaf	1	19.02113	Cleyera	Cleyera_japonica	L	3
109leaf	3	13.46815	Meliosma	Meliosma_squamulata	L	7
110leaf	3	13.46815	Limlia	Limlia_uriana	L	7
111leaf	3	20.37973	Limlia	Limlia_uriana	L	7
112leafA	2	4.250151	Blastus	Blastus_cochinchinensis	L	7
112leafB	2	4.250151	Blastus	Blastus_cochinchinensis	L	7
113leafA	3	12.80385	Schefflera	Schefflera_octophylla	L	7
113leafB	3	12.80385	Schefflera	Schefflera_octophylla	L	7
114leaf	2	6.015914	Helicia	Helicia_formosana	L	7
115leaf	3	16.75811	Cyathea	Cyathea_podophylla	L	4
116leaf	3	24.2118	Helicia	Helicia_formosana	L	4
117leaf	3	30.79582	Limlia	Limlia_uriana	L	7
118leaf	3	30.50139	Limlia	Limlia_uriana	L	7
119leaf	3	29.54256	Limlia	Limlia_uriana	L	7
120leaf	3	29.84645	Limlia	Limlia_uriana	L	7
121leaf	3	27.95829	Blastus	Blastus_cochinchinensis	L	4
122leaf	3	27.07621	Limlia	Limlia_uriana	L	7
125leaf	3	27.32744	Litsea	Litsea_acuminata	L	4
126leafA	3	20.16175	Blastus	Blastus_cochinchinensis	L	7
126leafB	3	20.16175	Blastus	Blastus_cochinchinensis	L	7
127leaf	3	11.69457	Helicia	Helicia_formosana	L	7
128leaf	3	75.54028	Castanopsis	Castanopsis_cuspidata_var_carlesii	L	3
88w	2	47.36483	Cryptocarya	Cryptocarya_chinensis	W	7
89w	2	25.33942	Helicia	Helicia_formosana	W	7
92w	2	31.43415	Blastus	Blastus_cochinchinensis	W	7
93w	2	27.63549	Prunus	Prunus_phaeosticta	W	6
94w	2	25.06622	Ficus	Ficus_erecta_var_beecheyana	W	6
95w	2	25.33385	Ficus	Ficus_erecta_var_beecheyana	W	6
96w	3	24.37728	Helicia	Helicia_formosana	W	7
97w	3	24.09903	Helicia	Helicia_formosana	W	7
99w	3	23.19752	Helicia	Helicia_formosana	W	7
100w	2	25.97654	Helicia	Helicia_formosana	W	7
101w	3	18.36984	Helicia	Helicia_formosana	W	7
102w	3	21.3725	Cleyera	Cleyera_japonica	W	7
104w	3	1.409998	Helicia	Helicia_formosana	W	7
106w	2	82.49734	Helicia	Helicia_formosana	W	3
107w	1	64.85876	Blastus	Blastus_cochinchinensis	W	3
108w	1	19.02113	Cleyera	Cleyera_japonica	W	3
109w	3	13.46815	Meliosma	Meliosma_squamulata	W	7
114w	2	6.015914	Helicia	Helicia_formosana	W	7
115w	3	16.75811	Cyathea	Cyathea_podophylla	W	4
121w	3	27.95829	Blastus	Blastus_cochinchinensis	W	4
124w	3	28.9161391514	Glochidion	Glochidion_acuminatum	W	7

## Controls

Our negative controls will be used to generally cleanup our biom table, by removing from all samples in the study the OTUs that we find in the negative control, to the levels of abundance that we find in our positive control. But our negative controls can also be used to estimate levels of tag-switching. Contaminants that appear to have originated from our mock community positive control that have been observed in the negative are candidates for tag-switching.

### Negative controls

```
In [89]: neg95 <- subset_samples(biom95, sample_names(biom95)=='Neg')
```

```
In [90]: neg95
```

```
phyloseq-class experiment-level object
otu_table()    OTU Table:           [ 11588 taxa and 1 samples ]
sample_data()  Sample Data:        [ 1 samples by 11 sample variables ]
tax_table()    Taxonomy Table:     [ 11588 taxa by 7 taxonomic ranks ]
```

How many reads are in our negative control?

```
In [16]: sum(taxa_sums(neg95)[taxa_sums(neg95)>0])
```

1501

How many OTUs?

```
In [17]: length(taxa_sums(neg95)[taxa_sums(neg95)>0])
```

43

Which OTUs, and what's their distribution?

```
In [19]: taxa_sums(neg95)[taxa_sums(neg95)>0]
```

```
OTU167:Dc-X 469 OTU187:Dc-PosG 85 OTU256:Dc-X 35 OTU762:Dc-X 53 OTU891:1w 58
OTU306:Dc-PosG 96 OTU119:Dc-PosG 1 OTU220:Dc-PosG 1 OTU164:Dc-PosG 51
OTU235:Dc-PosG 1 OTU386:Dc-PosG 88 OTU264:Dc-PosG 1 OTU560:Dc-PosG 39
OTU1183:36w 1 OTU64:1w 1 OTU409:4w 1 OTU417:1w 15 OTU437:1w 1 OTU18:9w 1
OTU414:13w 1 OTU655:1w 1 OTU2029:2w 1 OTU1432:2w 28 OTU1332:11w 44 OTU315:4w 162
OTU250:4w 1 OTU1599:9w 4 OTU2831:5w 4 OTU84:38w 12 OTU7329:38w 1 OTU2003:Neg 42
OTU925:133w 1 OTU1214:9w 53 OTU1747:11w 53 OTU1549:104w 28 OTU6852:Neg 20
OTU1496:23w 1 OTU588:32w 1 OTU1888:25w 1 OTU1444:49w 38 OTU46:60w 3 OTU972:130w 1
OTU2115:131w 2
```

To check for tag-switching, we'll look at our positive controls, or "mock-community" samples, for OTUs shared between the two controls. Such OTUs are good candidates for tag-switching.

```
In [20]: reads <- taxa_sums(neg95)[taxa_sums(neg95) > 0]
          reads <- sort(reads, decreasing = TRUE)
          sink('neg95names.txt') ## read out these OTUs so we can use them in python env
          names(reads)
          sink()
```

```

1. 'OTU167:Dc-X' 2. 'OTU315:4w' 3. 'OTU306:Dc-PosG' 4. 'OTU386:Dc-PosG'
5. 'OTU187:Dc-PosG' 6. 'OTU891:1w' 7. 'OTU762:Dc-X' 8. 'OTU1214:9w' 9. 'OTU1747:11w'
10. 'OTU164:Dc-PosG' 11. 'OTU1332:11w' 12. 'OTU2003:Neg' 13. 'OTU560:Dc-PosG'
14. 'OTU1444:49w' 15. 'OTU256:Dc-X' 16. 'OTU1432:2w' 17. 'OTU1549:104w' 18. 'OTU6852:Neg'
19. 'OTU417:1w' 20. 'OTU84:38w' 21. 'OTU1599:9w' 22. 'OTU2831:5w' 23. 'OTU46:60w'
24. 'OTU2115:131w' 25. 'OTU119:Dc-PosG' 26. 'OTU220:Dc-PosG' 27. 'OTU235:Dc-PosG'
28. 'OTU264:Dc-PosG' 29. 'OTU1183:36w' 30. 'OTU64:1w' 31. 'OTU409:4w' 32. 'OTU437:1w'
33. 'OTU18:9w' 34. 'OTU414:13w' 35. 'OTU655:1w' 36. 'OTU2029:2w' 37. 'OTU250:4w'
38. 'OTU7329:38w' 39. 'OTU925:133w' 40. 'OTU1496:23w' 41. 'OTU588:32w' 42. 'OTU1888:25w'
43. 'OTU972:130w'
```

Clean up with sed. Back into BASH kernel:

```
In [1]: ## BASH
sed 's/^\[.*\]//g' neg95names.txt | sed 's/^\\s*//g' | sed 's/\\s\\+/,/g' | sed '$s/,/$\\)/'|
```

Checked with vim, had to add a comma or two, not sure why. Anyway, its a tuple. And the second half of the script:

```
In [11]: cat MCseq.py
```

```

with open('otus_95_combo_nolb.fasta', 'r') as zoop:
    refseq = zoop.readlines()

with open('mcseq.txt', 'w') as goop:
    for j, otu in enumerate(mcseq):
        for i, line in enumerate(refseq):
            if otu in line:
                goop.write(line)
                goop.write(refseq[i+1])
```

```
In [12]: cat neg95list.txt MCseq.py > makeMCseq.py
```

```
In [13]: cat makeMCseq.py
```

```

mcseq=("OTU167:Dc-X", "OTU315:4w", "OTU306:Dc-PosG", "OTU386:Dc-PosG",
"OTU187:Dc-PosG", "OTU891:1w", "OTU762:Dc-X", "OTU1214:9w",
"OTU1747:11w", "OTU164:Dc-PosG", "OTU1332:11w", "OTU2003:Neg",
"OTU560:Dc-PosG", "OTU1444:49w", "OTU256:Dc-X", "OTU1432:2w",
"OTU1549:104w", "OTU6852:Neg", "OTU417:1w", "OTU84:38w",
"OTU1599:9w", "OTU2831:5w", "OTU46:60w", "OTU2115:131w",
"OTU119:Dc-PosG", "OTU220:Dc-PosG", "OTU235:Dc-PosG", "OTU264:Dc-PosG",
"OTU1183:36w", "OTU64:1w", "OTU409:4w", "OTU437:1w",
"OTU18:9w", "OTU414:13w", "OTU655:1w", "OTU2029:2w",
"OTU250:4w", "OTU7329:38w", "OTU925:133w", "OTU1496:23w",
"OTU588:32w", "OTU1888:25w", "OTU972:130w")
```

```

with open('otus_95_combo_nolb.fasta', 'r') as zoop:
    refseq = zoop.readlines()

with open('mcseq.txt', 'w') as goop:
    for j, otu in enumerate(mcseq):
        for i, line in enumerate(refseq):
            if otu in line:
                goop.write(line)
                goop.write(refseq[i+1])

```

In [14]: *## run this to get sequences of OTUs that are in our positive control:*  
`python3 makeMCseq.py`

Make a friendly link to our sanger sequences of full ITS region of our positive control cultures, and make a blastable database from them:

In [17]: `aa=$(find ~ -type f -name "BioI-6098_OConnor_34875.seq.txt")  
ln -s $aa ./mcsanger.fasta`

In [19]: *## make our searchable database of sanger positive control sequences for blast:*  
`makeblastdb -in mcsanger.fasta -dbtype nucl -logfile dberrors.txt`

Do our searches, with a couple of output formats:

In [20]: *## clean up read ids a little:*  
`sed '/>/ s/;size=.*//' mcseq.txt | sed '/>/ s/;size=.*//' mcseq.txt > mockseqs_Neg95.fasta  
blastn -query mockseqs_Neg95.fasta -db mcsanger.fasta -out mcblast_Neg95.txt -num_descriptions 10  
blastn -query mockseqs_Neg95.fasta -db mcsanger.fasta -out mcblast_Neg95.csv -outfmt 10  
sed '1 i\\qseqid,sseqid,pident,length,mismatch,gapopen,qstart,qend,sstart,send,evalue,bi' mcblast_Neg95.csv -i  
sed 's/_ITS[1,4],/,/g' mcblast_Neg95.csv -i  
sed 's/Sample//g' mcblast_Neg95.csv -i`

This csv file can be imported as a dataframe into R:

In [85]: *## change kernel to R*  
`library('phyloseq')  
blast <- read.csv("mcblast_Neg95.csv", stringsAsFactors=FALSE)  
blast`

qseqid	sseqid	pident	length	mismatch	gapopen	qstart	qend	sstart	send	evalu
OTU306:Dc-PosG	9	100.00	176	0	0	1	176	504	329	3e-92
OTU386:Dc-PosG	19	100.00	175	0	0	1	175	513	339	1e-91
OTU164:Dc-PosG	8	100.00	143	0	0	1	143	451	309	5e-74
OTU560:Dc-PosG	4	100.00	180	0	0	1	180	485	306	2e-94
OTU256:Dc-X	6	100.00	184	0	0	1	184	535	352	1e-96
OTU84:38w	9	100.00	31	0	0	27	57	530	500	1e-11
OTU1599:9w	1	90.22	92	8	1	37	128	459	369	6e-30
OTU119:Dc-PosG	1	99.31	144	1	0	1	144	434	291	7e-73
OTU220:Dc-PosG	22	100.00	168	0	0	1	168	512	345	8e-88
OTU235:Dc-PosG	20	100.00	217	0	0	1	217	605	389	6e-115
OTU264:Dc-PosG	16	100.00	180	0	0	1	180	502	323	2e-94

Get rid of the lower quality matches:

```
In [87]: goodblast <- blast[blast$pident > 94 & blast$length > 90,]
```

Make some useful vectors and graph:

```
In [91]: reads <- taxa_sums(neg95)[taxa_sums(neg95) > 0]
          reads <- sort(reads, decreasing = TRUE)
          Neg95.gen <- tax_table(neg95)[names(reads),6] ## genus, from initial tax assignments
          Neg95.gen[is.na(Neg95.gen)] <- "NoID"
          Neg95.species <- tax_table(neg95)[names(reads),7] ## species, from initial tax assignments
          Neg95.species[is.na(Neg95.species)] <- "NoID"
          member <- names(reads) %in% goodblast$qseqid ## membership in mock community (probably
          MC <- vector(length = length(reads)); MC[] <- 0 ## empty vector, for MC sample #, filled
```

```
In [92]: Neg95bar <- data.frame(reads, member, MC, Neg95.gen, Neg95.species, stringsAsFactors=FALSE)
```

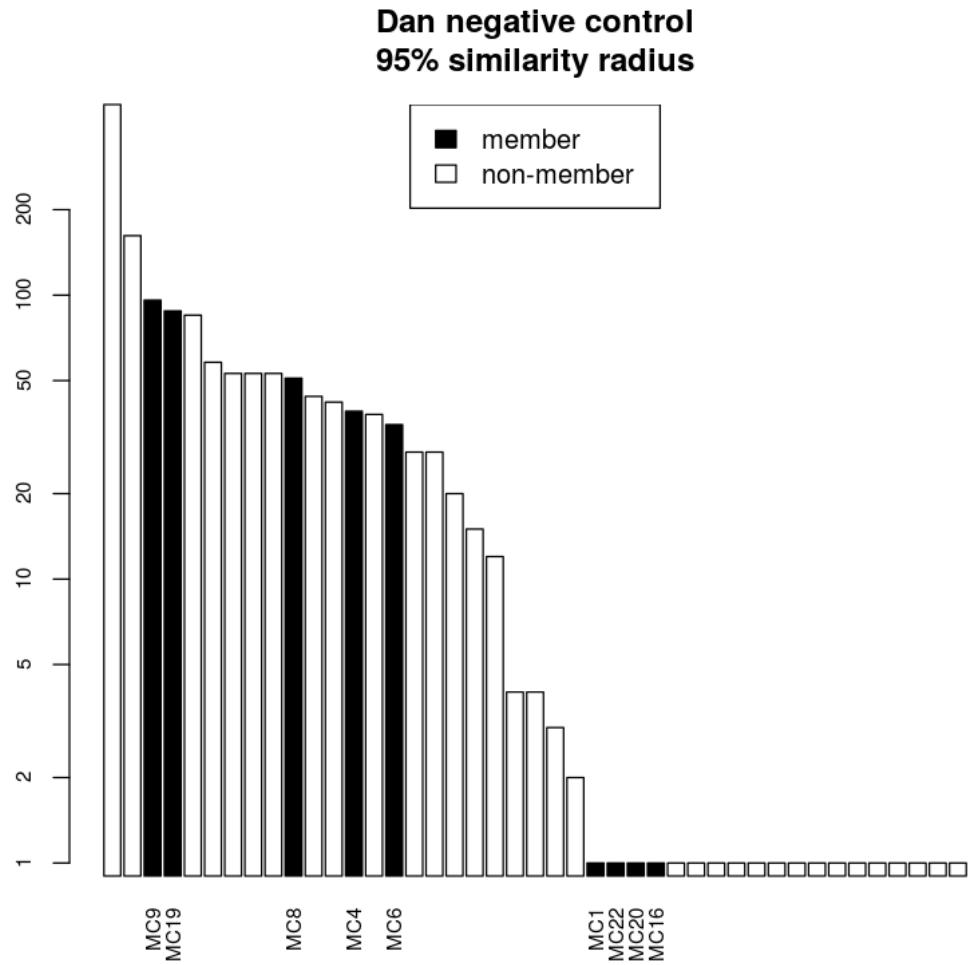
```
##fill the mock community sample number by querying our csv from blast results

colnames(Neg95bar)[1] <- "reads"
for (i in 1:nrow(Neg95bar)){
  if (rownames(Neg95bar)[i] %in% goodblast$qseqid) {
    search <- grep(rownames(Neg95bar)[i], goodblast$qseqid)
    Neg95bar$MC[i] <- goodblast$sseqid[search]
  }
}
```

```
In [95]: #save(Neg95bar, file='Neg95bar.rda')
```

```
In [7]: inout <- paste('MC', Neg95bar$MC, sep=' ')
inout[inout=='MCO'] <- NA
```

```
par(cex.axis = .75, mar=c(7,4,4,2))
barplot(Neg95bar$reads, las=3, log='y', names.arg=inout, col = member,
        main = 'Dan negative control\n95% similarity radius')
legend("top", legend=c('member','non-member'), fill=c(1,0))
```



Black indicates a member of our mock community, observed in the negative control.

```
In [13]: inout <- paste('MC', Neg95bar$MC, sep=' ')
inout[inout=='MCO'] <- NA

svg(file='Dan_negative_21.06.2018.svg')
par(cex.axis = .75, mar=c(7,4,4,2))
barplot(Neg95bar$reads, las=3, log='y', names.arg=inout, col = member,
        main = 'Dan negative control\n95% similarity radius')
legend("top", legend=c('member','non-member'), fill=c(1,0))
dev.off()
```

png: 2

This new pipeline shows a few less low-abundance OTUs shared between the negative and positive controls, but the big picture is more or less the same. A fair amount of tag-switched reads, with the two most abundant tag-switched OTUs at around 100 reads.

#### Positive controls

We had two types of positive controls

1. Genomic = Genomic DNA of each mock community fungi, in equal ng/uL, were combined into one sample.
2. ITS-only = PCR product from ITS1f and ITS4 from genomic DNA for each member of the mock community was diluted to equal concentrations and combined into one sample.

Let's take a look at our genomic positive controls.

```
In [10]: aa95 <- subset_samples(biom95, sample_names(biom95) == 'PosG')
dangen95 <- taxa_sums(aa95)[taxa_sums(aa95) > 0]
dangen95 <- sort(dangen95, decreasing = TRUE)
length(dangen95)
```

58

```
In [19]: save(dangen95, file='dangen95.rda')
```

24 species were intentionally put into the positive control. Here we detected 58.

```
In [18]: options(repr.plot.width = 10, repr.plot.height = 4)
barplot(dangen95, names.arg='')
```



```
In [20]: aa95 <- subset_samples(biom95, sample_names(biom95) == 'PosI')
danITS95 <- taxa_sums(aa95)[taxa_sums(aa95) > 0]
danITS95 <- sort(danITS95, decreasing = TRUE)
options(repr.plot.width = 10, repr.plot.height = 4)
barplot(danITS95, names.arg='')
```



### Genomic positive control

To show which of these are intended members of the mock community, not contaminants, we'll use the same kind of pipeline as we used above for the negative controls. Focus on the genomic positive control first, in which genomic DNA from all of our MC cultures were introduced in equal DNA concentrations into a single sample.

```
In [3]: aa95 <- subset_samples(biom95, sample_names(biom95) == 'PosG')
danGenreads <- taxa_sums(aa95)[taxa_sums(aa95) > 0]
danGenreads <- sort(danGenreads, decreasing = TRUE)
sink('danGenreads.txt') ## read out these OTUs so we can use them in python env
names(danGenreads)
sink()
```

1. 'OTU106:Dc-PosG'
2. 'OTU119:Dc-PosG'
3. 'OTU164:Dc-PosG'
4. 'OTU191:Dc-PosG'
5. 'OTU220:Dc-PosG'
6. 'OTU235:Dc-PosG'
7. 'OTU258:Dc-PosG'
8. 'OTU271:Dc-PosG'
9. 'OTU270:Dc-PosG'
10. 'OTU358:Dc-PosG'
11. 'OTU306:Dc-PosG'
12. 'OTU264:Dc-PosG'
13. 'OTU386:Dc-PosG'
14. 'OTU526:Dc-PosG'
15. 'OTU407:Dc-PosG'
16. 'OTU256:Dc-X'
17. 'OTU608:Dc-PosG'
18. 'OTU826:Dc-PosG'
19. 'OTU3183:Dc-PosG'
20. 'OTU3723:Dc-PosG'
21. 'OTU3674:Dc-PosG'
22. 'OTU12510:PosG'
23. 'OTU4453:Dc-PosG'
24. 'OTU4210:Dc-PosG'
25. 'OTU733:Dc-PosG'
26. 'OTU9035:Dc-PosG'
27. 'OTU1153:Dc-PosI'
28. 'OTU560:Dc-PosG'
29. 'OTU8984:Dc-PosG'
30. 'OTU10988:PosG'
31. 'OTU9833:Dc-PosI'
32. 'OTU248:20w'
33. 'OTU18:9w'
34. 'OTU610:1w'
35. 'OTU257:3w'
36. 'OTU12502:PosG'
37. 'OTU8986:Dc-PosG'
38. 'OTU8401:Dc-PosG'
39. 'OTU152:1w'
40. 'OTU77:1w'
41. 'OTU28:2w'
42. 'OTU249:1w'
43. 'OTU5738:17w'
44. 'OTU1075:100w'
45. 'OTU391:2w'
46. 'OTU1524:3w'
47. 'OTU7955:106w'
48. 'OTU5111:14w'
49. 'OTU84:38w'
50. 'OTU1124:36w'
51. 'OTU1466:9w'
52. 'OTU1214:9w'
53. 'OTU1941:17w'
54. 'OTU428:45w'
55. 'OTU2567:130w'
56. 'OTU1516:133w'
57. 'OTU1866:133w'
58. 'OTU12503:PosG'

The script to get the sequences of OTUs present in our illumina survey of our mock-community.

```
In [1]: cat danGen_makeMCseq.py
```

```
mcseq=("OTU256:Dc-X", "OTU306:Dc-PosG", "OTU119:Dc-PosG", "OTU220:Dc-PosG",
"OTU270:Dc-PosG", "OTU191:Dc-PosG", "OTU164:Dc-PosG", "OTU235:Dc-PosG",
```

```

"OTU358:Dc-PosG", "OTU106:Dc-PosG", "OTU3674:Dc-PosG", "OTU386:Dc-PosG",
"OTU271:Dc-PosG", "OTU407:Dc-PosG", "OTU258:Dc-PosG", "OTU264:Dc-PosG",
"OTU4210:Dc-PosG", "OTU1153:Dc-PosI", "OTU8984:Dc-PosG", "OTU526:Dc-PosG",
"OTU12510:PosG", "OTU3723:Dc-PosG", "OTU4453:Dc-PosG", "OTU608:Dc-PosG",
"OTU3183:Dc-PosG", "OTU826:Dc-PosG", "OTU733:Dc-PosG", "OTU8986:Dc-PosG",
"OTU9035:Dc-PosG", "OTU8401:Dc-PosG", "OTU560:Dc-PosG", "OTU10988:PosG",
"OTU9833:Dc-PosI", "OTU248:20w", "OTU152:1w", "OTU77:1w",
"OTU28:2w", "OTU249:1w", "OTU18:9w", "OTU5738:17w",
"OTU610:1w", "OTU1075:100w", "OTU391:2w", "OTU1524:3w",
"OTU257:3w", "OTU7955:106w", "OTU5111:14w", "OTU84:38w",
"OTU1124:36w", "OTU1466:9w", "OTU1214:9w", "OTU1941:17w",
"OTU428:45w", "OTU2567:130w", "OTU1516:133w", "OTU1866:133w",
"OTU12502:PosG", "OTU12503:PosG")

```

```

with open('otus_95_combo_nolb.fasta', 'r') as zoop:
    refseq = zoop.readlines()

with open('seqs_dangen95.fasta', 'w') as goop:
    for j, otu in enumerate(mcseq):
        for i, line in enumerate(refseq):
            if otu in line:
                goop.write(line)
                goop.write(refseq[i+1])

```

In [4]: python3 danGen\_makeMCseq.py

In [5]: sed -i '/^>/ s/;size=.\*//' seqs\_dangen95.fasta

```

blastn -query seqs_dangen95.fasta -db mcsanger.fasta -out mcblast_Dangen95.txt -num_des
blastn -query seqs_dangen95.fasta -db mcsanger.fasta -out mcblast_Dangen95.csv -outfmt 1

sed '1 i\qseqid,sseqid,pident,length,mismatch,gapopen,qstart,qend,sstart,send,evalue,bit'
sed 's/_ITS[1,4]_,/_g' mcblast_Dangen95.csv -i
sed 's/Sample//g' mcblast_Dangen95.csv -i

```

In [4]: ## change kernel to R  
library('phyloseq')

In [2]: blast <- read.csv("mcblast\_Dangen95.csv", stringsAsFactors=FALSE)  
blast

qseqid	sseqid	pident	length	mismatch	gapopen	qstart	qend	sstart	send	evalu
OTU256:Dc-X	6	100.00	184	0	0	1	184	535	352	1e-96
OTU306:Dc-PosG	9	100.00	176	0	0	1	176	504	329	3e-92
OTU119:Dc-PosG	1	99.31	144	1	0	1	144	434	291	7e-73
OTU220:Dc-PosG	22	100.00	168	0	0	1	168	512	345	8e-88
OTU270:Dc-PosG	12	100.00	178	0	0	1	178	501	324	2e-93
OTU191:Dc-PosG	14	100.00	197	0	0	1	197	566	370	7e-10
OTU164:Dc-PosG	8	100.00	143	0	0	1	143	451	309	5e-74
OTU235:Dc-PosG	20	100.00	217	0	0	1	217	605	389	6e-11
OTU358:Dc-PosG	15	100.00	241	0	0	1	241	553	313	3e-12
OTU106:Dc-PosG	10	100.00	174	0	0	1	174	482	309	4e-91
OTU3674:Dc-PosG	1	95.24	126	6	0	19	144	416	291	1e-54
OTU386:Dc-PosG	19	100.00	175	0	0	1	175	513	339	1e-91
OTU271:Dc-PosG	24	100.00	209	0	0	1	209	537	329	2e-11
OTU407:Dc-PosG	7	100.00	193	0	0	1	193	524	332	1e-10
OTU258:Dc-PosG	13	100.00	180	0	0	1	180	492	313	2e-94
OTU264:Dc-PosG	16	100.00	180	0	0	1	180	502	323	2e-94
OTU4210:Dc-PosG	8	96.00	125	5	0	19	143	433	309	1e-55
OTU1153:Dc-PosI	18	100.00	361	0	0	1	361	761	401	0e+00
OTU526:Dc-PosG	5	100.00	232	0	0	1	232	562	331	3e-12
OTU12510:PosG	8	95.62	137	4	2	1	135	451	315	4e-60
OTU3723:Dc-PosG	1	95.00	140	5	2	1	140	434	297	4e-60
OTU4453:Dc-PosG	2	100.00	143	0	0	1	143	463	321	5e-74
OTU608:Dc-PosG	11	100.00	174	0	0	1	174	487	314	4e-91
OTU3183:Dc-PosG	8	94.89	137	5	2	1	135	451	315	2e-58
OTU826:Dc-PosG	3	100.00	157	0	0	1	157	555	399	9e-82
OTU733:Dc-PosG	23	100.00	208	0	0	1	208	528	321	6e-11
OTU8986:Dc-PosG	13	99.39	163	1	0	15	177	475	313	2e-83
OTU8401:Dc-PosG	1	99.16	119	1	0	19	137	409	291	5e-59
OTU560:Dc-PosG	4	100.00	180	0	0	1	180	485	306	2e-94
OTU10988:PosG	8	100.00	118	0	0	19	136	426	309	4e-60
OTU84:38w	9	100.00	31	0	0	27	57	530	500	1e-11
OTU12503:PosG	16	95.43	175	2	4	8	181	492	323	1e-76

```
In [5]: blast <- read.csv("mcblast_Dangen95.csv", stringsAsFactors=FALSE)
blast
```

qseqid	sseqid	pident	length	mismatch	gapopen	qstart	qend	sstart	send	evalu
OTU256:Dc-X	6	100.00	184	0	0	1	184	535	352	1e-96
OTU306:Dc-PosG	9	100.00	176	0	0	1	176	504	329	3e-92
OTU119:Dc-PosG	1	99.31	144	1	0	1	144	434	291	7e-73
OTU220:Dc-PosG	22	100.00	168	0	0	1	168	512	345	8e-88
OTU270:Dc-PosG	12	100.00	178	0	0	1	178	501	324	2e-93
OTU191:Dc-PosG	14	100.00	197	0	0	1	197	566	370	7e-10
OTU164:Dc-PosG	8	100.00	143	0	0	1	143	451	309	5e-74
OTU235:Dc-PosG	20	100.00	217	0	0	1	217	605	389	6e-11
OTU358:Dc-PosG	15	100.00	241	0	0	1	241	553	313	3e-12
OTU106:Dc-PosG	10	100.00	174	0	0	1	174	482	309	4e-91
OTU3674:Dc-PosG	1	95.24	126	6	0	19	144	416	291	1e-54
OTU386:Dc-PosG	19	100.00	175	0	0	1	175	513	339	1e-91
OTU271:Dc-PosG	24	100.00	209	0	0	1	209	537	329	2e-11
OTU407:Dc-PosG	7	100.00	193	0	0	1	193	524	332	1e-10
OTU258:Dc-PosG	13	100.00	180	0	0	1	180	492	313	2e-94
OTU264:Dc-PosG	16	100.00	180	0	0	1	180	502	323	2e-94
OTU4210:Dc-PosG	8	96.00	125	5	0	19	143	433	309	1e-55
OTU1153:Dc-PosI	18	100.00	361	0	0	1	361	761	401	0e+0
OTU526:Dc-PosG	5	100.00	232	0	0	1	232	562	331	3e-12
OTU12510:PosG	8	95.62	137	4	2	1	135	451	315	4e-60
OTU3723:Dc-PosG	1	95.00	140	5	2	1	140	434	297	4e-60
OTU4453:Dc-PosG	2	100.00	143	0	0	1	143	463	321	5e-74
OTU608:Dc-PosG	11	100.00	174	0	0	1	174	487	314	4e-91
OTU3183:Dc-PosG	8	94.89	137	5	2	1	135	451	315	2e-58
OTU826:Dc-PosG	3	100.00	157	0	0	1	157	555	399	9e-82
OTU733:Dc-PosG	23	100.00	208	0	0	1	208	528	321	6e-11
OTU8986:Dc-PosG	13	99.39	163	1	0	15	177	475	313	2e-83
OTU8401:Dc-PosG	1	99.16	119	1	0	19	137	409	291	5e-59
OTU560:Dc-PosG	4	100.00	180	0	0	1	180	485	306	2e-94
OTU10988:PosG	8	100.00	118	0	0	19	136	426	309	4e-60
OTU84:38w	9	100.00	31	0	0	27	57	530	500	1e-11
OTU12503:PosG	16	95.43	175	2	4	8	181	492	323	1e-76

Let's get rid of the short match, this is probably just a bit of the ssu or 5.8s.

In [6]: `goodblast <- blast[blast$length > 100, ]`

The sseqid column values are equivalent to numbers that we gave each DNA sample from our individual pure cultures. They go from MC1 to MC24, with sample MC21 skipped.

In [7]: `1:24 %in% goodblast$sseqid`

1. TRUE 2. TRUE 3. TRUE 4. TRUE 5. TRUE 6. TRUE 7. TRUE 8. TRUE 9. TRUE 10. TRUE  
 11. TRUE 12. TRUE 13. TRUE 14. TRUE 15. TRUE 16. TRUE 17. FALSE 18. TRUE 19. TRUE  
 20. TRUE 21. FALSE 22. TRUE 23. TRUE 24. TRUE

In [8]: `sort(unique(goodblast$sseqid)); length(unique(goodblast$sseqid))`

1. 1 2. 2 3. 3 4. 4 5. 5 6. 6 7. 7 8. 8 9. 9 10. 10 11. 11 12. 12 13. 13 14. 14 15. 15 16. 16 17. 18 18. 19  
 19. 20 20. 22 21. 23 22. 24

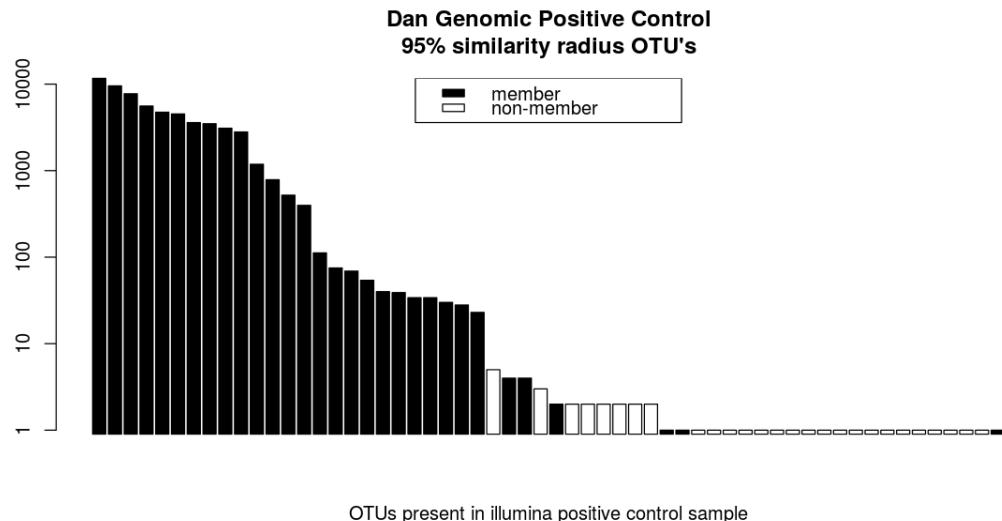
As per last time, we are missing MC17, which corresponds to *Schizosaccharomyces pombe*.  
Put together some info for a plot/dataframe

```
In [11]: member <- names(dangen95) %in% goodblast$qseqid
MC <- vector(length = length(dangen95))
dangen95bar <- data.frame(cbind(dangen95, member, MC))
for (i in 1:nrow(dangen95bar)){
  if (rownames(dangen95bar)[i] %in% goodblast$qseqid) {
    search <- grep(rownames(dangen95bar)[i], goodblast$qseqid)
    dangen95bar$MC[i] <- goodblast$sseqid[search]
  }
}
```

In [12]: head(dangen95bar)

	dangen95	member	MC
OTU106:Dc-PosG	11662	1	10
OTU119:Dc-PosG	9537	1	1
OTU164:Dc-PosG	7742	1	8
OTU191:Dc-PosG	5581	1	14
OTU220:Dc-PosG	4737	1	22
OTU235:Dc-PosG	4521	1	20

```
In [13]: options(repr.plot.width = 10, repr.plot.height = 5)
barplot(dangen95bar$dangen95, names.arg=NULL, col = dangen95bar$member,
log="y", main="Dan Genomic Positive Control\n95% similarity radius OTU's",
xlab='OTUs present in illumina positive control sample')
legend("top", legend=c('member', 'non-member'), fill=c(1,0))
```



Log transformed. OTUs with no match to our sanger sequences are colored white. Notice that most of the rare OTUs are not members of our original mock community. They are presumably contaminants or the results of tag-switching.

How many of these MCs have been split?

```
In [14]: dtab <- table(dangen95bar$MC)
dtab <- dtab[-1] #exclude non-members
dtab
```

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 18 19 20 22 23 24
4 1 1 1 1 1 1 5 1 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1 1
```

This table shows us that MC1,8,13,16 were split up by our bioinformatics. Let's visualize this, and compare to our other radius. For the plotter, get a matrix, each column representing our sanger-sequence sample numbers, and each row our otus in the illumina sample that matched to these sanger-sequences (MC#'s).

```
In [15]: stackmat <- matrix(nrow=max(dtab), ncol=24)
stackmat[] <- 0
stackmatnames <- paste("MC", 1:24, sep=' ')
colnames(stackmat) <- stackmatnames

for (i in 1:24){
  bb <- dangen95bar[dangen95bar$MC == i, , drop=FALSE]
  for (j in 1:nrow(bb)){stackmat[j,i] <- bb$dangen95[j]}
}
```

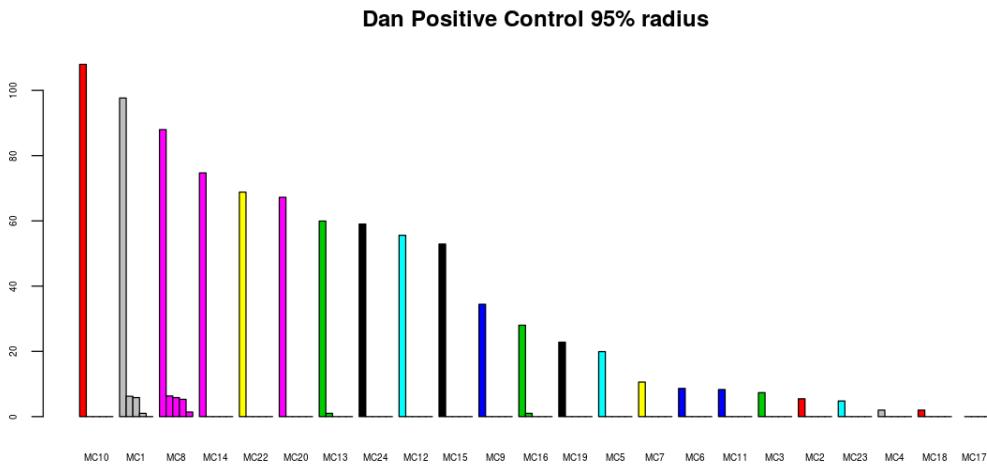
```
In [16]: stackmat <- stackmat[,-21] ## no MC21, wasn't put into MC mix
## sort by read abundances:
stackmat <- stackmat[,order(colSums(stackmat), decreasing = TRUE)]
stackmat
```

MC10	MC1	MC8	MC14	MC22	MC20	MC13	MC24	MC12	MC15	MC5	MC7	MC6
11662	9537	7742	5581	4737	4521	3593	3484	3090	2800	397	112	75
0	39	40	0	0	0	1	0	0	0	0	0	0
0	34	34	0	0	0	0	0	0	0	0	0	0
0	1	28	0	0	0	0	0	0	0	0	0	0
0	0	2	0	0	0	0	0	0	0	0	0	0

Now use this to plot our read abundances, to look at splitting of OTUs at this radius:

```
In [17]: ## make our colors from a palette. Randomize for better separation
bcols <- rep(sample(1:ncol(stackmat), replace=FALSE, size=23), each=nrow(stackmat))

par(cex.axis=.5)
## we can see our lower abundances better with sqrt:
barplot(sqrt(stackmat), beside = TRUE, col = bcards, main="Dan Positive Control 95% radii")
par(mfrow=c(1,1))
```



Square root transformed axis, so abundance differences are quite large. MC1 is *Phaeocryptopushaeumannii* and MC8 is a *Ramularia* sp.

#### ITS-only positive control

As a second positive control, we amplified the full ITS region of each our cultures from our mock community, then introduced equal concentrations of ITS PCR product into a single sample and put it in with the rest of the study. As per our genomic positive contol...

```
In [3]: aa95 <- subset_samples( biom95, sample_names( biom95 ) == 'PosI' )
danITSreads <- taxa_sums(aa95)[taxa_sums(aa95) > 0]
danITSreads <- sort(danITSreads, decreasing = TRUE)
sink('danITSreads.txt') ## read out these OTUs so we can use them in python env
names(danITSreads)
sink()
```

1. 'OTU256:Dc-X'
2. 'OTU119:Dc-PosG'
3. 'OTU106:Dc-PosG'
4. 'OTU386:Dc-PosG'
5. 'OTU306:Dc-PosG'
6. 'OTU264:Dc-PosG'
7. 'OTU407:Dc-PosG'
8. 'OTU191:Dc-PosG'
9. 'OTU270:Dc-PosG'
10. 'OTU271:Dc-PosG'
11. 'OTU220:Dc-PosG'
12. 'OTU258:Dc-PosG'
13. 'OTU526:Dc-PosG'
14. 'OTU235:Dc-PosG'
15. 'OTU560:Dc-PosG'
16. 'OTU164:Dc-PosG'
17. 'OTU608:Dc-PosG'
18. 'OTU733:Dc-PosG'
19. 'OTU358:Dc-PosG'
20. 'OTU826:Dc-PosG'
21. 'OTU1153:Dc-PosI'
22. 'OTU4210:Dc-PosG'
23. 'OTU3674:Dc-PosG'
24. 'OTU12510:PosG'
25. 'OTU3183:Dc-PosG'
26. 'OTU3723:Dc-PosG'
27. 'OTU4453:Dc-PosG'
28. 'OTU10694:Dc-PosI'
29. 'OTU77:1w'
30. 'OTU298:2w'
31. 'OTU87:17w'
32. 'OTU12521:Dc-PosG'
33. 'OTU8401:Dc-PosG'
34. 'OTU12516:PosI'
35. 'OTU1183:36w'
36. 'OTU98:2w'
37. 'OTU41:1w'
38. 'OTU249:1w'
39. 'OTU410:37w'
40. 'OTU84:38w'
41. 'OTU1520:30w'
42. 'OTU8984:Dc-PosG'
43. 'OTU8986:Dc-PosG'
44. 'OTU7914:Dc-PosG'
45. 'OTU9035:Dc-PosG'
46. 'OTU10988:PosG'
47. 'OTU9833:Dc-PosI'
48. 'OTU186:1w'
49. 'OTU471:5w'
50. 'OTU192:1w'
51. 'OTU28:2w'
52. 'OTU1216:1w'
53. 'OTU958:19w'
54. 'OTU338:1w'
55. 'OTU1292:32w'
56. 'OTU18:9w'
57. 'OTU6703:17w'
58. 'OTU2954:1w'
59. 'OTU161:2w'
60. 'OTU387:2w'
61. 'OTU2446:2w'
62. 'OTU1279:15w'
63. 'OTU2202:3w'
64. 'OTU5187:68w'
65. 'OTU250:4w'
66. 'OTU1599:9w'

67. 'OTU1973:131w' 68. 'OTU3711:17w' 69. 'OTU1612:18w' 70. 'OTU6688:19w' 71. 'OTU281:109w'  
72. 'OTU210:27w' 73. 'OTU2860:70w' 74. 'OTU839:57w' 75. 'OTU2567:130w' 76. 'OTU12503:PosG'

In [2]: cat danITS\_makeMCseq.py

```
mcseq= ("OTU256:Dc-X", "OTU119:Dc-PosG", "OTU106:Dc-PosG", "OTU386:Dc-PosG",  
"OTU306:Dc-PosG", "OTU264:Dc-PosG", "OTU407:Dc-PosG", "OTU191:Dc-PosG",  
"OTU270:Dc-PosG", "OTU271:Dc-PosG", "OTU220:Dc-PosG", "OTU258:Dc-PosG",  
"OTU526:Dc-PosG", "OTU235:Dc-PosG", "OTU560:Dc-PosG", "OTU164:Dc-PosG",  
"OTU608:Dc-PosG", "OTU733:Dc-PosG", "OTU358:Dc-PosG", "OTU826:Dc-PosG",  
"OTU1153:Dc-PosI", "OTU4210:Dc-PosG", "OTU3674:Dc-PosG", "OTU12510:PosG",  
"OTU3183:Dc-PosG", "OTU3723:Dc-PosG", "OTU4453:Dc-PosG", "OTU10694:Dc-PosI"  
"OTU77:1w", "OTU298:2w", "OTU87:17w", "OTU12521:Dc-PosG"  
"OTU8401:Dc-PosG", "OTU12516:PosI", "OTU1183:36w", "OTU98:2w",  
"OTU41:1w", "OTU249:1w", "OTU410:37w", "OTU84:38w",  
"OTU1520:30w", "OTU8984:Dc-PosG", "OTU8986:Dc-PosG", "OTU7914:Dc-PosG",  
"OTU9035:Dc-PosG", "OTU10988:PosG", "OTU9833:Dc-PosI", "OTU186:1w",  
"OTU471:5w", "OTU192:1w", "OTU28:2w", "OTU1216:1w",  
"OTU958:19w", "OTU338:1w", "OTU1292:32w", "OTU18:9w",  
"OTU6703:17w", "OTU2954:1w", "OTU161:2w", "OTU387:2w",  
"OTU2446:2w", "OTU1279:15w", "OTU2202:3w", "OTU5187:68w",  
"OTU250:4w", "OTU1599:9w", "OTU1973:131w", "OTU3711:17w",  
"OTU1612:18w", "OTU6688:19w", "OTU281:109w", "OTU210:27w",  
"OTU2860:70w", "OTU839:57w", "OTU2567:130w", "OTU12503:PosG")  
  
with open('otus_95_combo_nolb.fasta', 'r') as zoop:  
    refseq = zoop.readlines()  
  
with open('seqs_danITS95.fasta', 'w') as goop:  
    for j, otu in enumerate(mcseq):  
        for i, line in enumerate(refseq):  
            if otu in line:  
                goop.write(line)  
                goop.write(refseq[i+1])
```

In [3]: python3 danITS\_makeMCseq.py

In [4]: sed -i '/^>/ s/;size=.\*//' seqs\_danITS95.fasta

```
blastn -query seqs_danITS95.fasta -db mcsanger.fasta -out mcblast_danITS95.txt -num_des  
blastn -query seqs_danITS95.fasta -db mcsanger.fasta -out mcblast_danITS95.csv -outfmt 1  
  
sed '1 i\qseqid,sseqid,pident,length,mismatch,gapopen,qstart,qend,sstart,send,evalue,bit'  
sed 's/_ITS[1,4]_,/,/g' mcblast_danITS95.csv -i  
sed 's/Sample//g' mcblast_danITS95.csv -i
```

Back to R...

```
In [3]: blast <- read.csv("mcblast_danITS95.csv", stringsAsFactors=FALSE)
blast
```

qseqid	sseqid	pident	length	mismatch	gapopen	qstart	qend	sstart	send	evalu
OTU256:Dc-X	6	100.00	184	0	0	1	184	535	352	1e-96
OTU119:Dc-PosG	1	99.31	144	1	0	1	144	434	291	7e-73
OTU106:Dc-PosG	10	100.00	174	0	0	1	174	482	309	4e-91
OTU386:Dc-PosG	19	100.00	175	0	0	1	175	513	339	1e-91
OTU306:Dc-PosG	9	100.00	176	0	0	1	176	504	329	3e-92
OTU264:Dc-PosG	16	100.00	180	0	0	1	180	502	323	2e-94
OTU407:Dc-PosG	7	100.00	193	0	0	1	193	524	332	1e-10
OTU191:Dc-PosG	14	100.00	197	0	0	1	197	566	370	7e-10
OTU270:Dc-PosG	12	100.00	178	0	0	1	178	501	324	2e-93
OTU271:Dc-PosG	24	100.00	209	0	0	1	209	537	329	2e-11
OTU220:Dc-PosG	22	100.00	168	0	0	1	168	512	345	8e-88
OTU258:Dc-PosG	13	100.00	180	0	0	1	180	492	313	2e-94
OTU526:Dc-PosG	5	100.00	232	0	0	1	232	562	331	3e-12
OTU235:Dc-PosG	20	100.00	217	0	0	1	217	605	389	6e-11
OTU560:Dc-PosG	4	100.00	180	0	0	1	180	485	306	2e-94
OTU164:Dc-PosG	8	100.00	143	0	0	1	143	451	309	5e-74
OTU608:Dc-PosG	11	100.00	174	0	0	1	174	487	314	4e-91
OTU733:Dc-PosG	23	100.00	208	0	0	1	208	528	321	6e-11
OTU358:Dc-PosG	15	100.00	241	0	0	1	241	553	313	3e-12
OTU826:Dc-PosG	3	100.00	157	0	0	1	157	555	399	9e-82
OTU1153:Dc-PosI	18	100.00	361	0	0	1	361	761	401	0e+00
OTU4210:Dc-PosG	8	96.00	125	5	0	19	143	433	309	1e-55
OTU3674:Dc-PosG	1	95.24	126	6	0	19	144	416	291	1e-54
OTU12510:PosG	8	95.62	137	4	2	1	135	451	315	4e-60
OTU3183:Dc-PosG	8	94.89	137	5	2	1	135	451	315	2e-58
OTU3723:Dc-PosG	1	95.00	140	5	2	1	140	434	297	4e-60
OTU4453:Dc-PosG	2	100.00	143	0	0	1	143	463	321	5e-74
OTU12516:PosI	4	96.10	154	4	2	18	170	458	306	2e-69
OTU84:38w	9	100.00	31	0	0	27	57	530	500	1e-11
OTU8986:Dc-PosG	13	99.39	163	1	0	15	177	475	313	2e-83
OTU10988:PosG	8	100.00	118	0	0	19	136	426	309	4e-60
OTU2446:2w	19	79.88	169	22	7	12	174	501	339	2e-28
OTU1599:9w	1	90.22	92	8	1	37	128	459	369	6e-30
OTU6688:19w	11	89.22	167	15	3	3	167	485	320	4e-56
OTU12503:PosG	16	95.43	175	2	4	8	181	492	323	1e-76

```
In [4]: goodblast <- blast[blast$length > 100, ]
```

Check MC presence/absence...

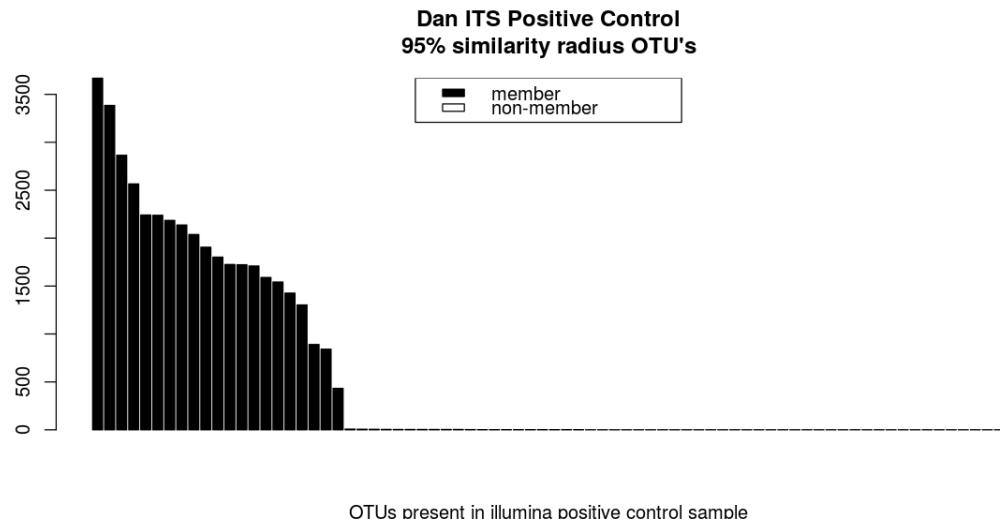
```
In [5]: 1:24 %in% goodblast$sseqid
```

1. TRUE
2. TRUE
3. TRUE
4. TRUE
5. TRUE
6. TRUE
7. TRUE
8. TRUE
9. TRUE
10. TRUE
11. TRUE
12. TRUE
13. TRUE
14. TRUE
15. TRUE
16. TRUE
17. FALSE
18. TRUE
19. TRUE
20. TRUE
21. FALSE
22. TRUE
23. TRUE
24. TRUE

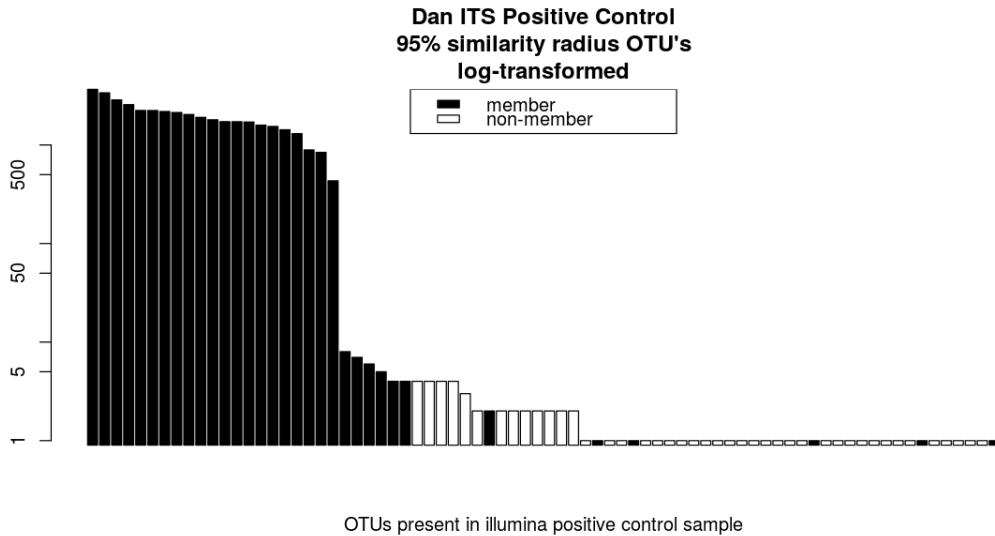
```
In [6]: sort(unique(goodblast$sseqid)); length(unique(goodblast$sseqid))
1. 1 2. 2 3. 3 4. 4 5. 5 6. 6 7. 7 8. 8 9. 9 10. 10 11. 11 12. 12 13. 13 14. 14 15. 15 16. 16 17. 18 18. 19
19. 20 20. 22 21. 23 22. 24
22
Still missing MC17.
```

```
In [12]: member <- names(danITS95) %in% goodblast$qseqid ## danITS95 created above
MC <- vector(length = length(danITS95))
danITS95bar <- data.frame(cbind(danITS95, member, MC))
for (i in 1:nrow(danITS95bar)){
  if (rownames(danITS95bar)[i] %in% goodblast$qseqid) {
    search <- grep(rownames(danITS95bar)[i], goodblast$qseqid)
    danITS95bar$MC[i] <- goodblast$sseqid[search]
  }
}
```

```
In [16]: options(repr.plot.width = 10, repr.plot.height = 5)
barplot(danITS95bar$danITS95, names.arg=NULL, col = danITS95bar$member,
        main="Dan ITS Positive Control\n95% similarity radius OTU's",
        xlab='OTUs present in illumina positive control sample')
legend("top", legend=c('member','non-member'), fill=c(1,0))
```



```
In [17]: options(repr.plot.width = 10, repr.plot.height = 5)
barplot(danITS95bar$danITS95, names.arg=NULL, col = danITS95bar$member,
        log="y", main="Dan ITS Positive Control\n95% similarity radius OTU's\nlog-transformed")
legend("top", legend=c('member','non-member'), fill=c(1,0))
```



Mostly intended members of the mock community were present... let's look at OTU splitting:

```
In [19]: dtab <- table(danITS95bar$MC)
        dtab <- dtab[-1] #exclude non-members
        dtab
```

```
1   2   3   4   5   6   7   8   9   10  11  12  13  14  15  16  18  19  20  22  23  24
3   1   1   2   1   1   1   5   1   1   2   1   2   1   2   1   1   2   1   2   1   1   1   1
```

Huh, seems like even more splitting in the ITS...

```
In [20]: stackmat <- matrix(nrow=max(dtab), ncol=24)
        stackmat[] <- 0
        stackmatnames <- paste("MC", 1:24, sep="")
        colnames(stackmat) <- stackmatnames

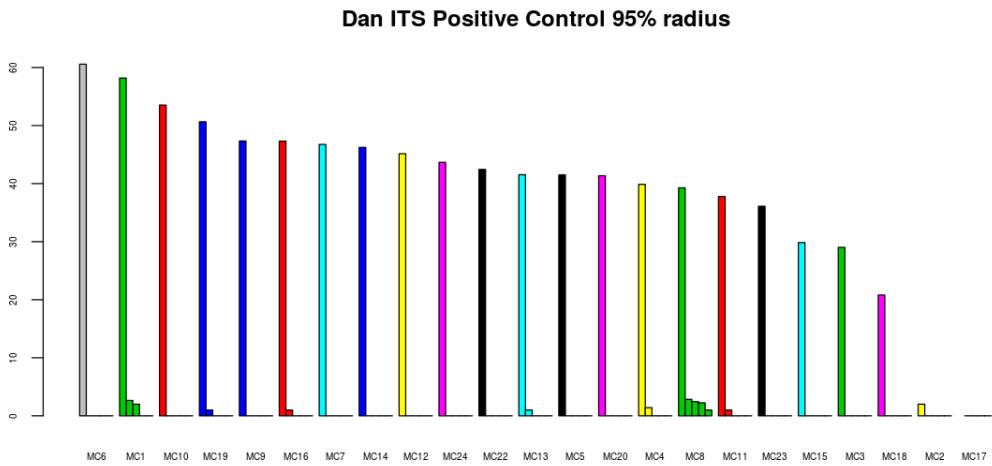
        for (i in 1:24){
            bb <- danITS95bar[danITS95bar$MC == i, , drop=FALSE]
            for (j in 1:nrow(bb)){stackmat[j,i] <- bb$danITS95[j]}
        }
```

```
In [21]: stackmat <- stackmat[,-21] ## no MC21, wasn't put into MC mix
        ## sort by read abundances:
        stackmat <- stackmat[,order(colSums(stackmat), decreasing = TRUE)]
        stackmat
```

MC6	MC1	MC10	MC19	MC9	MC16	MC7	MC14	MC12	MC24	MC20	MC4	MC8	M
3669	3385	2865	2566	2241	2239	2185	2137	2038	1906	1710	1590	1542	1
0	7	0	1	0	1	0	0	0	0	0	2	8	1
0	4	0	0	0	0	0	0	0	0	0	0	6	0
0	0	0	0	0	0	0	0	0	0	0	0	5	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0

```
In [22]: ## make our colors from a palette. Randomize for better separation
bcols <- rep(sample(1:ncol(stackmat), replace=FALSE, size=23), each=nrow(stackmat))

par(cex.axis=.5)
## we can see our lower abundances better with sqrt:
barplot(sqrt(stackmat), beside = TRUE, col = bcols, main="Dan ITS Positive Control 95%
par(mfrow=c(1,1))
```



Same basic story, as far as OTU splitting.

Cleanup using controls

We'll use our positive and negative controls to get estimates of rates of contamination and tag-switching, and institute cutoffs to reduce error from these sources. This is our pipeline for getting our reads ready to ask ecological questions.

Removing low abundance samples:

```
In [24]: sample_sums(biom95)[sample_sums(biom95) < 2000]
```

```
106leaf 8 112leafA 17 113leafA 78 113leafB 24 126leafB 20 18leaf 1648 74leaf 341 86leaf 1769
91leaf 12 Neg 1501
```

Looks like some of the leaf sites will be lost... Controls are low also, but we need to get rid of these anyway later.

```
In [37]: biom95eco <- subset_samples(biom95, SotC=='Sample')
biom95eco_hiread <- prune_samples(sample_sums(biom95eco)>=2000, biom95eco)
```

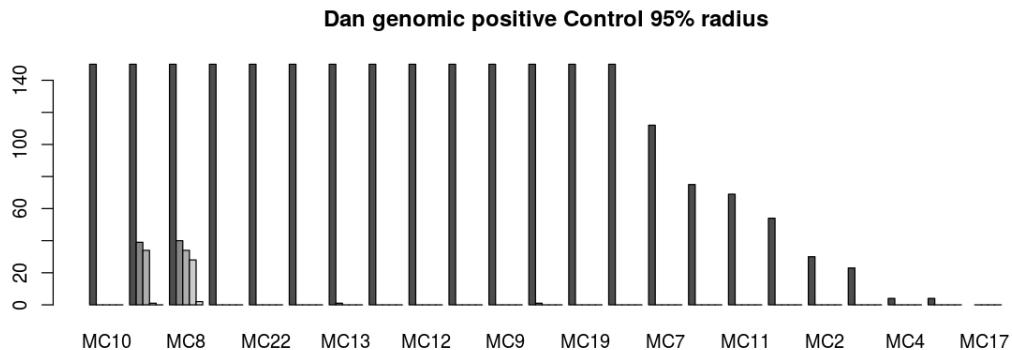
```
In [38]: biom95eco_hiread
```

```
phyloseq-class experiment-level object
otu_table()    OTU Table:           [ 11588 taxa and 214 samples ]
sample_data()  Sample Data:        [ 214 samples by 11 sample variables ]
tax_table()    Taxonomy Table:     [ 11588 taxa by 7 taxonomic ranks ]
```

Minimum abundances of observations and contaminant removal

I define an observation as the presence of a OTU within a sample, regardless of read abundance. But to be acknowledged as a "real" observation, it is reasonable to ask that an observation meet a certain abundance, or we can throw it out. To address spurious OTUs that result from OTU-splitting, we can define this abundance threshold by looking a little more closely at our genomic positive and negative controls:

```
In [33]: smShort <- stackmat ## rerun genomic control code above to get this.
smShort[smShort > 150] <- 150
#save(smShort, file='posGtrun.rda')
barplot(smShort, beside = TRUE, main="Dan genomic positive Control 95% radius")
par(mfrow=c(1,1))
```



```
In [32]: svg(file='outsplitting_genomic.svg')
smShort <- stackmat ## rerun genomic control code above to get this.
smShort[smShort > 150] <- 150
barplot(smShort, beside = TRUE, main="Dan genomic positive Control 95% radius")
par(mfrow=c(1,1))
dev.off()
```

**png: 2**

So to control for OTU splitting, looks like, in my positive control, most OTUs that are a result of splitting in the genomic positive control are at ~40 reads. For tag switching, if we look at the tagswitched OTUs in our negative control, we see that in our negative control that tag switching events seem to be ~100 reads or lower (Section ??):

```
In [102]: Neg95bar[Neg95bar$member==TRUE,]
```

	reads	member	MC	Genus	Species
OTU306:Dc-PosG	96	TRUE	9	NoID	NoID
OTU386:Dc-PosG	88	TRUE	19	Nemania	Nemania_aenea_SH217392.07FU
OTU164:Dc-PosG	51	TRUE	8	Mycosphaerella	Mycosphaerella_rubella_SH206848.07FU
OTU560:Dc-PosG	39	TRUE	4	Diaporthe	Diaporthe_cynaroidis_SH090428.07FU
OTU256:Dc-X	35	TRUE	6	Trametes	NoID
OTU119:Dc-PosG	1	TRUE	1	Phaeocryptopus	NoID
OTU220:Dc-PosG	1	TRUE	22	NoID	NoID
OTU235:Dc-PosG	1	TRUE	20	Psilocybe	Psilocybe_cyanescens_SH262625.07FU
OTU264:Dc-PosG	1	TRUE	16	Xylaria	NoID

But I don't want to cut off observations at ~100 reads, we would lose a lot of information. This would begin to be as intellectually dishonest as doing no removal at all, erring too far in excluding real information from our results. As per our previous analysis, it seems about require a minimum threshold of 60 reads per observation in our wood samples. This eliminates all the observed events of OTU splitting, and 7 out of 9 tag switch events in our negative control. It also keeps us in line with our previous analysis. This threshold of 60 reads does eliminate the 7 lowest-abundance members of our 22 detected members of the mock community, indicating a large loss of information. Always a trade-off.

As per our previous analysis, we'll scale up the number of leaf reads to be proportionate. To quote myself:

We have no positive controls from the leaf data, since Roo sequenced those in a separate study, so for now we'll assume that these samples have a +/- similar level of error from tag-switching. To figure the level of subtraction for tag-switching and splinter OTUs in his study, we'll subtract the same proportion of reads/library as the wood. Relative to the total number of reads (4,521,655) in the wood library, this is a reduction in each OTU/sample observation of  $60/4,521,655 = 0.000013269$  times the library size. If we multiply the size of the leaf library by this, we get  $(0.000013269) \times (10,751,750 \text{ reads}) = 142.6972$ , or 143 reads.

We'll round to 140 reads.

In addition, this is a good time to remove the contaminants we find in our controls, while the leaf and wood abundances are being considered separately. We're going to remove all of our mock community members from wherever they appear in the study in a downstream step anyway (see below), so we can treat our positive controls just like negatives here, removing all that we find.

Split the OTU abundance table into leaf and wood libraries:

```
In [204]: leafeco <- subset_samples(biom95eco_hiread, Library=='L')
woodeco <- subset_samples(biom95eco_hiread, Library=='W')
```

Wood reads:

```
In [205]: ## what are the maximums from our controls?:
```

```
woodControls <- prune_samples(sample_names(biom95)=='Neg' |
                                sample_names(biom95)=='PosG' |
                                sample_names(biom95)=='PosI',
```

```
biom95)
```

```
## get maximums
woodContam <- apply(otu_table(woodControls), 1, max)

## how many control OTUs are there?
sum(woodContam>0)
```

122

In [218]: `sort(woodContam[woodContam > 0])`

```
OTU8986:Dc-PosG 1 OTU7914:Dc-PosG 1 OTU186:1w 1 OTU471:5w 1 OTU64:1w 1
OTU192:1w 1 OTU409:4w 1 OTU152:1w 1 OTU28:2w 1 OTU1216:1w 1 OTU958:19w 1
OTU437:1w 1 OTU338:1w 1 OTU1292:32w 1 OTU5738:17w 1 OTU414:13w 1 OTU6703:17w 1
OTU655:1w 1 OTU2954:1w 1 OTU1075:100w 1 OTU161:2w 1 OTU391:2w 1 OTU2029:2w 1
OTU387:2w 1 OTU2446:2w 1 OTU1279:15w 1 OTU2202:3w 1 OTU5187:68w 1 OTU250:4w 1
OTU1524:3w 1 OTU7955:106w 1 OTU1973:131w 1 OTU5111:14w 1 OTU1124:36w 1
OTU7329:38w 1 OTU1466:9w 1 OTU925:133w 1 OTU1941:17w 1 OTU3711:17w 1 OTU1612:18w
1 OTU6688:19w 1 OTU1496:23w 1 OTU588:32w 1 OTU1888:25w 1 OTU281:109w 1
OTU210:27w 1 OTU2860:70w 1 OTU428:45w 1 OTU839:57w 1 OTU2567:130w 1 OTU972:130w 1
OTU1516:133w 1 OTU1866:133w 1 OTU12503:PosG 1 OTU8401:Dc-PosG 2 OTU10988:PosG 2
OTU12516:PosI 2 OTU9833:Dc-PosI 2 OTU1183:36w 2 OTU248:20w 2 OTU98:2w 2 OTU41:1w 2
OTU249:1w 2 OTU18:9w 2 OTU610:1w 2 OTU257:3w 2 OTU410:37w 2 OTU1520:30w 2
OTU2115:131w 2 OTU12502:PosG 2 OTU8984:Dc-PosG 3 OTU12521:Dc-PosG 3 OTU46:60w 3
OTU10694:Dc-PosI 4 OTU77:1w 4 OTU298:2w 4 OTU1599:9w 4 OTU2831:5w 4 OTU87:17w 4
OTU9035:Dc-PosG 5 OTU84:38w 12 OTU417:1w 15 OTU6852:Neg 20 OTU4210:Dc-PosG 28
OTU1432:2w 28 OTU1549:104w 28 OTU4453:Dc-PosG 30 OTU3674:Dc-PosG 34
OTU12510:PosG 34 OTU1444:49w 38 OTU3723:Dc-PosG 39 OTU3183:Dc-PosG 40
OTU2003:Neg 42 OTU1332:11w 44 OTU762:Dc-X 53 OTU1214:9w 53 OTU1747:11w 53
OTU891:1w 58 OTU187:Dc-PosG 85 OTU315:4w 162 OTU1153:Dc-PosI 433 OTU167:Dc-X 469
OTU826:Dc-PosG 842 OTU733:Dc-PosG 1303 OTU608:Dc-PosG 1427 OTU560:Dc-PosG 1590
OTU526:Dc-PosG 1723 OTU407:Dc-PosG 2185 OTU264:Dc-PosG 2239 OTU306:Dc-PosG 2241
OTU386:Dc-PosG 2566 OTU358:Dc-PosG 2800 OTU270:Dc-PosG 3090 OTU271:Dc-PosG 3484
OTU258:Dc-PosG 3593 OTU256:Dc-X 3669 OTU235:Dc-PosG 4521 OTU220:Dc-PosG 4737
OTU191:Dc-PosG 5581 OTU164:Dc-PosG 7742 OTU119:Dc-PosG 9537 OTU106:Dc-PosG 11662
```

In [206]: `## just checking...`

```
length(woodContam); all(names(woodContam) == rownames(otu_table(woodeco)))
```

11588

TRUE

```
In [207]: woodCotu <- otu_table(woodeco) - woodContam ## subtract contaminants
woodCotu <- woodCotu-60 ## subtract minimum threshold
woodCotu[woodCotu < 0] <- 0 ## bring negatives up to zero
woodeco.contam.rem <- woodeco ## make a copy, just in case
otu_table(woodeco.contam.rem) <- woodCotu ## insert the corrected OTU
```

Now leaf reads, subtract contaminants and minimum threshold:

In [208]: *## what are the maximums from our controls?:*

```
leafControls <- prune_samples(sample_names(leaf95) == 'leafNCA' |  
                                sample_names(leaf95) == 'leafNCB',  
                                leaf95)  
  
## get maximums  
leafContam <- apply(otu_table(leafControls), 1, max)  
  
## how many control OTUs are there?  
sum(leafContam > 0)
```

89

In [209]: *## just checking...*

```
length(leafContam); all(names(leafContam) == rownames(otu_table(leafeco)))
```

11588

TRUE

In [210]: leafCotu <- otu\_table(leafeco) - leafContam *## subtract contaminants*  
leafCotu <- leafCotu - 140 *## subtract minimum threshold*  
leafCotu[leafCotu < 0] <- 0 *## bring negatives up to zero*  
leafeco.contam.rem <- leafeco *## make a copy, just in case*  
otu\_table(leafeco.contam.rem) <- leafCotu *## insert the corrected OTU*

Okay, now reunite the leaf and wood reads:

In [211]: *## does "merging" work here to bring these back together?*

```
biom95.mend <- merge_phyloseq(woodeco.contam.rem, leafeco.contam.rem)
```

In [253]: *#save(biom95.mend, file="biom95.mend.rda")*

Run some checks:

In [243]: *## check to make sure structure is intact*

```
dim(otu_table(biom95eco_hiread))  
dim(otu_table(biom95.mend))
```

1. 11588 2. 214

1. 11588 2. 214

In [244]: *## row and column order changed after merging?*

```
all(rownames(otu_table(biom95.mend)) %in% rownames(otu_table(biom95eco_hiread)))  
all(colnames(otu_table(biom95.mend)) %in% colnames(otu_table(biom95eco_hiread)))  
  
all(rownames(otu_table(biom95eco_hiread)) %in% rownames(otu_table(biom95.mend)))  
all(colnames(otu_table(biom95eco_hiread)) %in% colnames(otu_table(biom95.mend))))
```

```
TRUE  
TRUE  
TRUE  
TRUE
```

```
In [245]: ## dup tables  
aa <- otu_table(biom95eco_hiread); bb <- otu_table(biom95.mend)  
## reorder columns so they match, for spot checking the matrices  
eco_reord <- aa[,order(colnames(aa))]; mend_reord <- bb[,order(colnames(bb))]
```

```
In [246]: ## spot check  
eco_reord[1:5,1:5]  
mend_reord[1:5,1:5] ## should be lower, more zeroes but no negs
```

	100leaf	100w	101leaf	101w	102leaf
OTU19:100leaf	81511	0	178	0	0
OTU108:100leaf	26184	0	0	0	0
OTU1:100leaf	2735	0	6795	0	586
OTU202:100leaf	3214	0	0	0	0
OTU426:100leaf	5943	0	0	0	0
	100leaf	100w	101leaf	101w	102leaf
OTU19:100leaf	81371	0	38	0	0
OTU108:100leaf	26044	0	0	0	0
OTU1:100leaf	2594	0	6654	0	445
OTU202:100leaf	3074	0	0	0	0
OTU426:100leaf	5803	0	0	0	0

```
In [110]: ## spot check  
eco_reord[1:5,1:5]  
mend_reord[1:5,1:5] ## should be lower, more zeroes but no negs
```

	100leaf	100w	101leaf	101w	102leaf
OTU19:100leaf	81511	0	178	0	0
OTU108:100leaf	26184	0	0	0	0
OTU1:100leaf	2735	0	6795	0	586
OTU202:100leaf	3214	0	0	0	0
OTU426:100leaf	5943	0	0	0	0
	100leaf	100w	101leaf	101w	102leaf
OTU19:100leaf	81371	0	38	0	0
OTU108:100leaf	26044	0	0	0	0
OTU1:100leaf	2595	0	6655	0	446
OTU202:100leaf	3074	0	0	0	0
OTU426:100leaf	5803	0	0	0	0

```
In [111]: ## more spot checks  
eco_reord[5140:5145,172:177]  
mend_reord[5140:5145,172:177]
```

	76w	77leaf	78leaf	79leaf	79w	7leaf
OTU1101:Dc-Neg	0	0	0	0	0	0
OTU1517:Dc-Neg	0	0	0	0	0	0
OTU351:Dc-Neg	6945	0	0	0	0	0
OTU1562:Dc-Neg	0	0	0	0	0	0
OTU1227:Dc-Neg	0	0	0	0	11	0
OTU1949:Dc-Neg	0	0	0	0	0	0
	76w	77leaf	78leaf	79leaf	79w	7leaf
OTU1101:Dc-Neg	0	0	0	0	0	0
OTU1517:Dc-Neg	0	0	0	0	0	0
OTU351:Dc-Neg	6885	0	0	0	0	0
OTU1562:Dc-Neg	0	0	0	0	0	0
OTU1227:Dc-Neg	0	0	0	0	0	0
OTU1949:Dc-Neg	0	0	0	0	0	0

```
In [82]: ## more spot checks
eco_reord[5140:5145,172:177]
mend_reord[5140:5145,172:177]
```

	76w	77leaf	78leaf	79leaf	79w	7leaf
OTU1101:Dc-Neg	0	0	0	0	0	0
OTU1517:Dc-Neg	0	0	0	0	0	0
OTU351:Dc-Neg	6945	0	0	0	0	0
OTU1562:Dc-Neg	0	0	0	0	0	0
OTU1227:Dc-Neg	0	0	0	0	11	0
OTU1949:Dc-Neg	0	0	0	0	0	0
	76w	77leaf	78leaf	79leaf	79w	7leaf
OTU1101:Dc-Neg	0	0	0	0	0	0
OTU1517:Dc-Neg	0	0	0	0	0	0
OTU351:Dc-Neg	6885	0	0	0	0	0
OTU1562:Dc-Neg	0	0	0	0	0	0
OTU1227:Dc-Neg	0	0	0	0	0	0
OTU1949:Dc-Neg	0	0	0	0	0	0

```
In [80]: ## more spot checks
eco_reord[1140:1150,120:130]
mend_reord[1140:1150,120:130]
```

	49w	4leaf	4w	50leaf	50w	51leaf	52leaf	52w	53leaf	54leaf	55leaf
OTU9024:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU10740:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU10278:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU9776:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU2355:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU9034:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU51:109leaf	0	0	0	13568	0	6203	0	0	0	496	0
OTU8420:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU8927:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU11707:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU1985:109leaf	0	0	0	0	0	0	0	0	0	0	0

	49w	4leaf	4w	50leaf	50w	51leaf	52leaf	52w	53leaf	54leaf	55leaf
OTU9024:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU10740:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU10278:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU9776:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU2355:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU9034:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU51:109leaf	0	0	0	13428	0	6063	0	0	0	356	0
OTU8420:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU8927:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU11707:109leaf	0	0	0	0	0	0	0	0	0	0	0
OTU1985:109leaf	0	0	0	0	0	0	0	0	0	0	0

To check our contaminant removal: wood otus that are contaminants should be 60 + highest abundance found in any of the three wood controls. For instance, OTU315:4w is a contaminant found in our negative control:

```
In [240]: otu_table(woodControls)[ "OTU315:4w" , ]
```

	Neg	PosG	PosI
OTU315:4w	162	0	0

This means that after our thresholds and subtraction of negative control abundances we should see a  $162+60=222$  read loss where ever OTU315:4w is observed in wood samples (or  $162+140=302$  read loss for leaves):

```
In [242]: aa <- t(rbind(eco_reord[ "OTU315:4w" , ],mend_reord[ "OTU315:4w" , ]))
colnames(aa) <- c('before cleanup', 'after cleanup')
aa[rowSums(aa) > 0, ]
```

	before cleanup	after cleanup
102w	18	0
10w	3	0
11w	16	0
121w	2	0
130w	14	0
14w	1	0
16w	1	0
17w	8	0
18w	1	0
19w	3612	3390
20w	76	0
21w	44	0
25w	21	0
26w	3	0
27w	2	0
28w	7	0
29w	131	0
30w	17	0
35w	24	0
3w	8	0
45w	2	0
49w	101	0
4w	16	0
50w	2	0
55w	3726	3504
56w	7	0
57w	16	0
5w	48	0
64w	1	0
68w	17	0
73w	11	0
75w	7	0
76w	1	0
89w	7	0
95w	14	0
99w	30	0

Looks like the cleanup worked as intended.

Remove all mock community OTUs

Unfortunately, it looks like our mock community bled out quite a lot into our other samples. So I can't really trust any OTUs that match to our positive controls, any observations of these OTUs could well be the result of tag switching. To get rid of them, we'll use the same basic method as above when identifying which OTUs were members of the mock community. But this time we'll blast our entire set of OTUs against our list of sanger sequences for the positive control. Any strong hits will be considered as possibly resulting from tag-switching, and will be removed.

```
In [1]: ## clean up our cluster names a little
sed '/>OTU/ s/;size=.*//g' otus_95_combo_nolb.fasta > otus_95.fasta
```

```
In [2]: blastn -query otus_95.fasta -db mcsanger.fasta -out mcblast_allOTUs.csv -outfmt 10 -max_
```

Read this into R, assign some column names.

```
In [4]: ## pick the biom tables back up into workspace
```

```
load("biom95.mend.rda")
```

```
load('biom95.rda')
```

```
In [8]: blast <- read.csv('mcblast_allOTUs.csv', header=FALSE)
```

```
blasthead <- c('qseqid','sseqid','pident','length','mismatch','gapopen','qstart','qend',  
colnames(blast) <- blasthead
```

```
In [10]: head(blast); dim(blast)
```

	qseqid	sseqid	pident	length	mismatch	gapopen	qstart	qend	ssta
OTU3:leafNotChim_102	Sample1_ITS4	87.67	146	16	1	1	146	307	
OTU10:leafNotChim_100	Sample15_ITS4	86.94	245	19	10	3	239	330	
OTU16:leafNotChim_100	Sample15_ITS4	86.90	229	16	10	20	240	347	
OTU20:leafNotChim_108	Sample16_ITS4	88.57	70	5	1	89	155	449	
OTU21:leafNotChim_112B	Sample1_ITS4	85.71	147	14	7	2	144	307	
OTU33:Dc-X	Sample16_ITS4	95.24	63	3	0	1	63	502	
1. 769 2. 12									

```
In [13]: ## keep only strong matches
```

```
goodblast <- blast[blast$pident > 95 & blast$length > 100,]
```

```
In [14]: nrow(blast)
```

```
nrow(goodblast)
```

769

54

769 otus matched somehow to an MC sequence, but only 54 OTUs are strong matches, so these are the ones to get rid of.

```
In [15]: ## which of our rownames (= OTU names) are not in this list of strong matches?
```

```
pcotus <- !(rownames(otu_table(biom95)) %in% goodblast$qseqid)
```

```
# keep only these:
```

```
biom95.mc.rem <- prune_taxa(pcotus, biom95.mend)
```

```
In [16]: #save(biom95.mc.rem, file='biom95.mc.rem.rda')
```

Check the losses in reads, samples, etc:

```
In [17]: biom95; biom95.mc.rem
```

```
phyloseq-class experiment-level object
otu_table()    OTU Table:           [ 11588 taxa and 232 samples ]
sample_data()  Sample Data:        [ 232 samples by 11 sample variables ]
tax_table()    Taxonomy Table:     [ 11588 taxa by 7 taxonomic ranks ]
```

```
phyloseq-class experiment-level object
otu_table()    OTU Table:           [ 11553 taxa and 214 samples ]
sample_data()  Sample Data:        [ 214 samples by 11 sample variables ]
tax_table()    Taxonomy Table:     [ 11553 taxa by 7 taxonomic ranks ]
```

```
In [18]: sum(sample_sums(biom95))
          sum(sample_sums(biom95.mc.rem))
```

15442054

13189412

2,252,642 reads removed by the cleanup from controls. How many observations were lost?

```
In [19]: sum(otu_table(biom95) > 0)
          sum(otu_table(biom95.mc.rem) > 0)
```

51625

9684

41941 observations removed, ~80% of all observations... that's a lot.

How many taxa removed entirely from the study?

```
In [29]: sum(rowSums(otu_table(biom95)) > 0)

          sum(rowSums(otu_table(biom95.mc.rem)) > 0)
```

11588

3330

That's a lot. Oh well. Onward.

Variance stabilization of reads

Apply DESeq2 algorithms

Let's stabilize the variance among our samples using DESeq2. Let's use Roo's old script for doing this, hope it still works after three years...

```
In [32]: DESeq_varstab <- function(phyloseq, design) {
  # phyloseq = the input phyloseq object that you want to get DESeq transformed counts from
  # design_variable = the design for the conversion to the DESeq object. must be in the form of a factor
  deseq.vst = NULL
  geo_Means = NULL
  phyloseq.DESeq = NULL
  # Convert to a DESeq object
  deseq = phyloseq_to_deseq2(phyloseq, design)
  # calculate geometric means prior to estimate size factors
  gm_mean = function(x, na.rm=TRUE){
    exp(sum(log(x[x > 0])), na.rm=na.rm) / length(x)
  }
  geo_Means = apply(counts(deseq), 1, gm_mean)
  # Check to see if any columns (samples) don't have any OTUs in them:
  if(sum(colSums(counts(deseq)) == 0) == 0) { # if all samples have taxa, go on
    # Now we step through the size factors, dispersions, and variance stabilizers
  }
}
```

```

deseq = estimateSizeFactors(deseq, geoMeans = geo_Means)
deseq = estimateDispersions(deseq) # long step
deseq.vst = getVarianceStabilizedData(deseq)
# replace negatives with zeros
deseq.vst[deseq.vst <0] <- 0
# add the variance stabilized otu numbers into the dataset:
otu_table(phyloseq) <- otu_table(deseq.vst, taxa_are_rows = TRUE)
# create a new object for the variance stabilized set
phyloseq -> phyloseq.DESeq
# And, filter any taxa that became 0s all the way across
phyloseq.DESeq = filter_taxa(phyloseq.DESeq, function(x) sum(x) > 0.1,
# return the new phyloseq object
return(phyloseq.DESeq)
} # end of IF loop
else {return("Error: your phyloseq object has samples with no taxa present.")}
} # end function

```

In [33]: `deseq95 <- DESeq_varstab(biom95.mc.rem, ~Library)`

converting counts to integer mode

Warning message in `DESeqDataSet(se, design = design, ignoreRank)`:

some variables in design formula are characters, converting to factors gene-wise dispersion estimation  
mean-dispersion relationship

final dispersion estimates

In [34]: `#save(deseq95, file='deseq95.rda')`

In [35]: `deseq95`

```

phyloseq-class experiment-level object
otu_table()    OTU Table:           [ 3327 taxa and 214 samples ]
sample_data()  Sample Data:        [ 214 samples by 11 sample variables ]
tax_table()    Taxonomy Table:     [ 3327 taxa by 7 taxonomic ranks ]

```

In [36]: `biom95.mc.rem`

```

phyloseq-class experiment-level object
otu_table()    OTU Table:           [ 11553 taxa and 214 samples ]
sample_data()  Sample Data:        [ 214 samples by 11 sample variables ]
tax_table()    Taxonomy Table:     [ 11553 taxa by 7 taxonomic ranks ]

```

Looks like a big drop in taxa, but not really. Most of these taxa-rows, were emptied out during our cleanup from the controls. We really only lost  $3330 - 3327 = 3$  otus.

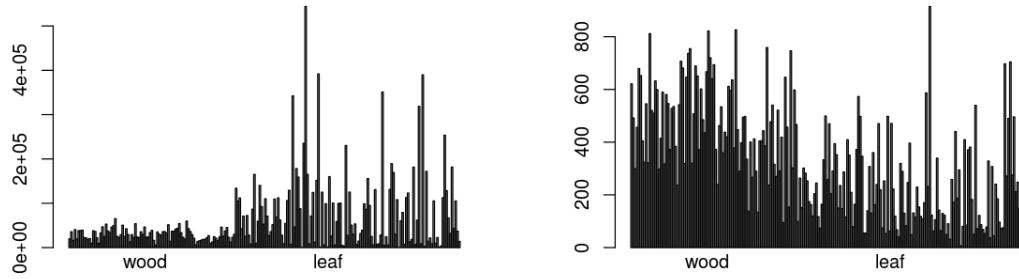
Compare pre/post variance-stabilization distributions

Overall distributions of reads among leaf/wood samples:

```
In [38]: par(mfrow = c(1,2))
options(repr.plot.width = 10, repr.plot.height = 4)

barplot(sample_sums(biom95.mc.rem), axisnames=FALSE)
mtext(text='wood', side = 1, at=50)
mtext(text='leaf', side = 1, at=170)

barplot(sample_sums(deseq95), axisnames=FALSE)
mtext(text='wood', side = 1, at=50)
mtext(text='leaf', side = 1, at=170)
```



### Positive controls after variance stabilization

```
In [72]: aa <- biom95
#bb <- DESeq_varstab(biom95, ~Library)

aaOTU <- (otu_table(aa)[, 'PosG']) ## our control, before
aaOTU <- aaOTU[aaOTU != 0]

bbOTU <- (otu_table(bb)[, 'PosG']) ## our control, after
bbOTU <- bbOTU[bbOTU != 0]

rownames(aaOTU) %in% rownames(bbOTU) ## did we lose any?

## tag them for plotting and comparison
lost <- !(rownames(aaOTU) %in% rownames(bbOTU))
aalost <- cbind(aaOTU, lost)
aalost <- aalost[order(aalost[, 'PosG'], decreasing = TRUE),]
bbOTU <- bbOTU[order(bbOTU[, 'PosG'], decreasing = TRUE),]
```

1. TRUE
2. TRUE
3. TRUE
4. TRUE
5. TRUE
6. TRUE
7. TRUE
8. TRUE
9. TRUE
10. TRUE
11. TRUE
12. TRUE
13. TRUE
14. TRUE
15. TRUE
16. TRUE
17. TRUE
18. TRUE
19. TRUE
20. TRUE
21. TRUE
22. TRUE
23. TRUE
24. TRUE
25. TRUE
26. TRUE
27. TRUE
28. TRUE
29. TRUE
30. TRUE

31. TRUE 32. TRUE 33. TRUE 34. TRUE 35. TRUE 36. TRUE 37. TRUE 38. TRUE 39. TRUE 40. TRUE  
 41. TRUE 42. TRUE 43. TRUE 44. TRUE 45. TRUE 46. TRUE 47. TRUE 48. TRUE 49. TRUE 50. TRUE  
 51. TRUE 52. TRUE 53. TRUE 54. TRUE 55. TRUE 56. TRUE 57. TRUE 58. TRUE

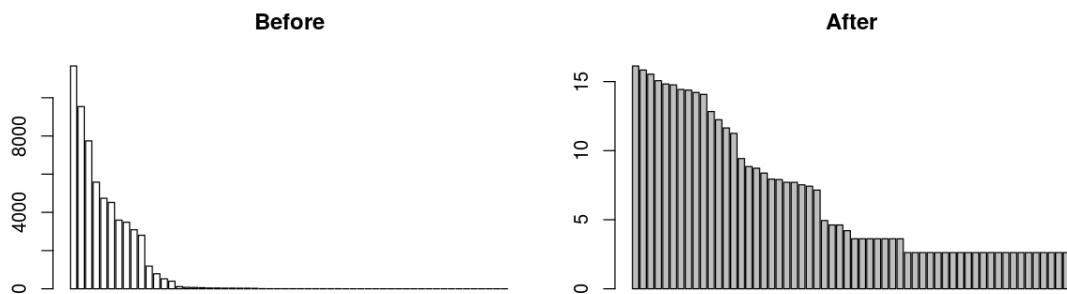
```
In [73]: par(mfrow=c(1,2))
par(mar=c(2,2,4,2))
par(oma=c(0,0,4,0))

barplot(aalost[, 'PosG'],
        col=aalost[, 'lost'],
        names.arg = '',
        main = 'Before',
        )

barplot(as.vector(bbOTU[, 'PosG']),
        main = 'After',
        )

mtext("Genomic Postive control before and after DeSeq",
      side = 3,
      line = 1,
      outer = TRUE,
      cex = 2,
      )
```

Genomic Positive control before and after DeSeq



How about the ITS-only positive control? Use the same pipeline...

```
In [75]: #aa <- biom95
#bb <- DESeq_varstab(biom95, ~Library)
aaOTU <- (otu_table(aa)[, 'PosI']) ## our control, before
aaOTU <- aaOTU[aaOTU != 0]

bbOTU <- (otu_table(bb)[, 'PosI']) ## our control, after
```

```

bbOTU <- bbOTU[bbOTU != 0]

rownames(aaOTU) %in% rownames(bbOTU) ## did we lose any?

## tag them for plotting and comparison
lost <- !(rownames(aaOTU) %in% rownames(bbOTU))
aalost <- cbind(aaOTU,lost)
aalost <- aalost[order(aalost[, 'PosI'], decreasing = TRUE),]
bbOTU <- bbOTU[order(bbOTU[, 'PosI'], decreasing = TRUE),]

```

1. TRUE 2. TRUE 3. TRUE 4. TRUE 5. TRUE 6. TRUE 7. TRUE 8. TRUE 9. TRUE 10. TRUE  
 11. TRUE 12. TRUE 13. TRUE 14. TRUE 15. TRUE 16. TRUE 17. TRUE 18. TRUE 19. TRUE 20. TRUE  
 21. TRUE 22. TRUE 23. TRUE 24. TRUE 25. TRUE 26. TRUE 27. TRUE 28. TRUE 29. TRUE 30. TRUE  
 31. TRUE 32. TRUE 33. TRUE 34. TRUE 35. TRUE 36. TRUE 37. TRUE 38. TRUE 39. TRUE 40. TRUE  
 41. TRUE 42. TRUE 43. TRUE 44. TRUE 45. TRUE 46. TRUE 47. TRUE 48. TRUE 49. TRUE 50. TRUE  
 51. TRUE 52. TRUE 53. TRUE 54. TRUE 55. TRUE 56. TRUE 57. TRUE 58. TRUE 59. TRUE 60. TRUE  
 61. TRUE 62. TRUE 63. TRUE 64. TRUE 65. TRUE 66. TRUE 67. TRUE 68. TRUE 69. TRUE 70. TRUE  
 71. TRUE 72. TRUE 73. TRUE 74. TRUE 75. TRUE 76. TRUE

```

In [76]: par(mfrow=c(1,2))
          par(mar=c(2,2,4,2))
          par(oma=c(0,0,4,0))

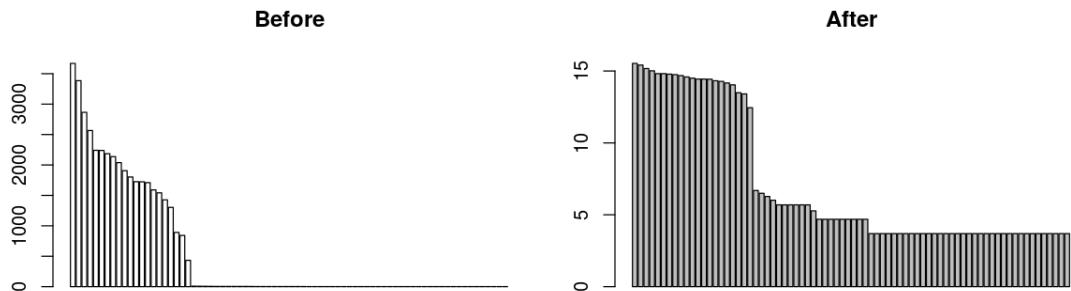
          barplot(aalost[, 'PosI'],
                    col=aalost[, 'lost'],
                    names.arg = '',
                    main = 'Before',
                    )

          barplot(as.vector(bbOTU[, 'PosI']),
                    main = 'After',
                    )

          mtext("ITS Postive control before and after DeSeq",
                 side = 3,
                 line = 1,
                 outer = TRUE,
                 cex = 2,
                 )

```

## ITS Positive control before and after DeSeq



If we get time, color these in with membership of the mock community.  
Ecological samples after variance stabilization

```
In [54]: load('biom95.mc.rem.rda')
aa <- biom95.mc.rem
load('deseq95.rda')
bb <- deseq95

aaOTU <- (otu_table(aa)[, '1w'])
aaOTU <- aaOTU[aaOTU != 0]
bbOTU <- (otu_table(bb)[, '1w'])
bbOTU <- bbOTU[bbOTU != 0]
rownames(aaOTU) %in% rownames(bbOTU)
## let's tag these, plot them
lost <- !(rownames(aaOTU) %in% rownames(bbOTU))*1
aalost <- cbind(aaOTU, lost)
aalost <- aalost[order(aalost[, '1w'], decreasing = TRUE),]
bbOTU <- bbOTU[order(bbOTU[, '1w'], decreasing = TRUE),]
```

1. TRUE 2. TRUE 3. TRUE 4. TRUE 5. TRUE 6. TRUE 7. TRUE 8. TRUE 9. TRUE 10. TRUE  
11. TRUE 12. TRUE 13. TRUE 14. TRUE 15. TRUE 16. TRUE 17. TRUE 18. TRUE 19. TRUE 20. TRUE  
21. TRUE 22. TRUE 23. TRUE 24. TRUE 25. TRUE 26. TRUE 27. TRUE 28. TRUE 29. TRUE 30. TRUE  
31. TRUE 32. TRUE 33. TRUE 34. TRUE 35. TRUE 36. TRUE 37. TRUE 38. TRUE 39. TRUE 40. TRUE  
41. TRUE 42. TRUE 43. TRUE 44. TRUE 45. TRUE 46. TRUE 47. TRUE 48. TRUE 49. TRUE 50. TRUE  
51. TRUE 52. TRUE 53. TRUE 54. TRUE 55. TRUE 56. TRUE 57. TRUE 58. TRUE 59. TRUE 60. TRUE  
61. TRUE 62. TRUE 63. TRUE 64. TRUE 65. TRUE 66. TRUE 67. TRUE 68. TRUE 69. TRUE 70. TRUE  
71. TRUE 72. TRUE 73. TRUE 74. TRUE 75. TRUE 76. TRUE 77. TRUE 78. TRUE 79. TRUE 80. TRUE  
81. TRUE 82. TRUE 83. TRUE 84. TRUE 85. TRUE

```
In [56]: par(mfrow=c(1,2))
par(mar=c(2,2,4,2))
par(oma=c(0,0,4,0))
```

```

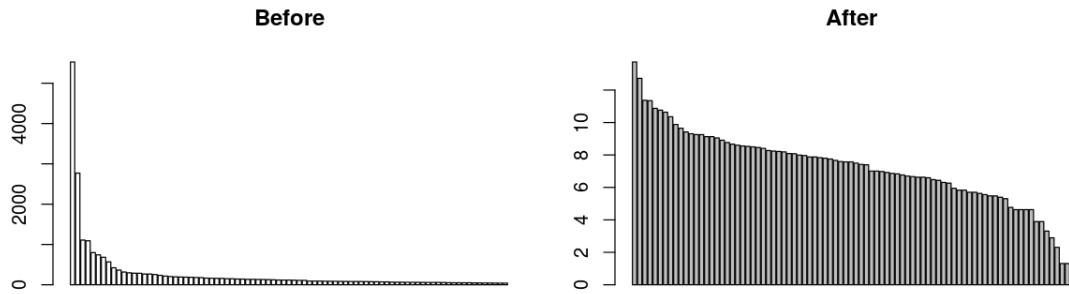
barplot(aalost[, '1w']+40, ## add some to make visible
        col=aalost[, 'lost'],
        names.arg = '',
        main = 'Before',
        )

barplot(as.vector(bbOTU[, '1w']),
        main = 'After',
        )

mtext("Ecological wood sample before and after DeSeq",
      side = 3,
      line = 1,
      outer = TRUE,
      cex = 2,
      )

```

## Ecological wood sample before and after DeSeq



Huh, unlike in our last analysis, we didn't lose our rare OTUs. I'm not sure why/what was different in the case. Generally, though, the transformation here is pretty clean. As before, deseq's algorithmns improve but don't completely ameliorate the issues of uneven amplification and sequencer detection.

Ideally, one should be able to train an algorithm based on performance with a series of positive controls. I can imagine this being part of the future of hi-throughput metabarcoding pipelines.

NMS/PERMANOVA Ordinations

Wood vs. Leaf libraries

```
In [80]: ## get two otu tables out, one for all wood samples, one for all leaf samples:
woodOTU <- otu_table(subset_samples(deseq95, Library='W'))
wood <- t(woodOTU@.Data) ## transpose so samples are rows

leafOTU <- otu_table(subset_samples(deseq95, Library='L'))
leaf <- t(leafOTU@.Data) ## transpose so samples are rows
```

```

In [81]: dim(wood); dim(leaf) ## note this
          nmsmat <- rbind(wood, leaf)

1. 91 2. 3327
1. 123 2. 3327

In [85]: save(nmsmat, file='nmsmat.rda')

In [82]: WvsL <- metaMDS(nmsmat)

Wisconsin double standardization
Run 0 stress 0.0001791434
Run 1 stress 0.0002137209
... Procrustes: rmse 0.0001871009 max resid 0.002720356
... Similar to previous best
Run 2 stress 0.0001897826
... Procrustes: rmse 0.0002596393 max resid 0.003777302
... Similar to previous best
Run 3 stress 0.0002119864
... Procrustes: rmse 2.941868e-05 max resid 0.0003928597
... Similar to previous best
Run 4 stress 0.0002037484
... Procrustes: rmse 2.253602e-05 max resid 0.0002837903
... Similar to previous best
Run 5 stress 0.0001947748
... Procrustes: rmse 7.080397e-05 max resid 0.001015304
... Similar to previous best
Run 6 stress 0.0001851663
... Procrustes: rmse 0.000251969 max resid 0.003664053
... Similar to previous best
Run 7 stress 0.0001863611
... Procrustes: rmse 0.0002514439 max resid 0.003658659
... Similar to previous best
Run 8 stress 0.0001810524
... Procrustes: rmse 0.0002002467 max resid 0.00290918
... Similar to previous best
Run 9 stress 0.0001733723
... New best solution
... Procrustes: rmse 0.0002069196 max resid 0.003008232
... Similar to previous best
Run 10 stress 0.0001788233
... Procrustes: rmse 0.0001326841 max resid 0.0019226
... Similar to previous best
Run 11 stress 0.0001751423
... Procrustes: rmse 0.0001709301 max resid 0.00248209
... Similar to previous best
Run 12 stress 0.0001997238
... Procrustes: rmse 2.283373e-05 max resid 0.0002803192
... Similar to previous best

```

```

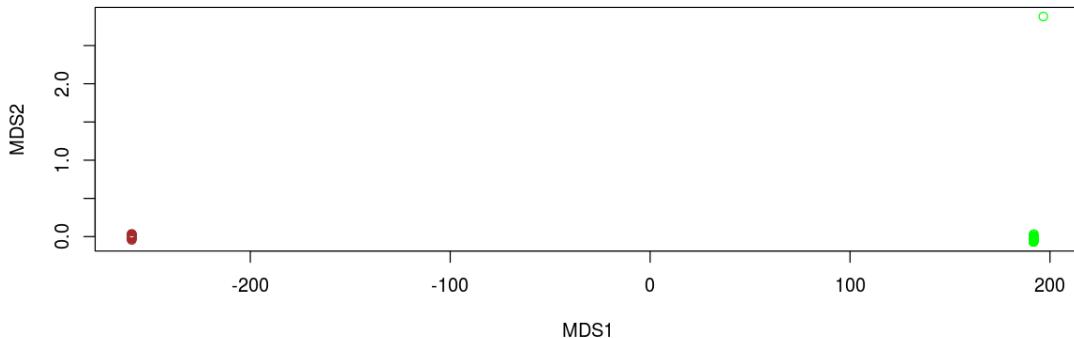
Run 13 stress 0.0001823538
... Procrustes: rmse 0.0001957423 max resid 0.002838862
... Similar to previous best
Run 14 stress 0.0001663224
... New best solution
... Procrustes: rmse 7.493005e-05 max resid 0.001080864
... Similar to previous best
Run 15 stress 0.0001671187
... Procrustes: rmse 7.849875e-05 max resid 0.00113247
... Similar to previous best
Run 16 stress 0.0002009742
... Procrustes: rmse 0.000119317 max resid 0.001728107
... Similar to previous best
Run 17 stress 0.00019731
... Procrustes: rmse 2.572417e-05 max resid 0.0003460788
... Similar to previous best
Run 18 stress 0.0001843638
... Procrustes: rmse 0.000155249 max resid 0.002257569
... Similar to previous best
Run 19 stress 0.0001991182
... Procrustes: rmse 2.880919e-05 max resid 0.0003949238
... Similar to previous best
Run 20 stress 0.0001940288
... Procrustes: rmse 0.0002059825 max resid 0.002994173
... Similar to previous best
*** Solution reached

```

Warning message in metaMDS(nmsmat):  
 stress is (nearly) zero: you may have insufficient data

```
In [83]: ## make a color vector, leaves green, wood brown:
  color <- NULL
  color[1:91] <- 'brown'
  color[92:214] <- 'green'
```

```
In [84]: plot(WvsL$points, col=color)
```



Weird, we have complete separation, unlike last time... Cutoffs too stringent? Look at this without the outlier...

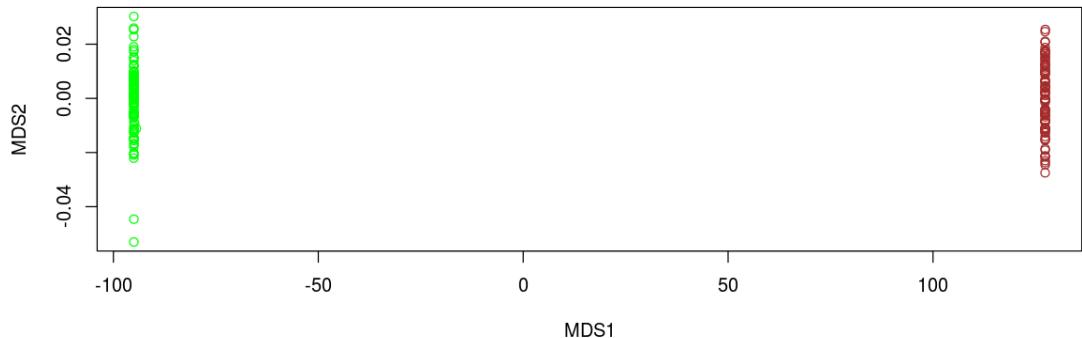
```
In [ ]: identify(WvsL$points) ##181

In [86]: nmsmat_noOut <- nmsmat[-181,]
          WvsL_noOut <- metaMDS(nmsmat_noOut)
          color <- NULL
          color[1:91] <- 'brown'
          color[92:214] <- 'green'
          plot(WvsL_noOut$points, col=color)

Wisconsin double standardization
Run 0 stress 8.954737e-05
Run 1 stress 9.190804e-05
... Procrustes: rmse 3.928785e-05 max resid 0.0004408048
... Similar to previous best
Run 2 stress 9.379337e-05
... Procrustes: rmse 3.602598e-05 max resid 0.0003702779
... Similar to previous best
Run 3 stress 9.147567e-05
... Procrustes: rmse 3.263903e-05 max resid 0.0003767296
... Similar to previous best
Run 4 stress 8.765039e-05
... New best solution
... Procrustes: rmse 2.178718e-05 max resid 0.0001101781
... Similar to previous best
Run 5 stress 8.773564e-05
... Procrustes: rmse 3.303455e-05 max resid 0.000338349
... Similar to previous best
Run 6 stress 9.959286e-05
... Procrustes: rmse 3.305523e-05 max resid 0.0003628473
... Similar to previous best
```

```
Run 7 stress 9.758524e-05
... Procrustes: rmse 3.523646e-05 max resid 0.0003298133
... Similar to previous best
Run 8 stress 9.685904e-05
... Procrustes: rmse 3.613546e-05 max resid 0.0004043215
... Similar to previous best
Run 9 stress 9.122151e-05
... Procrustes: rmse 2.222369e-05 max resid 6.818163e-05
... Similar to previous best
Run 10 stress 8.06415e-05
... New best solution
... Procrustes: rmse 3.406082e-05 max resid 0.000414843
... Similar to previous best
Run 11 stress 9.751769e-05
... Procrustes: rmse 6.139747e-05 max resid 0.0007810606
... Similar to previous best
Run 12 stress 9.802037e-05
... Procrustes: rmse 3.5533e-05 max resid 0.0003937992
... Similar to previous best
Run 13 stress 8.753637e-05
... Procrustes: rmse 5.371636e-05 max resid 0.0007021068
... Similar to previous best
Run 14 stress 9.370536e-05
... Procrustes: rmse 1.6256e-05 max resid 0.0001076227
... Similar to previous best
Run 15 stress 9.862719e-05
... Procrustes: rmse 5.977791e-05 max resid 0.0007828993
... Similar to previous best
Run 16 stress 8.373466e-05
... Procrustes: rmse 3.920683e-05 max resid 0.0004795359
... Similar to previous best
Run 17 stress 8.656762e-05
... Procrustes: rmse 3.398172e-05 max resid 0.000369271
... Similar to previous best
Run 18 stress 9.267289e-05
... Procrustes: rmse 1.886256e-05 max resid 7.686701e-05
... Similar to previous best
Run 19 stress 8.838017e-05
... Procrustes: rmse 4.11886e-05 max resid 0.0005268706
... Similar to previous best
Run 20 stress 9.032688e-05
... Procrustes: rmse 5.425464e-05 max resid 0.0007003611
... Similar to previous best
*** Solution reached
```

Warning message in metaMDS(nmsmat\_noOut):  
stress is (nearly) zero: you may have insufficient data



Hmm. Total separation... Our last analysis showed some overlap... the only difference I can think of is that I cleaned the negatives separately. Apparently it made a difference.

Do we see any sort of difference if we transform our data to presence/absence?

```
In [89]: nmsmat.PA <- nmsmat_noOut
          nmsmat.PA[nmsmat.PA > 0] <- 1
          WvsL.PA <- metaMDS(nmsmat.PA)

Run 0 stress 9.075196e-05
Run 1 stress 9.119895e-05
... Procrustes: rmse 2.578939e-05 max resid 9.191718e-05
... Similar to previous best
Run 2 stress 8.633431e-05
... New best solution
... Procrustes: rmse 3.826129e-05 max resid 0.0003420187
... Similar to previous best
Run 3 stress 9.589004e-05
... Procrustes: rmse 3.221965e-05 max resid 0.0002469573
... Similar to previous best
Run 4 stress 9.096515e-05
... Procrustes: rmse 3.41114e-05 max resid 0.0003903983
... Similar to previous best
Run 5 stress 8.86225e-05
... Procrustes: rmse 2.104221e-05 max resid 0.0001564683
... Similar to previous best
Run 6 stress 9.54635e-05
... Procrustes: rmse 4.057168e-05 max resid 0.0004528917
... Similar to previous best
Run 7 stress 9.080032e-05
... Procrustes: rmse 4.615923e-05 max resid 0.0005363398
... Similar to previous best
Run 8 stress 8.31665e-05
... New best solution
```

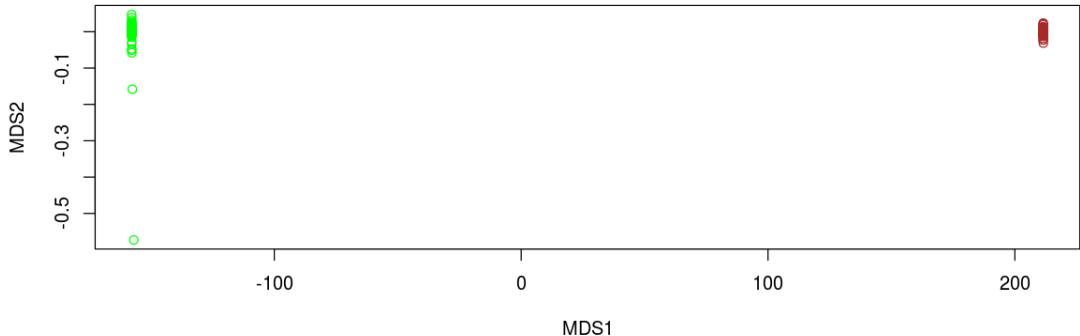
```

... Procrustes: rmse 3.373823e-05 max resid 0.0003856375
... Similar to previous best
Run 9 stress 9.105911e-05
... Procrustes: rmse 2.83769e-05 max resid 0.0003234669
... Similar to previous best
Run 10 stress 8.992835e-05
... Procrustes: rmse 4.363141e-05 max resid 0.0005350718
... Similar to previous best
Run 11 stress 8.63618e-05
... Procrustes: rmse 3.850697e-05 max resid 0.0004973861
... Similar to previous best
Run 12 stress 9.556513e-05
... Procrustes: rmse 2.868351e-05 max resid 0.0003184438
... Similar to previous best
Run 13 stress 9.794221e-05
... Procrustes: rmse 2.356721e-05 max resid 0.0001937728
... Similar to previous best
Run 14 stress 9.042516e-05
... Procrustes: rmse 2.414163e-05 max resid 0.0002467038
... Similar to previous best
Run 15 stress 8.834618e-05
... Procrustes: rmse 2.145298e-05 max resid 0.0001742086
... Similar to previous best
Run 16 stress 8.636309e-05
... Procrustes: rmse 5.031913e-05 max resid 0.0006647162
... Similar to previous best
Run 17 stress 9.162188e-05
... Procrustes: rmse 4.437512e-05 max resid 0.000571679
... Similar to previous best
Run 18 stress 8.885708e-05
... Procrustes: rmse 3.008939e-05 max resid 0.0003440576
... Similar to previous best
Run 19 stress 8.991465e-05
... Procrustes: rmse 1.731948e-05 max resid 5.576528e-05
... Similar to previous best
Run 20 stress 9.574878e-05
... Procrustes: rmse 3.137099e-05 max resid 0.0003400765
... Similar to previous best
*** Solution reached

```

Warning message in metaMDS(nmsmat.PA):  
 stress is (nearly) zero: you may have insufficient data

In [90]: `plot(WvsL.PA$points, col=color)`



Pretty similar...

Wood vs. Leaf endophyte comparison, look for shared species

```
In [3]: #load('deseq95.rda')
wood <- subset_samples(deseq95, Library=="W")
leaf <- subset_samples(deseq95, Library=="L")
```

How many OTUs are detected in each?

```
In [7]: sum(rowSums(otu_table(leaf)) > 0); sum(rowSums(otu_table(wood)) > 0)
```

1302

2025

Huh, many more wood OTUs.

```
In [94]: ## reduce these OTU tables to non-zero rows:
woodOTU <- otu_table(wood) ## otus are rows
woodspp <- rownames(woodOTU[rowSums(woodOTU)>0,])
```

```
In [ ]: ## do the same for the leaves:
```

```
leafOTU <- otu_table(leaf)
leafspp <- rownames(leafOTU[rowSums(leafOTU)>0,])
```

```
In [99]: leafspp
```

1. 'OTU19:100leaf' 2. 'OTU108:100leaf' 3. 'OTU1:100leaf' 4. 'OTU202:100leaf'
5. 'OTU426:100leaf' 6. 'OTU10:100leaf' 7. 'OTU17:100leaf' 8. 'OTU429:100leaf'
9. 'OTU15:100leaf' 10. 'OTU27:100leaf' 11. 'OTU355:100leaf' 12. 'OTU22:101leaf'
13. 'OTU91:100leaf' 14. 'OTU570:100leaf' 15. 'OTU406:100leaf' 16. 'OTU181:100leaf'
17. 'OTU520:100leaf' 18. 'OTU73:100leaf' 19. 'OTU159:100leaf' 20. 'OTU794:16leaf'
21. 'OTU297:100leaf' 22. 'OTU252:117leaf' 23. 'OTU1199:41leaf' 24. 'OTU951:14leaf'
25. 'OTU412:102leaf' 26. 'OTU78:100leaf' 27. 'OTU226:100leaf' 28. 'OTU282:100leaf'
29. 'OTU468:107leaf' 30. 'OTU209:101leaf' 31. 'OTU368:100leaf' 32. 'OTU381:66leaf'

33. 'OTU883:107leaf' 34. 'OTU239:100leaf' 35. 'OTU1006:100leaf' 36. 'OTU1938:130leaf'  
 37. 'OTU1495:100leaf' 38. 'OTU93:100leaf' 39. 'OTU1071:100leaf' 40. 'OTU263:100leaf'  
 41. 'OTU371:110leaf' 42. 'OTU2975:100leaf' 43. 'OTU319:100leaf' 44. 'OTU303:111leaf'  
 45. 'OTU1159:103leaf' 46. 'OTU613:100leaf' 47. 'OTU1134:100leaf' 48. 'OTU1182:100leaf'  
 49. 'OTU360:100leaf' 50. 'OTU70:100leaf' 51. 'OTU1690:100leaf' 52. 'OTU247:17leaf'  
 53. 'OTU16:100leaf' 54. 'OTU45:100leaf' 55. 'OTU1731:100leaf' 56. 'OTU584:100leaf'  
 57. 'OTU53:107leaf' 58. 'OTU591:100leaf' 59. 'OTU1657:104leaf' 60. 'OTU1224:101leaf'  
 61. 'OTU1151:51leaf' 62. 'OTU668:100leaf' 63. 'OTU280:100leaf' 64. 'OTU736:100leaf'  
 65. 'OTU514:125leaf' 66. 'OTU1543:100leaf' 67. 'OTU1169:100leaf' 68. 'OTU61:100leaf'  
 69. 'OTU890:100leaf' 70. 'OTU97:102leaf' 71. 'OTU540:14leaf' 72. 'OTU1140:100leaf'  
 73. 'OTU3:102leaf' 74. 'OTU1816:105leaf' 75. 'OTU94:114leaf' 76. 'OTU2267:100leaf'  
 77. 'OTU635:100leaf' 78. 'OTU2082:16leaf' 79. 'OTU199:101leaf' 80. 'OTU74:101leaf'  
 81. 'OTU262:101leaf' 82. 'OTU110:101leaf' 83. 'OTU797:101leaf' 84. 'OTU667:101leaf'  
 85. 'OTU284:101leaf' 86. 'OTU1586:101leaf' 87. 'OTU446:101leaf' 88. 'OTU42:101leaf'  
 89. 'OTU729:116leaf' 90. 'OTU1830:101leaf' 91. 'OTU1927:101leaf' 92. 'OTU1047:101leaf'  
 93. 'OTU4:101leaf' 94. 'OTU2005:101leaf' 95. 'OTU372:117leaf' 96. 'OTU340:103leaf'  
 97. 'OTU2175:101leaf' 98. 'OTU194:119leaf' 99. 'OTU462:101leaf' 100. 'OTU457:101leaf'  
 101. 'OTU124:101leaf' 102. 'OTU405:101leaf' 103. 'OTU545:47leaf' 104. 'OTU436:102leaf'  
 105. 'OTU43:101leaf' 106. 'OTU2259:101leaf' 107. 'OTU346:101leaf' 108. 'OTU846:101leaf'  
 109. 'OTU115:101leaf' 110. 'OTU2066:101leaf' 111. 'OTU666:120leaf' 112. 'OTU8842:131leaf'  
 113. 'OTU511:101leaf' 114. 'OTU705:114leaf' 115. 'OTU824:120leaf' 116. 'OTU343:103leaf'  
 117. 'OTU1144:101leaf' 118. 'OTU546:101leaf' 119. 'OTU116:117leaf' 120. 'OTU261:102leaf'  
 121. 'OTU549:132leaf' 122. 'OTU218:102leaf' 123. 'OTU677:15leaf' 124. 'OTU663:110leaf'  
 125. 'OTU1841:116leaf' 126. 'OTU946:101leaf' 127. 'OTU583:13leaf' 128. 'OTU831:101leaf'  
 129. 'OTU3181:107leaf' 130. 'OTU3042:41leaf' 131. 'OTU1142:101leaf' 132. 'OTU454:101leaf'  
 133. 'OTU1213:101leaf' 134. 'OTU378:101leaf' 135. 'OTU1813:117leaf' 136. 'OTU1436:101leaf'  
 137. 'OTU940:103leaf' 138. 'OTU788:104leaf' 139. 'OTU8836:14leaf' 140. 'OTU2103:15leaf'  
 141. 'OTU744:104leaf' 142. 'OTU681:101leaf' 143. 'OTU5825:114leaf' 144. 'OTU717:128leaf'  
 145. 'OTU489:125leaf' 146. 'OTU182:30leaf' 147. 'OTU448:1leaf' 148. 'OTU1576:14leaf'  
 149. 'OTU2497:131leaf' 150. 'OTU5:102leaf' 151. 'OTU109:102leaf' 152. 'OTU54:102leaf'  
 153. 'OTU100:102leaf' 154. 'OTU173:13leaf' 155. 'OTU422:62leaf' 156. 'OTU1088:102leaf'  
 157. 'OTU812:102leaf' 158. 'OTU89:102leaf' 159. 'OTU1025:3leaf' 160. 'OTU660:102leaf'  
 161. 'OTU3689:102leaf' 162. 'OTU644:102leaf' 163. 'OTU244:115leaf' 164. 'OTU79:102leaf'  
 165. 'OTU2372:102leaf' 166. 'OTU1318:1leaf' 167. 'OTU208:103leaf' 168. 'OTU1359:130leaf'  
 169. 'OTU1554:102leaf' 170. 'OTU2:102leaf' 171. 'OTU1963:102leaf' 172. 'OTU1064:75leaf'  
 173. 'OTU311:102leaf' 174. 'OTU470:103leaf' 175. 'OTU852:102leaf' 176. 'OTU65:102leaf'  
 177. 'OTU956:103leaf' 178. 'OTU440:102leaf' 179. 'OTU309:117leaf' 180. 'OTU330:102leaf'  
 181. 'OTU1734:102leaf' 182. 'OTU572:102leaf' 183. 'OTU1654:2leaf' 184. 'OTU509:102leaf'  
 185. 'OTU86:102leaf' 186. 'OTU268:102leaf' 187. 'OTU48:102leaf' 188. 'OTU538:112leafB'  
 189. 'OTU13:102leaf' 190. 'OTU14:102leaf' 191. 'OTU255:102leaf' 192. 'OTU24:102leaf'  
 193. 'OTU200:102leaf' 194. 'OTU769:102leaf' 195. 'OTU620:110leaf' 196. 'OTU1045:102leaf'  
 197. 'OTU3582:14leaf' 198. 'OTU401:4leaf' 199. 'OTU375:102leaf' 200. 'OTU653:102leaf'  
 201. 'OTU970:65leaf' 202. 'OTU479:104leaf' 203. 'OTU2625:23leaf' 204. 'OTU771:2leaf'  
 205. 'OTU1204:103leaf' 206. 'OTU212:103leaf' 207. 'OTU1460:50leaf' 208. 'OTU2656:103leaf'  
 209. 'OTU692:103leaf' 210. 'OTU242:103leaf' 211. 'OTU1781:103leaf' 212. 'OTU1439:103leaf'  
 213. 'OTU2230:103leaf' 214. 'OTU605:103leaf' 215. 'OTU1594:103leaf' 216. 'OTU602:103leaf'  
 217. 'OTU11:103leaf' 218. 'OTU349:103leaf' 219. 'OTU398:103leaf' 220. 'OTU1420:103leaf'  
 221. 'OTU789:103leaf' 222. 'OTU3130:103leaf' 223. 'OTU3356:103leaf' 224. 'OTU582:118leaf'

225. 'OTU3142:103leaf' 226. 'OTU529:103leaf' 227. 'OTU2102:94leaf' 228. 'OTU664:103leaf'  
 229. 'OTU1474:103leaf' 230. 'OTU1535:118leaf' 231. 'OTU25:103leaf' 232. 'OTU785:103leaf'  
 233. 'OTU670:103leaf' 234. 'OTU1001:103leaf' 235. 'OTU813:103leaf' 236. 'OTU1477:27leaf'  
 237. 'OTU3006:103leaf' 238. 'OTU2616:103leaf' 239. 'OTU615:103leaf' 240. 'OTU1661:103leaf'  
 241. 'OTU678:131leaf' 242. 'OTU2075:103leaf' 243. 'OTU4316:103leaf' 244. 'OTU606:103leaf'  
 245. 'OTU685:117leaf' 246. 'OTU332:103leaf' 247. 'OTU408:103leaf' 248. 'OTU763:103leaf'  
 249. 'OTU1665:117leaf' 250. 'OTU929:103leaf' 251. 'OTU1089:81leaf' 252. 'OTU1481:88leaf'  
 253. 'OTU714:103leaf' 254. 'OTU523:103leaf' 255. 'OTU809:103leaf' 256. 'OTU954:103leaf'  
 257. 'OTU62:112leafB' 258. 'OTU561:103leaf' 259. 'OTU1097:16leaf' 260. 'OTU2290:103leaf'  
 261. 'OTU12:22leaf' 262. 'OTU1953:125leaf' 263. 'OTU5684:68leaf' 264. 'OTU651:71leaf'  
 265. 'OTU2762:103leaf' 266. 'OTU259:103leaf' 267. 'OTU2083:103leaf' 268. 'OTU31:105leaf'  
 269. 'OTU1350:104leaf' 270. 'OTU601:130leaf' 271. 'OTU515:103leaf' 272. 'OTU1164:17leaf'  
 273. 'OTU1400:50leaf' 274. 'OTU1571:15leaf' 275. 'OTU8:105leaf' 276. 'OTU39:103leaf'  
 277. 'OTU878:96leaf' 278. 'OTU1093:130leaf' 279. 'OTU88:115leaf' 280. 'OTU1382:44leaf'  
 281. 'OTU1302:104leaf' 282. 'OTU75:104leaf' 283. 'OTU2383:104leaf' 284. 'OTU2885:104leaf'  
 285. 'OTU1450:104leaf' 286. 'OTU718:104leaf' 287. 'OTU2836:104leaf' 288. 'OTU3336:104leaf'  
 289. 'OTU1990:23leaf' 290. 'OTU1429:27leaf' 291. 'OTU814:125leaf' 292. 'OTU376:104leaf'  
 293. 'OTU990:114leaf' 294. 'OTU643:116leaf' 295. 'OTU1672:104leaf' 296. 'OTU67:104leaf'  
 297. 'OTU1208:104leaf' 298. 'OTU229:104leaf' 299. 'OTU2440:96leaf' 300. 'OTU223:104leaf'  
 301. 'OTU1035:114leaf' 302. 'OTU2824:52leaf' 303. 'OTU2956:51leaf' 304. 'OTU1331:104leaf'  
 305. 'OTU1123:104leaf' 306. 'OTU730:104leaf' 307. 'OTU1559:66leaf' 308. 'OTU2924:41leaf'  
 309. 'OTU1648:104leaf' 310. 'OTU1201:17leaf' 311. 'OTU4083:105leaf' 312. 'OTU2737:105leaf'  
 313. 'OTU617:105leaf' 314. 'OTU1529:105leaf' 315. 'OTU1149:105leaf' 316. 'OTU1761:105leaf'  
 317. 'OTU554:105leaf' 318. 'OTU1738:105leaf' 319. 'OTU1575:119leaf' 320. 'OTU1518:105leaf'  
 321. 'OTU236:105leaf' 322. 'OTU2234:117leaf' 323. 'OTU170:105leaf' 324. 'OTU1253:14leaf'  
 325. 'OTU505:107leaf' 326. 'OTU1412:105leaf' 327. 'OTU536:105leaf' 328. 'OTU1242:105leaf'  
 329. 'OTU23:119leaf' 330. 'OTU758:105leaf' 331. 'OTU776:22leaf' 332. 'OTU1005:81leaf'  
 333. 'OTU1410:105leaf' 334. 'OTU2676:105leaf' 335. 'OTU1014:118leaf' 336. 'OTU1468:105leaf'  
 337. 'OTU34:116leaf' 338. 'OTU3152:117leaf' 339. 'OTU1172:105leaf' 340. 'OTU713:105leaf'  
 341. 'OTU1835:105leaf' 342. 'OTU525:112leafB' 343. 'OTU50:117leaf' 344. 'OTU818:105leaf'  
 345. 'OTU2285:105leaf' 346. 'OTU1358:51leaf' 347. 'OTU36:105leaf' 348. 'OTU2153:131leaf'  
 349. 'OTU230:107leaf' 350. 'OTU882:50leaf' 351. 'OTU552:107leaf' 352. 'OTU2490:107leaf'  
 353. 'OTU834:107leaf' 354. 'OTU537:107leaf' 355. 'OTU912:107leaf' 356. 'OTU2163:107leaf'  
 357. 'OTU1043:115leaf' 358. 'OTU1155:1leaf' 359. 'OTU442:111leaf' 360. 'OTU1305:108leaf'  
 361. 'OTU1106:108leaf' 362. 'OTU7214:40leaf' 363. 'OTU743:107leaf' 364. 'OTU704:108leaf'  
 365. 'OTU2007:108leaf' 366. 'OTU1624:131leaf' 367. 'OTU2945:108leaf' 368. 'OTU3239:108leaf'  
 369. 'OTU1237:110leaf' 370. 'OTU969:108leaf' 371. 'OTU485:68leaf' 372. 'OTU30:116leaf'  
 373. 'OTU2740:108leaf' 374. 'OTU1409:108leaf' 375. 'OTU795:121leaf' 376. 'OTU1196:108leaf'  
 377. 'OTU59:111leaf' 378. 'OTU427:125leaf' 379. 'OTU483:111leaf' 380. 'OTU1061:109leaf'  
 381. 'OTU640:109leaf' 382. 'OTU1464:109leaf' 383. 'OTU1176:109leaf' 384. 'OTU2879:109leaf'  
 385. 'OTU3382:109leaf' 386. 'OTU1365:109leaf' 387. 'OTU993:109leaf' 388. 'OTU2355:109leaf'  
 389. 'OTU51:109leaf' 390. 'OTU190:112leafA' 391. 'OTU47:110leaf' 392. 'OTU32:110leaf'  
 393. 'OTU359:110leaf' 394. 'OTU1187:110leaf' 395. 'OTU793:110leaf' 396. 'OTU1258:110leaf'  
 397. 'OTU445:17leaf' 398. 'OTU2495:110leaf' 399. 'OTU721:110leaf' 400. 'OTU760:110leaf'  
 401. 'OTU1087:51leaf' 402. 'OTU416:110leaf' 403. 'OTU369:110leaf' 404. 'OTU2638:51leaf'  
 405. 'OTU1406:111leaf' 406. 'OTU215:110leaf' 407. 'OTU1567:110leaf' 408. 'OTU80:110leaf'  
 409. 'OTU183:110leaf' 410. 'OTU1195:110leaf' 411. 'OTU1133:43leaf' 412. 'OTU938:110leaf'  
 413. 'OTU1667:50leaf' 414. 'OTU1362:110leaf' 415. 'OTU193:47leaf' 416. 'OTU1383:110leaf'

417. 'OTU953:110leaf' 418. 'OTU26:110leaf' 419. 'OTU274:119leaf' 420. 'OTU71:115leaf'  
 421. 'OTU1355:111leaf' 422. 'OTU135:117leaf' 423. 'OTU1652:110leaf' 424. 'OTU314:111leaf'  
 425. 'OTU474:111leaf' 426. 'OTU927:111leaf' 427. 'OTU2637:111leaf' 428. 'OTU1171:75leaf'  
 429. 'OTU753:111leaf' 430. 'OTU451:111leaf' 431. 'OTU2156:47leaf' 432. 'OTU464:111leaf'  
 433. 'OTU810:115leaf' 434. 'OTU708:111leaf' 435. 'OTU1945:111leaf' 436. 'OTU1003:111leaf'  
 437. 'OTU1678:111leaf' 438. 'OTU472:111leaf' 439. 'OTU2700:111leaf' 440. 'OTU1803:113leafA'  
 441. 'OTU1255:111leaf' 442. 'OTU2042:111leaf' 443. 'OTU2139:41leaf' 444. 'OTU930:30leaf'  
 445. 'OTU694:112leafA' 446. 'OTU1007:116leaf' 447. 'OTU21:112leafB' 448. 'OTU498:112leafB'  
 449. 'OTU2752:112leafB' 450. 'OTU221:112leafB' 451. 'OTU1161:112leafB' 452. 'OTU35:112leafB'  
 453. 'OTU344:112leafB' 454. 'OTU847:112leafB' 455. 'OTU1306:112leafB' 456. 'OTU52:112leafB'  
 457. 'OTU1364:112leafB' 458. 'OTU353:119leaf' 459. 'OTU57:113leafB' 460. 'OTU496:112leafB'  
 461. 'OTU3432:116leaf' 462. 'OTU1873:113leafA' 463. 'OTU2618:113leafA' 464. 'OTU6470:120leaf'  
 465. 'OTU960:113leafA' 466. 'OTU1260:113leafA' 467. 'OTU2387:117leaf' 468. 'OTU2580:113leafB'  
 469. 'OTU1200:40leaf' 470. 'OTU614:121leaf' 471. 'OTU2124:114leaf' 472. 'OTU1198:114leaf'  
 473. 'OTU2623:114leaf' 474. 'OTU1637:128leaf' 475. 'OTU460:114leaf' 476. 'OTU111:115leaf'  
 477. 'OTU169:115leaf' 478. 'OTU347:115leaf' 479. 'OTU3094:115leaf' 480. 'OTU480:115leaf'  
 481. 'OTU2380:115leaf' 482. 'OTU177:115leaf' 483. 'OTU7:115leaf' 484. 'OTU2884:115leaf'  
 485. 'OTU138:115leaf' 486. 'OTU1573:115leaf' 487. 'OTU1021:115leaf' 488. 'OTU83:115leaf'  
 489. 'OTU701:20leaf' 490. 'OTU2840:20leaf' 491. 'OTU2130:20leaf' 492. 'OTU2019:115leaf'  
 493. 'OTU1522:14leaf' 494. 'OTU796:50leaf' 495. 'OTU862:50leaf' 496. 'OTU1920:130leaf'  
 497. 'OTU727:115leaf' 498. 'OTU2159:20leaf' 499. 'OTU1282:115leaf' 500. 'OTU1343:115leaf'  
 501. 'OTU1784:20leaf' 502. 'OTU2180:20leaf' 503. 'OTU802:115leaf' 504. 'OTU1776:115leaf'  
 505. 'OTU2306:59leaf' 506. 'OTU1222:115leaf' 507. 'OTU1390:115leaf' 508. 'OTU6:115leaf'  
 509. 'OTU143:15leaf' 510. 'OTU1297:127leaf' 511. 'OTU680:1leaf' 512. 'OTU557:116leaf'  
 513. 'OTU1723:116leaf' 514. 'OTU2061:1leaf' 515. 'OTU1993:32leaf' 516. 'OTU1615:116leaf'  
 517. 'OTU2169:116leaf' 518. 'OTU1885:50leaf' 519. 'OTU1704:116leaf' 520. 'OTU1146:116leaf'  
 521. 'OTU682:116leaf' 522. 'OTU1991:57leaf' 523. 'OTU556:116leaf' 524. 'OTU1023:116leaf'  
 525. 'OTU749:116leaf' 526. 'OTU487:116leaf' 527. 'OTU2045:116leaf' 528. 'OTU858:117leaf'  
 529. 'OTU777:116leaf' 530. 'OTU715:69leaf' 531. 'OTU1076:116leaf' 532. 'OTU2607:116leaf'  
 533. 'OTU302:117leaf' 534. 'OTU1793:117leaf' 535. 'OTU362:117leaf' 536. 'OTU1694:117leaf'  
 537. 'OTU2878:75leaf' 538. 'OTU2288:117leaf' 539. 'OTU3062:117leaf' 540. 'OTU2126:117leaf'  
 541. 'OTU1281:117leaf' 542. 'OTU1192:13leaf' 543. 'OTU2018:117leaf' 544. 'OTU1254:129leaf'  
 545. 'OTU944:117leaf' 546. 'OTU2028:129leaf' 547. 'OTU1127:117leaf' 548. 'OTU1726:117leaf'  
 549. 'OTU1262:90leaf' 550. 'OTU1483:117leaf' 551. 'OTU987:117leaf' 552. 'OTU764:117leaf'  
 553. 'OTU961:117leaf' 554. 'OTU2782:117leaf' 555. 'OTU1469:117leaf' 556. 'OTU1148:117leaf'  
 557. 'OTU3763:117leaf' 558. 'OTU1986:117leaf' 559. 'OTU1709:117leaf' 560. 'OTU2499:117leaf'  
 561. 'OTU2046:117leaf' 562. 'OTU1333:117leaf' 563. 'OTU2176:117leaf' 564. 'OTU2279:127leaf'  
 565. 'OTU2087:13leaf' 566. 'OTU1117:117leaf' 567. 'OTU2307:29leaf' 568. 'OTU1312:4leaf'  
 569. 'OTU999:117leaf' 570. 'OTU1564:117leaf' 571. 'OTU2604:117leaf' 572. 'OTU29:117leaf'  
 573. 'OTU1962:51leaf' 574. 'OTU2895:51leaf' 575. 'OTU731:118leaf' 576. 'OTU38:14leaf'  
 577. 'OTU153:132leaf' 578. 'OTU316:120leaf' 579. 'OTU2553:118leaf' 580. 'OTU1272:51leaf'  
 581. 'OTU2266:118leaf' 582. 'OTU1298:13leaf' 583. 'OTU936:118leaf' 584. 'OTU724:63leaf'  
 585. 'OTU1321:119leaf' 586. 'OTU870:118leaf' 587. 'OTU232:77leaf' 588. 'OTU2821:28leaf'  
 589. 'OTU877:121leaf' 590. 'OTU1357:51leaf' 591. 'OTU888:119leaf' 592. 'OTU1924:75leaf'  
 593. 'OTU567:75leaf' 594. 'OTU433:67leaf' 595. 'OTU55:129leaf' 596. 'OTU1011:22leaf'  
 597. 'OTU1129:4leaf' 598. 'OTU612:63leaf' 599. 'OTU1246:14leaf' 600. 'OTU2767:51leaf'  
 601. 'OTU366:125leaf' 602. 'OTU879:71leaf' 603. 'OTU456:14leaf' 604. 'OTU354:119leaf'  
 605. 'OTU832:119leaf' 606. 'OTU1274:119leaf' 607. 'OTU148:129leaf' 608. 'OTU3493:119leaf'

609. 'OTU3000:119leaf' 610. 'OTU3349:119leaf' 611. 'OTU2528:119leaf' 612. 'OTU1854:119leaf'  
 613. 'OTU1113:119leaf' 614. 'OTU592:41leaf' 615. 'OTU9780:39leaf' 616. 'OTU1538:119leaf'  
 617. 'OTU898:119leaf' 618. 'OTU2200:23leaf' 619. 'OTU2297:119leaf' 620. 'OTU2679:119leaf'  
 621. 'OTU1555:122leaf' 622. 'OTU3003:55leaf' 623. 'OTU804:120leaf' 624. 'OTU139:120leaf'  
 625. 'OTU492:129leaf' 626. 'OTU2324:120leaf' 627. 'OTU738:120leaf' 628. 'OTU1112:120leaf'  
 629. 'OTU3141:121leaf' 630. 'OTU2001:63leaf' 631. 'OTU1462:121leaf' 632. 'OTU1116:38leaf'  
 633. 'OTU3035:16leaf' 634. 'OTU1642:8leaf' 635. 'OTU476:121leaf' 636. 'OTU1424:122leaf'  
 637. 'OTU1812:122leaf' 638. 'OTU851:23leaf' 639. 'OTU1284:122leaf' 640. 'OTU123:122leaf'  
 641. 'OTU2733:122leaf' 642. 'OTU1239:26leaf' 643. 'OTU1775:122leaf' 644. 'OTU1892:122leaf'  
 645. 'OTU1351:50leaf' 646. 'OTU390:122leaf' 647. 'OTU207:125leaf' 648. 'OTU3765:49leaf'  
 649. 'OTU901:14leaf' 650. 'OTU1498:125leaf' 651. 'OTU2173:125leaf' 652. 'OTU2479:125leaf'  
 653. 'OTU741:125leaf' 654. 'OTU1696:4leaf' 655. 'OTU2331:125leaf' 656. 'OTU1623:90leaf'  
 657. 'OTU1887:88leaf' 658. 'OTU2407:125leaf' 659. 'OTU1083:57leaf' 660. 'OTU1178:125leaf'  
 661. 'OTU6397:30leaf' 662. 'OTU3200:125leaf' 663. 'OTU1799:125leaf' 664. 'OTU1515:125leaf'  
 665. 'OTU1470:125leaf' 666. 'OTU118:125leaf' 667. 'OTU1977:125leaf' 668. 'OTU761:125leaf'  
 669. 'OTU3033:125leaf' 670. 'OTU2436:125leaf' 671. 'OTU1488:125leaf' 672. 'OTU1294:125leaf'  
 673. 'OTU1718:125leaf' 674. 'OTU1413:126leafA' 675. 'OTU1861:126leafA' 676. 'OTU1328:126leafA'  
 677. 'OTU382:126leafA' 678. 'OTU910:14leaf' 679. 'OTU1079:126leafA' 680. 'OTU1619:14leaf'  
 681. 'OTU1901:126leafA' 682. 'OTU1570:126leafB' 683. 'OTU2776:127leaf' 684. 'OTU1824:127leaf'  
 685. 'OTU875:127leaf' 686. 'OTU2664:128leaf' 687. 'OTU1875:16leaf' 688. 'OTU1997:131leaf'  
 689. 'OTU581:64leaf' 690. 'OTU656:127leaf' 691. 'OTU2109:127leaf' 692. 'OTU2381:98leaf'  
 693. 'OTU2092:72leaf' 694. 'OTU392:36leaf' 695. 'OTU1314:130leaf' 696. 'OTU1211:128leaf'  
 697. 'OTU2299:128leaf' 698. 'OTU3089:128leaf' 699. 'OTU3039:128leaf' 700. 'OTU1929:128leaf'  
 701. 'OTU632:128leaf' 702. 'OTU1532:128leaf' 703. 'OTU2817:128leaf' 704. 'OTU1319:14leaf'  
 705. 'OTU2819:56leaf' 706. 'OTU1712:128leaf' 707. 'OTU624:14leaf' 708. 'OTU1453:5leaf'  
 709. 'OTU1080:129leaf' 710. 'OTU1062:129leaf' 711. 'OTU3197:129leaf' 712. 'OTU4749:129leaf'  
 713. 'OTU134:129leaf' 714. 'OTU1039:129leaf' 715. 'OTU1608:129leaf' 716. 'OTU1249:14leaf'  
 717. 'OTU1820:69leaf' 718. 'OTU1675:129leaf' 719. 'OTU4173:129leaf' 720. 'OTU4800:129leaf'  
 721. 'OTU2165:129leaf' 722. 'OTU2785:129leaf' 723. 'OTU735:129leaf' 724. 'OTU2058:129leaf'  
 725. 'OTU3746:70leaf' 726. 'OTU1386:14leaf' 727. 'OTU1853:129leaf' 728. 'OTU3209:87leaf'  
 729. 'OTU1210:129leaf' 730. 'OTU2008:14leaf' 731. 'OTU1240:83leaf' 732. 'OTU2214:14leaf'  
 733. 'OTU1912:85leaf' 734. 'OTU2601:15leaf' 735. 'OTU2095:69leaf' 736. 'OTU4012:70leaf'  
 737. 'OTU2591:14leaf' 738. 'OTU2215:14leaf' 739. 'OTU1774:129leaf' 740. 'OTU5246:83leaf'  
 741. 'OTU1906:69leaf' 742. 'OTU1307:14leaf' 743. 'OTU543:28leaf' 744. 'OTU2240:83leaf'  
 745. 'OTU2228:83leaf' 746. 'OTU1750:12leaf' 747. 'OTU1367:12leaf' 748. 'OTU2419:27leaf'  
 749. 'OTU2289:130leaf' 750. 'OTU2223:130leaf' 751. 'OTU865:130leaf' 752. 'OTU1621:58leaf'  
 753. 'OTU2040:17leaf' 754. 'OTU3244:130leaf' 755. 'OTU1964:130leaf' 756. 'OTU2253:130leaf'  
 757. 'OTU2735:32leaf' 758. 'OTU1760:130leaf' 759. 'OTU2513:13leaf' 760. 'OTU1720:3leaf'  
 761. 'OTU151:69leaf' 762. 'OTU171:22leaf' 763. 'OTU1301:131leaf' 764. 'OTU2255:131leaf'  
 765. 'OTU1209:131leaf' 766. 'OTU2485:131leaf' 767. 'OTU1590:131leaf' 768. 'OTU2492:61leaf'  
 769. 'OTU385:133leaf' 770. 'OTU1132:131leaf' 771. 'OTU2432:131leaf' 772. 'OTU950:13leaf'  
 773. 'OTU1837:131leaf' 774. 'OTU3195:131leaf' 775. 'OTU1728:131leaf' 776. 'OTU2201:131leaf'  
 777. 'OTU1796:131leaf' 778. 'OTU2852:131leaf' 779. 'OTU2707:131leaf' 780. 'OTU1056:131leaf'  
 781. 'OTU2252:94leaf' 782. 'OTU1714:131leaf' 783. 'OTU1388:65leaf' 784. 'OTU160:30leaf'  
 785. 'OTU5563:6leaf' 786. 'OTU430:132leaf' 787. 'OTU461:132leaf' 788. 'OTU1658:132leaf'  
 789. 'OTU887:132leaf' 790. 'OTU2459:132leaf' 791. 'OTU2597:132leaf' 792. 'OTU2801:132leaf'  
 793. 'OTU1236:132leaf' 794. 'OTU2826:132leaf' 795. 'OTU102:13leaf' 796. 'OTU765:55leaf'  
 797. 'OTU2385:133leaf' 798. 'OTU897:133leaf' 799. 'OTU2805:24leaf' 800. 'OTU2524:75leaf'

801. 'OTU2134:14leaf' 802. 'OTU180:27leaf' 803. 'OTU1530:13leaf' 804. 'OTU1273:26leaf'  
 805. 'OTU1295:13leaf' 806. 'OTU1428:13leaf' 807. 'OTU1752:13leaf' 808. 'OTU866:13leaf'  
 809. 'OTU2079:13leaf' 810. 'OTU2301:13leaf' 811. 'OTU8678:13leaf' 812. 'OTU1078:13leaf'  
 813. 'OTU1229:13leaf' 814. 'OTU982:21leaf' 815. 'OTU2343:69leaf' 816. 'OTU300:13leaf'  
 817. 'OTU860:13leaf' 818. 'OTU2619:26leaf' 819. 'OTU1944:13leaf' 820. 'OTU2411:13leaf'  
 821. 'OTU1261:13leaf' 822. 'OTU503:1leaf' 823. 'OTU1290:14leaf' 824. 'OTU2633:14leaf'  
 825. 'OTU3069:14leaf' 826. 'OTU6592:14leaf' 827. 'OTU1605:1leaf' 828. 'OTU1677:14leaf'  
 829. 'OTU1972:14leaf' 830. 'OTU2276:14leaf' 831. 'OTU304:15leaf' 832. 'OTU957:1leaf'  
 833. 'OTU2275:14leaf' 834. 'OTU322:30leaf' 835. 'OTU2677:83leaf' 836. 'OTU3132:14leaf'  
 837. 'OTU2892:80leaf' 838. 'OTU2648:14leaf' 839. 'OTU3104:14leaf' 840. 'OTU849:14leaf'  
 841. 'OTU917:14leaf' 842. 'OTU609:14leaf' 843. 'OTU2489:14leaf' 844. 'OTU7715:83leaf'  
 845. 'OTU205:15leaf' 846. 'OTU1048:17leaf' 847. 'OTU1959:14leaf' 848. 'OTU92:15leaf'  
 849. 'OTU1958:14leaf' 850. 'OTU647:81leaf' 851. 'OTU1086:94leaf' 852. 'OTU2319:14leaf'  
 853. 'OTU745:14leaf' 854. 'OTU172:15leaf' 855. 'OTU2405:15leaf' 856. 'OTU4503:83leaf'  
 857. 'OTU1482:14leaf' 858. 'OTU564:15leaf' 859. 'OTU2898:83leaf' 860. 'OTU203:15leaf'  
 861. 'OTU3367:14leaf' 862. 'OTU2576:14leaf' 863. 'OTU40:14leaf' 864. 'OTU2919:14leaf'  
 865. 'OTU1215:27leaf' 866. 'OTU571:15leaf' 867. 'OTU246:1leaf' 868. 'OTU530:14leaf'  
 869. 'OTU367:14leaf' 870. 'OTU120:14leaf' 871. 'OTU941:14leaf' 872. 'OTU2566:14leaf'  
 873. 'OTU671:15leaf' 874. 'OTU1329:14leaf' 875. 'OTU1681:14leaf' 876. 'OTU3281:14leaf'  
 877. 'OTU1193:94leaf' 878. 'OTU2568:15leaf' 879. 'OTU2634:15leaf' 880. 'OTU3019:15leaf'  
 881. 'OTU1223:16leaf' 882. 'OTU2722:15leaf' 883. 'OTU2291:32leaf' 884. 'OTU3079:15leaf'  
 885. 'OTU838:62leaf' 886. 'OTU1984:15leaf' 887. 'OTU2207:92leaf' 888. 'OTU37:15leaf'  
 889. 'OTU231:51leaf' 890. 'OTU1897:51leaf' 891. 'OTU1139:16leaf' 892. 'OTU2560:16leaf'  
 893. 'OTU1510:16leaf' 894. 'OTU1416:16leaf' 895. 'OTU962:16leaf' 896. 'OTU2999:16leaf'  
 897. 'OTU1557:16leaf' 898. 'OTU1843:16leaf' 899. 'OTU989:20leaf' 900. 'OTU2859:16leaf'  
 901. 'OTU324:4leaf' 902. 'OTU373:4leaf' 903. 'OTU1451:17leaf' 904. 'OTU1092:17leaf'  
 905. 'OTU1998:17leaf' 906. 'OTU1870:17leaf' 907. 'OTU1815:17leaf' 908. 'OTU2968:17leaf'  
 909. 'OTU2742:17leaf' 910. 'OTU2256:17leaf' 911. 'OTU2842:17leaf' 912. 'OTU3246:19leaf'  
 913. 'OTU2084:75leaf' 914. 'OTU2431:18leaf' 915. 'OTU1855:66leaf' 916. 'OTU2602:85leaf'  
 917. 'OTU1989:21leaf' 918. 'OTU2987:51leaf' 919. 'OTU1084:21leaf' 920. 'OTU1779:18leaf'  
 921. 'OTU8291:20leaf' 922. 'OTU2996:19leaf' 923. 'OTU1448:71leaf' 924. 'OTU1449:19leaf'  
 925. 'OTU1687:68leaf' 926. 'OTU1152:1leaf' 927. 'OTU2364:1leaf' 928. 'OTU44:1leaf'  
 929. 'OTU2273:85leaf' 930. 'OTU2420:1leaf' 931. 'OTU994:1leaf' 932. 'OTU423:35leaf'  
 933. 'OTU2024:85leaf' 934. 'OTU2926:1leaf' 935. 'OTU1465:85leaf' 936. 'OTU1879:1leaf'  
 937. 'OTU1638:1leaf' 938. 'OTU2184:35leaf' 939. 'OTU2071:1leaf' 940. 'OTU593:20leaf'  
 941. 'OTU198:30leaf' 942. 'OTU2401:21leaf' 943. 'OTU82:22leaf' 944. 'OTU2053:22leaf'  
 945. 'OTU1630:22leaf' 946. 'OTU2644:22leaf' 947. 'OTU1160:22leaf' 948. 'OTU676:30leaf'  
 949. 'OTU2378:23leaf' 950. 'OTU1231:23leaf' 951. 'OTU1430:23leaf' 952. 'OTU1487:23leaf'  
 953. 'OTU1814:23leaf' 954. 'OTU1893:23leaf' 955. 'OTU820:23leaf' 956. 'OTU2588:23leaf'  
 957. 'OTU817:23leaf' 958. 'OTU2268:23leaf' 959. 'OTU7180:4leaf' 960. 'OTU2589:24leaf'  
 961. 'OTU308:24leaf' 962. 'OTU2962:24leaf' 963. 'OTU895:24leaf' 964. 'OTU1622:24leaf'  
 965. 'OTU1947:24leaf' 966. 'OTU986:39leaf' 967. 'OTU137:26leaf' 968. 'OTU217:26leaf'  
 969. 'OTU659:26leaf' 970. 'OTU307:26leaf' 971. 'OTU1131:26leaf' 972. 'OTU1068:85leaf'  
 973. 'OTU2829:26leaf' 974. 'OTU1157:26leaf' 975. 'OTU2248:26leaf' 976. 'OTU1417:26leaf'  
 977. 'OTU1501:26leaf' 978. 'OTU2123:26leaf' 979. 'OTU2658:26leaf' 980. 'OTU2352:26leaf'  
 981. 'OTU2692:26leaf' 982. 'OTU1395:26leaf' 983. 'OTU2410:26leaf' 984. 'OTU2586:26leaf'  
 985. 'OTU90:48leaf' 986. 'OTU2771:26leaf' 987. 'OTU3004:62leaf' 988. 'OTU1158:62leaf'  
 989. 'OTU175:30leaf' 990. 'OTU443:27leaf' 991. 'OTU2073:27leaf' 992. 'OTU2341:27leaf'

993. 'OTU1534:27leaf' 994. 'OTU871:27leaf' 995. 'OTU880:4leaf' 996. 'OTU2236:27leaf'  
 997. 'OTU945:27leaf' 998. 'OTU1948:29leaf' 999. 'OTU1186:29leaf' 1000. 'OTU1051:29leaf'  
 1001. 'OTU633:29leaf' 1002. 'OTU2414:29leaf' 1003. 'OTU2445:29leaf' 1004. 'OTU2936:29leaf'  
 1005. 'OTU1565:29leaf' 1006. 'OTU1102:2leaf' 1007. 'OTU189:2leaf' 1008. 'OTU2059:2leaf'  
 1009. 'OTU919:65leaf' 1010. 'OTU222:2leaf' 1011. 'OTU1371:62leaf' 1012. 'OTU1669:2leaf'  
 1013. 'OTU1692:30leaf' 1014. 'OTU899:30leaf' 1015. 'OTU1547:30leaf' 1016. 'OTU1070:30leaf'  
 1017. 'OTU909:30leaf' 1018. 'OTU2526:30leaf' 1019. 'OTU7414:66leaf' 1020. 'OTU1699:30leaf'  
 1021. 'OTU2468:30leaf' 1022. 'OTU2025:30leaf' 1023. 'OTU463:30leaf' 1024. 'OTU2448:32leaf'  
 1025. 'OTU665:30leaf' 1026. 'OTU1918:32leaf' 1027. 'OTU1138:32leaf' 1028. 'OTU2927:31leaf'  
 1029. 'OTU393:32leaf' 1030. 'OTU2687:32leaf' 1031. 'OTU2041:32leaf' 1032. 'OTU563:32leaf'  
 1033. 'OTU1934:32leaf' 1034. 'OTU981:32leaf' 1035. 'OTU1842:32leaf' 1036. 'OTU2048:32leaf'  
 1037. 'OTU1265:32leaf' 1038. 'OTU2114:32leaf' 1039. 'OTU1933:32leaf' 1040. 'OTU1381:32leaf'  
 1041. 'OTU1865:32leaf' 1042. 'OTU1537:32leaf' 1043. 'OTU2023:32leaf' 1044. 'OTU2669:32leaf'  
 1045. 'OTU2816:32leaf' 1046. 'OTU3364:32leaf' 1047. 'OTU241:68leaf' 1048. 'OTU3672:33leaf'  
 1049. 'OTU1968:55leaf' 1050. 'OTU475:55leaf' 1051. 'OTU1081:75leaf' 1052. 'OTU323:39leaf'  
 1053. 'OTU2466:36leaf' 1054. 'OTU283:36leaf' 1055. 'OTU1293:37leaf' 1056. 'OTU2599:37leaf'  
 1057. 'OTU2719:37leaf' 1058. 'OTU2872:37leaf' 1059. 'OTU690:43leaf' 1060. 'OTU1275:38leaf'  
 1061. 'OTU1454:38leaf' 1062. 'OTU2283:38leaf' 1063. 'OTU937:58leaf' 1064. 'OTU1733:39leaf'  
 1065. 'OTU2172:39leaf' 1066. 'OTU2548:39leaf' 1067. 'OTU1742:39leaf' 1068. 'OTU2617:39leaf'  
 1069. 'OTU1018:3leaf' 1070. 'OTU1441:40leaf' 1071. 'OTU1337:40leaf' 1072. 'OTU2476:72leaf'  
 1073. 'OTU2715:40leaf' 1074. 'OTU2206:40leaf' 1075. 'OTU2101:40leaf' 1076. 'OTU1525:41leaf'  
 1077. 'OTU1414:41leaf' 1078. 'OTU3271:41leaf' 1079. 'OTU2203:41leaf' 1080. 'OTU105:41leaf'  
 1081. 'OTU7908:59leaf' 1082. 'OTU2116:42leaf' 1083. 'OTU687:43leaf' 1084. 'OTU1651:42leaf'  
 1085. 'OTU1063:43leaf' 1086. 'OTU1125:43leaf' 1087. 'OTU907:44leaf' 1088. 'OTU803:49leaf'  
 1089. 'OTU1767:45leaf' 1090. 'OTU1898:44leaf' 1091. 'OTU691:44leaf' 1092. 'OTU1054:44leaf'  
 1093. 'OTU2329:44leaf' 1094. 'OTU576:44leaf' 1095. 'OTU2133:45leaf' 1096. 'OTU1834:45leaf'  
 1097. 'OTU2294:45leaf' 1098. 'OTU661:47leaf' 1099. 'OTU1663:52leaf' 1100. 'OTU1845:57leaf'  
 1101. 'OTU3121:51leaf' 1102. 'OTU1503:47leaf' 1103. 'OTU654:47leaf' 1104. 'OTU2674:71leaf'  
 1105. 'OTU1703:52leaf' 1106. 'OTU786:4leaf' 1107. 'OTU3230:4leaf' 1108. 'OTU869:4leaf'  
 1109. 'OTU900:4leaf' 1110. 'OTU2112:4leaf' 1111. 'OTU1542:4leaf' 1112. 'OTU1233:4leaf'  
 1113. 'OTU1883:4leaf' 1114. 'OTU700:4leaf' 1115. 'OTU3554:4leaf' 1116. 'OTU1418:4leaf'  
 1117. 'OTU1849:4leaf' 1118. 'OTU1634:4leaf' 1119. 'OTU2371:4leaf' 1120. 'OTU2456:4leaf'  
 1121. 'OTU886:88leaf' 1122. 'OTU975:88leaf' 1123. 'OTU2120:88leaf' 1124. 'OTU1644:4leaf'  
 1125. 'OTU842:88leaf' 1126. 'OTU1739:51leaf' 1127. 'OTU1402:51leaf' 1128. 'OTU3012:51leaf'  
 1129. 'OTU996:50leaf' 1130. 'OTU2665:51leaf' 1131. 'OTU742:51leaf' 1132. 'OTU1317:50leaf'  
 1133. 'OTU3037:50leaf' 1134. 'OTU1584:50leaf' 1135. 'OTU2325:51leaf' 1136. 'OTU830:51leaf'  
 1137. 'OTU2212:51leaf' 1138. 'OTU1819:51leaf' 1139. 'OTU2838:51leaf' 1140. 'OTU11279:51leaf'  
 1141. 'OTU1682:51leaf' 1142. 'OTU2798:51leaf' 1143. 'OTU1363:51leaf' 1144. 'OTU1673:51leaf'  
 1145. 'OTU3206:51leaf' 1146. 'OTU1807:51leaf' 1147. 'OTU1756:51leaf' 1148. 'OTU2803:51leaf'  
 1149. 'OTU1507:51leaf' 1150. 'OTU2491:51leaf' 1151. 'OTU2681:51leaf' 1152. 'OTU2056:51leaf'  
 1153. 'OTU1732:94leaf' 1154. 'OTU1749:52leaf' 1155. 'OTU1915:52leaf' 1156. 'OTU2251:65leaf'  
 1157. 'OTU3252:54leaf' 1158. 'OTU859:55leaf' 1159. 'OTU1583:55leaf' 1160. 'OTU2953:55leaf'  
 1161. 'OTU2208:55leaf' 1162. 'OTU1872:56leaf' 1163. 'OTU2650:58leaf' 1164. 'OTU1839:59leaf'  
 1165. 'OTU2965:59leaf' 1166. 'OTU711:81leaf' 1167. 'OTU1686:60leaf' 1168. 'OTU513:60leaf'  
 1169. 'OTU2881:60leaf' 1170. 'OTU1798:61leaf' 1171. 'OTU6594:61leaf' 1172. 'OTU648:71leaf'  
 1173. 'OTU1607:61leaf' 1174. 'OTU968:61leaf' 1175. 'OTU195:61leaf' 1176. 'OTU2344:62leaf'  
 1177. 'OTU1794:62leaf' 1178. 'OTU9756:62leaf' 1179. 'OTU1074:84leaf' 1180. 'OTU253:63leaf'  
 1181. 'OTU1886:64leaf' 1182. 'OTU2361:64leaf' 1183. 'OTU1708:64leaf' 1184. 'OTU1373:64leaf'

1185. 'OTU2501:64leaf' 1186. 'OTU2533:64leaf' 1187. 'OTU2818:64leaf' 1188. 'OTU1463:66leaf'  
1189. 'OTU325:66leaf' 1190. 'OTU916:66leaf' 1191. 'OTU1851:66leaf' 1192. 'OTU1100:66leaf'  
1193. 'OTU1913:66leaf' 1194. 'OTU1940:66leaf' 1195. 'OTU1551:66leaf' 1196. 'OTU780:66leaf'  
1197. 'OTU2049:66leaf' 1198. 'OTU1473:66leaf' 1199. 'OTU2605:68leaf' 1200. 'OTU1494:6leaf'  
1201. 'OTU3046:81leaf' 1202. 'OTU1531:69leaf' 1203. 'OTU1033:69leaf' 1204. 'OTU974:69leaf'  
1205. 'OTU1643:69leaf' 1206. 'OTU1560:69leaf' 1207. 'OTU420:69leaf' 1208. 'OTU2577:69leaf'  
1209. 'OTU2171:69leaf' 1210. 'OTU6205:70leaf' 1211. 'OTU980:69leaf' 1212. 'OTU1378:71leaf'  
1213. 'OTU1353:69leaf' 1214. 'OTU1028:69leaf' 1215. 'OTU1177:69leaf' 1216. 'OTU2498:70leaf'  
1217. 'OTU2013:6leaf' 1218. 'OTU7913:6leaf' 1219. 'OTU2181:6leaf' 1220. 'OTU1656:90leaf'  
1221. 'OTU2482:6leaf' 1222. 'OTU2198:90leaf' 1223. 'OTU1836:6leaf' 1224. 'OTU2190:90leaf'  
1225. 'OTU2150:6leaf' 1226. 'OTU2357:90leaf' 1227. 'OTU1497:90leaf' 1228. 'OTU1423:6leaf'  
1229. 'OTU2877:6leaf' 1230. 'OTU2515:6leaf' 1231. 'OTU2078:90leaf' 1232. 'OTU3219:70leaf'  
1233. 'OTU784:71leaf' 1234. 'OTU568:71leaf' 1235. 'OTU1577:71leaf' 1236. 'OTU775:71leaf'  
1237. 'OTU2022:71leaf' 1238. 'OTU1899:71leaf' 1239. 'OTU1735:71leaf' 1240. 'OTU2427:71leaf'  
1241. 'OTU770:72leaf' 1242. 'OTU3180:72leaf' 1243. 'OTU2525:85leaf' 1244. 'OTU342:73leaf'  
1245. 'OTU1002:73leaf' 1246. 'OTU1285:73leaf' 1247. 'OTU854:73leaf' 1248. 'OTU1264:75leaf'  
1249. 'OTU1024:75leaf' 1250. 'OTU1446:75leaf' 1251. 'OTU3117:75leaf' 1252. 'OTU1617:85leaf'  
1253. 'OTU2424:90leaf' 1254. 'OTU2929:90leaf' 1255. 'OTU2766:90leaf' 1256. 'OTU2460:90leaf'  
1257. 'OTU2706:90leaf' 1258. 'OTU2121:79leaf' 1259. 'OTU2830:90leaf' 1260. 'OTU1982:81leaf'  
1261. 'OTU2122:81leaf' 1262. 'OTU2347:81leaf' 1263. 'OTU327:85leaf' 1264. 'OTU1589:93leaf'  
1265. 'OTU1094:85leaf' 1266. 'OTU867:85leaf' 1267. 'OTU1179:85leaf' 1268. 'OTU188:85leaf'  
1269. 'OTU1375:82leaf' 1270. 'OTU2603:85leaf' 1271. 'OTU1804:96leaf' 1272. 'OTU1403:96leaf'  
1273. 'OTU1561:96leaf' 1274. 'OTU2292:85leaf' 1275. 'OTU2545:85leaf' 1276. 'OTU1685:85leaf'  
1277. 'OTU2861:93leaf' 1278. 'OTU2880:88leaf' 1279. 'OTU1926:88leaf' 1280. 'OTU3277:89leaf'  
1281. 'OTU2400:90leaf' 1282. 'OTU2094:90leaf' 1283. 'OTU3140:92leaf' 1284. 'OTU3177:93leaf'  
1285. 'OTU1461:94leaf' 1286. 'OTU3005:94leaf' 1287. 'OTU2930:94leaf' 1288. 'OTU3193:94leaf'  
1289. 'OTU3137:94leaf' 1290. 'OTU2328:94leaf' 1291. 'OTU2795:95leaf' 1292. 'OTU2624:95leaf'  
1293. 'OTU3013:95leaf' 1294. 'OTU1582:96leaf' 1295. 'OTU1956:96leaf' 1296. 'OTU2388:96leaf'  
1297. 'OTU2514:96leaf' 1298. 'OTU2630:98leaf' 1299. 'OTU2703:98leaf' 1300. 'OTU977:98leaf'  
1301. 'OTU2250:98leaf' 1302. 'OTU1660:99leaf'

In [101]: `any(woodsp %in% leafspp)`

FALSE

In [102]: `any(leafspp %in% woodsp)`

FALSE

Huh, no shared species... might be in for a ride now... let's see what other differences present.

Host NMS/PermANOVA

Let's look for grouping of samples by host. It is difficult here to disentangle spatial patterns of the host and microbe here, if we find "host preference" this may be due to fungal species simply preferring the same habitat as the host, and not host-microbe interactions, etc. But a first look is needed.

Wood library, Host effects

In [104]: `woodbiom <- subset_samples(deseq95, Library=='W')`  
`woodOTU <- otu_table(woodbiom)`  
`wood <- t(woodOTU@.Data) ## transpose, default phyloseq uses otus as rows`

```

wood_data <- sample_data(woodbiom)
WoodMDS <- metaMDS(wood)
#save(WoodMDS, file = "WoodHostMDS.rda")

Wisconsin double standardization
Run 0 stress 0.2762519
Run 1 stress 0.2861963
Run 2 stress 0.2798003
Run 3 stress 0.2768649
Run 4 stress 0.2800364
Run 5 stress 0.2760376
... New best solution
... Procrustes: rmse 0.05330076 max resid 0.3131359
Run 6 stress 0.2745264
... New best solution
... Procrustes: rmse 0.09014175 max resid 0.4031058
Run 7 stress 0.2769832
Run 8 stress 0.2737327
... New best solution
... Procrustes: rmse 0.04985925 max resid 0.2799055
Run 9 stress 0.2792227
Run 10 stress 0.2783602
Run 11 stress 0.2774815
Run 12 stress 0.2722429
... New best solution
... Procrustes: rmse 0.04791228 max resid 0.2363086
Run 13 stress 0.2743483
Run 14 stress 0.279028
Run 15 stress 0.2753582
Run 16 stress 0.2803394
Run 17 stress 0.2746752
Run 18 stress 0.2766783
Run 19 stress 0.2758248
Run 20 stress 0.279634
*** No convergence -- monoMDS stopping criteria:
  1: no. of iterations >= maxit
  19: stress ratio > sratmax

```

```

In [108]: samps <- table(wood_data$Host_genus_species)
spp <- names(samps)
hostspp <- sapply(wood_data$Host_genus_species, FUN= function(x){which(spp == x)})
n <- length(spp)
palette <- distinctColorPalette(n)

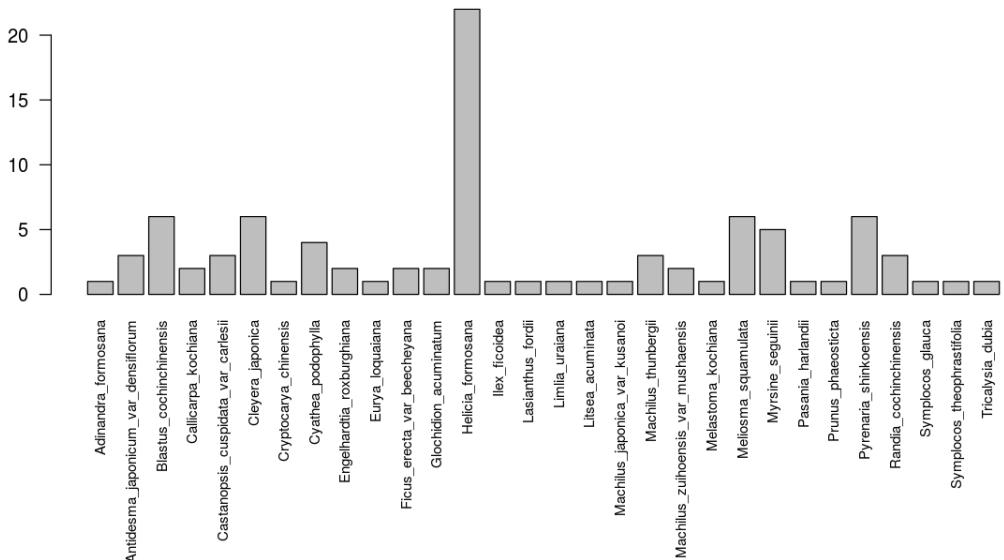
```

Before we plot, just checking, what do our samples sizes look like for each tree host?

```

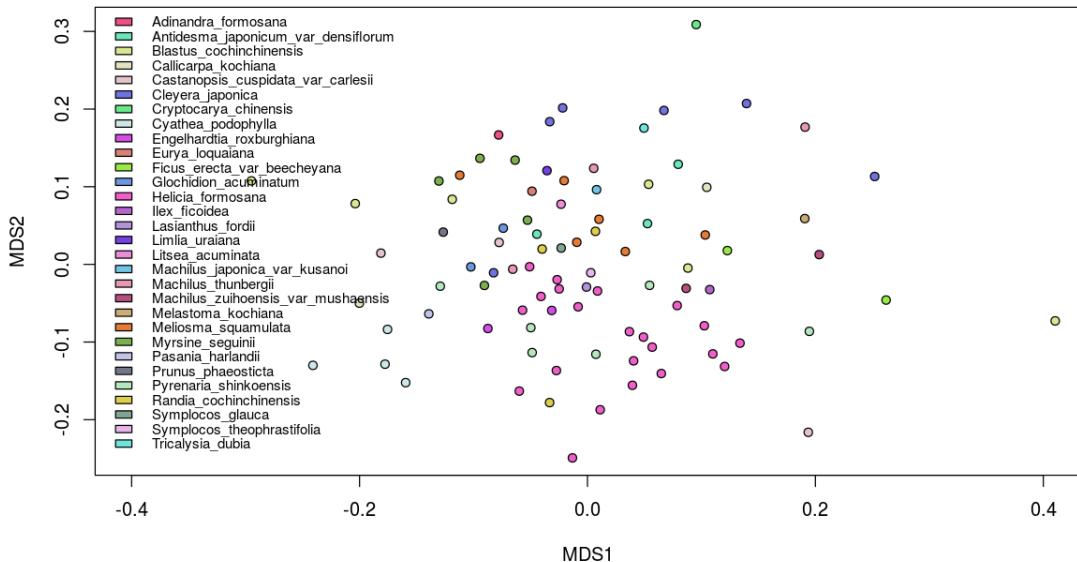
In [109]: options(repr.plot.width = 10,repr.plot.height = 6)
par(oma=c(8,0,0,0))
barplot(samps, las=2, cex.names=.75)

```



Definitely some pretty uneven sample sizes.

```
In [110]: plot(WoodMDS$points,
            bg=palette[hostsp], 
            col='black',
            pch=21,
            xlim=c(-0.4,0.4)
            )
legend('topleft',
       legend=spp,
       fill=palette,
       cex=.75,
       bty='n')
```



Suggestive, looks like some host communities are grouping together. We have a lot of low-sample hosts, we can remove host trees with less than 3 samples from the graphic to clear things up a bit.

```
In [111]: hisamps <- samps[samps > 2] ## only hosts with >2 samples
hihosts <- wood_data$Host_genus_species %in% names(hisamps)
## make new groups/color palette
hispp <- names(hisamps)
hihostsp <- sapply(wood_data$Host_genus_species[hihosts], FUN= function(x){which(hisp
n <- length(hispp)
palette <- distinctColorPalette(n)

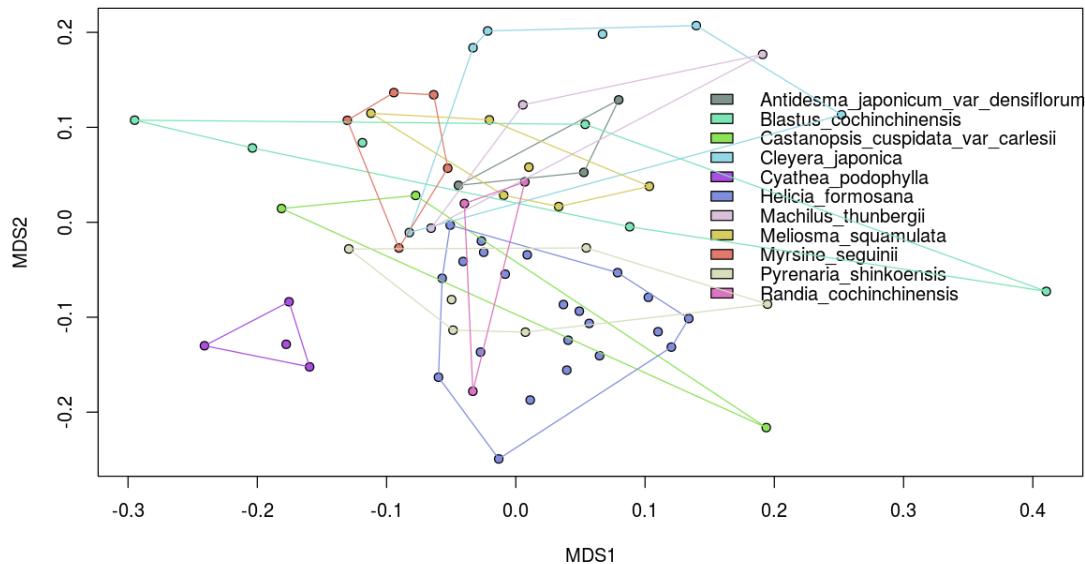
In [113]: #png('woodhostNMS.png')
plot(WoodMDS$points[hihosts,],
      col="black",
      pch = 21,
      bg = palette[hihostsp],
      #xlim = c(-0.15, .35),
      )
legend( x=.13,
        y=0.15,
        legend=hispp,
        fill=palette,
        bty='n')

## hulls have to be drawn one by one if we use ordihull():
for(i in 1:length(hispp)){
```

```

ordihull(WoodMDS, groups=wood_data$Host_genus_species, show.groups=hispp[i], col=palette)
}
#dev.off()

```



In [114]: testwoodhost <- adonis(wood ~ wood\_data\$Host\_genus\_species, permutations=10000)

In [115]: testwoodhost

Call:

adonis(formula = wood ~ wood\_data\$Host\_genus\_species, permutations = 10000)

Permutation: free

Number of permutations: 10000

Terms added sequentially (first to last)

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
wood_data\$Host_genus_species	29	14.849	0.51205	1.4945	0.41538	9.999e-05 ***
Residuals	61	20.900	0.34262		0.58462	
Total	90	35.749			1.00000	
---						
Signif. codes:	0	***	0.001	** 0.01	* 0.05	. 0.1
						1

$R^2 = .415$ , significant. But again, hard to trust these numbers given that we are ignoring spatial aspects of this, other covariates, and there are dispersion differences. The NMS figures show a centroid location difference among several groups, though.

Check the dispersion.

```
In [117]: dis <- vegdist(wood)
          disp <- betadisper(dis, hostspp)
          disp
          anova(disp) ## still significantly different
```

Homogeneity of multivariate dispersions

Call: betadisper(d = dis, group = hostspp)

No. of Positive Eigenvalues: 90

No. of Negative Eigenvalues: 0

Average distance to median:

1	2	3	4	5	6	7	8	9	10	11
0.0000	0.4780	0.6045	0.3732	0.5271	0.5659	0.0000	0.5197	0.3966	0.0000	0.3808
12	13	14	15	16	17	18	19	20	21	22
0.3524	0.5647	0.0000	0.0000	0.0000	0.0000	0.0000	0.5006	0.4173	0.0000	0.5045
23	24	25	26	27	28	29	30			
0.4595	0.0000	0.0000	0.5112	0.4714	0.0000	0.0000	0.0000			

Eigenvalues for PCoA axes:

PCoA1	PCoA2	PCoA3	PCoA4	PCoA5	PCoA6	PCoA7	PCoA8
2.1934	1.9167	1.6608	1.4984	1.1265	0.9894	0.9303	0.8360

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Groups	29	3.4684323	0.119601113	42.94714	1.081307e-30
Residuals	61	0.1698755	0.002784845	NA	NA

```
In [118]: permute(disp, pairwise = TRUE)
```

Permutation test for homogeneity of multivariate dispersions

Permutation: free

Number of permutations: 999

Response: Distances

Df	Sum Sq	Mean Sq	F	N.Perm	Pr(>F)
Groups	29	3.4684	0.119601	42.947	999 0.001 ***
Residuals	61	0.1699	0.002785		
---					
Signif. codes:	0	***	0.001	** 0.01 *	0.05 . 0.1 1

Pairwise comparisons:

(Observed p-value below diagonal, permuted p-value above diagonal)

	1	2	3	4	5	6	7	8			
1											
2		1.1000e-02	3.4000e-02	5.3400e-01	1.8000e-02		2.6100e-01				
3	8.1856e-03		7.0000e-03	1.6000e-01	2.0700e-01		2.1000e-02				
4	2.6798e-02	1.1792e-03		1.1400e-01	1.2000e-02		1.5000e-02				
5	4.9550e-01	1.8071e-01	1.5267e-01		4.9800e-01		9.2200e-01				
6	1.6664e-02	1.9517e-01	8.0972e-04	4.4475e-01			1.3700e-01				
7											
8	2.4381e-01	3.2321e-02	1.2622e-02	9.0480e-01	1.3549e-01						
9	5.0923e-02	2.0364e-03	5.0584e-29	2.0443e-01	1.5888e-03		2.2460e-02				
10											
11	3.2613e-02	1.4020e-03	1.3296e-28	1.6763e-01	1.0010e-03		1.5113e-02				
12	1.6493e-02	7.5029e-04	1.7819e-29	1.1914e-01	4.6810e-04		7.9703e-03				
13	1.0102e-03	4.6056e-02	4.6146e-07	2.1788e-01	9.4349e-01		4.3281e-02				
14											
15											
16											
17											
18											
19	5.2349e-01	2.4131e-02	3.0908e-02	7.1507e-01	6.6219e-02		6.0250e-01				
20	9.9709e-02	3.4194e-03	3.9663e-30	2.6685e-01	3.0493e-03		3.9616e-02				
21											
22	3.1804e-01	3.4176e-03	2.4198e-03	6.3864e-01	2.0211e-02		5.6621e-01				
23	7.4029e-01	7.5565e-03	2.3753e-01	3.6233e-01	2.4480e-02		2.4889e-01				
24											
25											
26	4.4577e-01	2.2373e-02	2.9871e-02	7.8709e-01	1.1477e-01		8.2842e-01				
27	9.2616e-01	3.8362e-02	3.1359e-01	5.6446e-01	9.0644e-02		4.5145e-01				
28											
29											
30											
	9	10	11	12	13	14	15	16	17	18	19
1											
2	3.6000e-02		3.8000e-02	2.3000e-02	6.0000e-03						6.1600e-01
3	9.0000e-03		1.5000e-02	8.0000e-03	3.7000e-02						1.8000e-02
4	1.0000e-03		1.0000e-03	1.0000e-03	2.0000e-03						3.6000e-02
5	1.6700e-01		1.2500e-01	6.9000e-02	1.8300e-01						7.4200e-01
6	1.2000e-02		1.3000e-02	4.0000e-03	9.5600e-01						5.0000e-02
7											
8	2.4000e-02		2.6000e-02	1.2000e-02	3.9000e-02						6.7100e-01
9				1.0000e-03	1.0000e-03						3.6000e-02
10											
11	8.0200e-29			1.0000e-03	1.0000e-03						3.7000e-02
12	1.0249e-29		0.0000e+00		1.0000e-03						2.8000e-02
13	3.2236e-06		8.5922e-07	8.9038e-08							1.5000e-02
14											

```

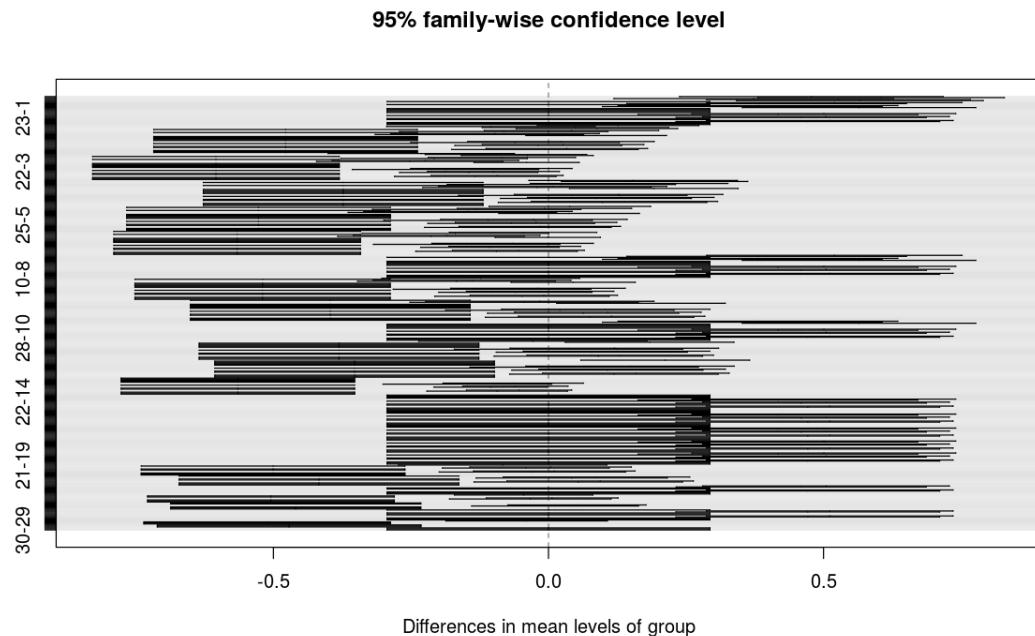
15
16
17
18
19 5.1510e-02    3.6210e-02 2.0738e-02 1.2361e-02
20 4.6943e-29    0.0000e+00 0.0000e+00 1.9236e-05          8.6211e-02
21
22 6.2183e-03    3.2483e-03 1.1442e-03 1.6661e-03          8.8840e-01
23 3.7336e-01    2.7585e-01 1.5688e-01 1.9728e-04          4.7899e-01
24
25
26 5.6917e-02    3.6729e-02 1.7275e-02 1.4710e-02          8.1018e-01
27 4.2539e-01    3.4635e-01 2.3949e-01 4.6310e-03          6.8982e-01
28
29
30
      20 21       22       23 24 25       26       27 28 29 30
1
2 8.2000e-02    3.3000e-01 7.7600e-01        4.9000e-01 9.3500e-01
3 1.6000e-02    1.0000e-03 4.0000e-03        2.0000e-02 3.6000e-02
4               1.1000e-02 2.1300e-01        3.7000e-02 3.2600e-01
5 2.3100e-01    7.0400e-01 4.0200e-01        8.3000e-01 6.4000e-01
6 2.2000e-02    8.0000e-03 2.3000e-02        1.0900e-01 7.6000e-02
7
8 3.4000e-02    5.9400e-01 2.8100e-01        8.6100e-01 5.3300e-01
9 1.0000e-03    1.1000e-02 3.8200e-01        5.7000e-02 4.7000e-01
10
11               1.2000e-02 2.6700e-01        4.0000e-02
12 1.0000e-03    7.0000e-03 1.4500e-01        1.7000e-02 2.1500e-01
13 1.0000e-03    4.0000e-03 2.0000e-03        1.7000e-02 1.1000e-02
14
15
16
17
18
19 6.3000e-02    9.1700e-01 5.3000e-01        8.3600e-01 7.6800e-01
20               2.0000e-02 5.9700e-01        8.4000e-02 6.2100e-01
21
22 1.5782e-02            3.0200e-01        8.5700e-01 5.4500e-01
23 5.4081e-01    2.6754e-01            3.1800e-01 8.7700e-01
24
25
26 1.0213e-01    8.2959e-01 2.8496e-01          5.5100e-01
27 5.5300e-01    4.9755e-01 8.6762e-01          5.0617e-01
28
29
30

```

Do a type of tukeys to show why there are significant differences in dispersion among our host/microbe groupings.

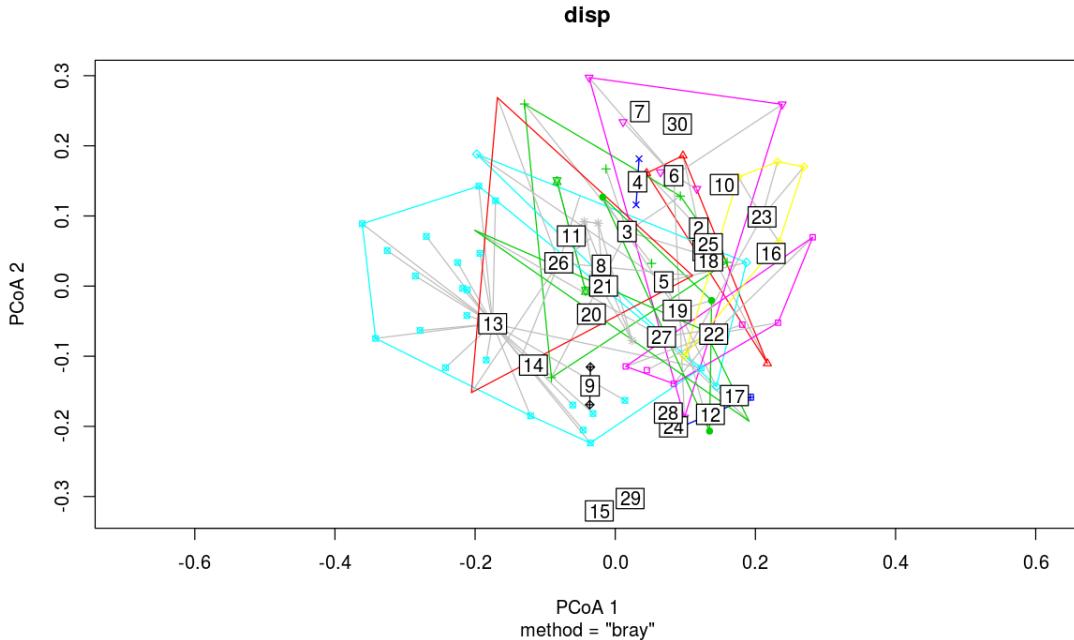
```
In [119]: disp.HSD <- TukeyHSD(disp)
```

```
In [120]: plot(disp.HSD)
```



```
In [121]: plot(disp)
```

```
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '26'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '26'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '26'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '27'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '30'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '27'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '29'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '27'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '26'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '28'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '26'  
Warning message in plot.xy(xy.coords(x, y), type = type, ...):  
unimplemented pch value '26'
```



These are distances along principal coordinates of the BC dissimilarity matrix of the community matrix. The beta-dispersions checks online propose this as an alternative to NMS. The first graph above this is a classic tukeys comparison among all the host-groups. I.e., if comparisons that wander far away from the zero are very different in their dispersions and having comparisons all over the chart like this reinforces that we have lots of different sized dispersions.

I don't really think there is much here that we can't see in the NMS - dispersions vary among the samples when grouped by host, centroids and sometimes shells among some groups are clearly separate. Just seems to reinforce the previous results - if the communities are separate in NMS solutions with reasonably low stresses, we can trust the PERMANOVA results as an indicator of dissimilarity due to host (or whatever factor). So I don't feel the need to repeat this with the leaves, as long as the NMS solutions are reasonable.

But I don't really trust the effect sizes here. This test is to

Leaf library, Host effects

```
In [208]: load('deseq95.rda')
leafbiom <- subset_samples(deseq95, Library=='L')
leafOTU <- otu_table(leafbiom)
leaf <- t(leafOTU@.Data)
leaf_data <- sample_data(leafbiom)
```

```
In [130]: leafMDS <- metaMDS(leaf)
```

```
Wisconsin double standardization
Run 0 stress 0.1260886
Run 1 stress 0.126286
... Procrustes: rmse 0.007593137 max resid 0.02835048
Run 2 stress 0.1261514
```

```

... Procrustes: rmse 0.005385356 max resid 0.02302638
Run 3 stress 0.1262231
... Procrustes: rmse 0.006093234 max resid 0.02625702
Run 4 stress 0.1261953
... Procrustes: rmse 0.00556488 max resid 0.02256784
Run 5 stress 0.1261427
... Procrustes: rmse 0.008918071 max resid 0.0310009
Run 6 stress 0.1261769
... Procrustes: rmse 0.009147585 max resid 0.03085108
Run 7 stress 0.1263541
... Procrustes: rmse 0.009491294 max resid 0.03489594
Run 8 stress 0.1262606
... Procrustes: rmse 0.007201247 max resid 0.01976357
Run 9 stress 0.1260835
... New best solution
... Procrustes: rmse 0.006365073 max resid 0.02962531
Run 10 stress 0.1263009
... Procrustes: rmse 0.006884862 max resid 0.03019338
Run 11 stress 0.126175
... Procrustes: rmse 0.004563682 max resid 0.03072516
Run 12 stress 0.1262071
... Procrustes: rmse 0.008727001 max resid 0.02625479
Run 13 stress 0.1263572
... Procrustes: rmse 0.008898466 max resid 0.01921627
Run 14 stress 0.1263834
... Procrustes: rmse 0.0087503 max resid 0.02006456
Run 15 stress 0.12625
... Procrustes: rmse 0.005737872 max resid 0.02035297
Run 16 stress 0.1262919
... Procrustes: rmse 0.008109421 max resid 0.02859299
Run 17 stress 0.1260761
... New best solution
... Procrustes: rmse 0.006008886 max resid 0.02786557
Run 18 stress 0.1261404
... Procrustes: rmse 0.006638012 max resid 0.03038438
Run 19 stress 0.1260613
... New best solution
... Procrustes: rmse 0.006506066 max resid 0.02293031
Run 20 stress 0.1263235
... Procrustes: rmse 0.00590993 max resid 0.02475982
*** No convergence -- monoMDS stopping criteria:
 20: stress ratio > sratmax

```

In [132]: `#save(leafMDS, file='leafMDS.rda')`

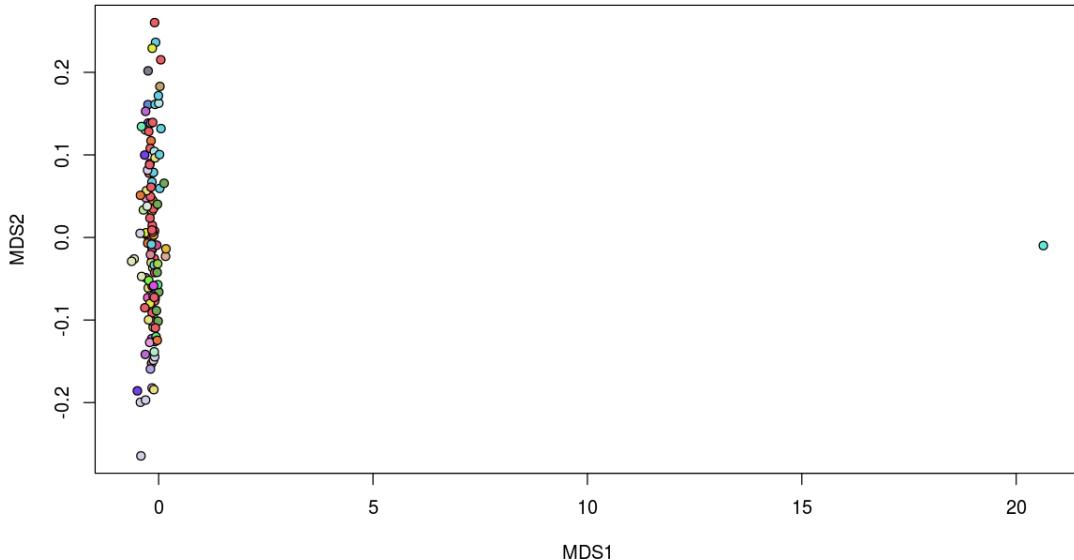
In [131]: `## host groups/colors`  
`samps <- table(leaf_data$Host_genus_species)`

```

spp <- names(samps)
hostspp <- sapply(leaf_data$Host_genus_species, FUN= function(x){which(spp == x)})
#spp == sort(unique(leaf_data$Host_genus_species)) ## yup, table function does this
n <- length(spp)
palette <- distinctColorPalette(n)

plot(leafMDS$points,
      col="black",
      pch = 21,
      bg = palette[hostspp]
)

```



Deal with outlier.

```
In [135]: aa <- identify(leafMDS$points)
leafMDS$points[aa, , drop=FALSE]
```

	MDS1	MDS2
67leaf	20.63293	-0.009804052

```
In [166]: load('deseq95.rda')
#leafbiom <- subset_samples(deseq95, Library=='L')
leafbiom <- subset_samples(deseq95, Library=='L' & sample_names(deseq95) != '67leaf')
leafOTU <- otu_table(leafbiom)
leaf <- t(leafOTU@.Data)
leaf_data <- sample_data(leafbiom)
leafMDS <- metaMDS(leaf)
```

```

Wisconsin double standardization
Run 0 stress 0.2272767
Run 1 stress 0.2228183
... New best solution
... Procrustes: rmse 0.06748874 max resid 0.2225008
Run 2 stress 0.2234125
Run 3 stress 0.221964
... New best solution
... Procrustes: rmse 0.03399047 max resid 0.1716347
Run 4 stress 0.2261661
Run 5 stress 0.2242256
Run 6 stress 0.2207596
... New best solution
... Procrustes: rmse 0.05095369 max resid 0.2120125
Run 7 stress 0.2233848
Run 8 stress 0.224871
Run 9 stress 0.2242449
Run 10 stress 0.2209459
... Procrustes: rmse 0.01820455 max resid 0.1361952
Run 11 stress 0.2256308
Run 12 stress 0.220368
... New best solution
... Procrustes: rmse 0.04457831 max resid 0.1884812
Run 13 stress 0.2278026
Run 14 stress 0.2212119
Run 15 stress 0.2212353
Run 16 stress 0.2264097
Run 17 stress 0.2227323
Run 18 stress 0.2222018
Run 19 stress 0.2208622
... Procrustes: rmse 0.05525574 max resid 0.2311904
Run 20 stress 0.2240812
*** No convergence -- monoMDS stopping criteria:
  3: no. of iterations >= maxit
  17: stress ratio > sratmax

```

Save this solution for reproducibility:

```
In [191]: #save(leafMDS, file='leafNMS_260618.rda')
```

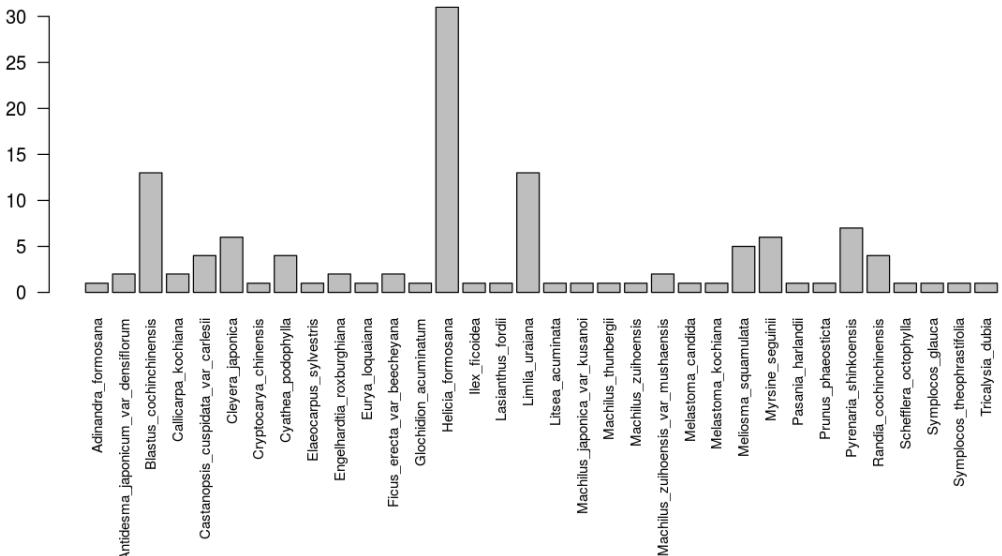
Check sample sizes and replot NMS

```
In [167]: ## host groups/colors
samps <- table(leaf_data$Host_genus_species)
```

```
In [168]: samps
```

Adinandra_formosana	Antidesma_japonicum_var_densiflorum	
1		2
Blastus_cochinchinensis	Callicarpa_kochiana	
13		2
Castanopsis_cuspidata_var_carlesii	Cleyera_japonica	
4		6
Cryptocarya_chinensis	Cyathea_podophylla	
1		4
Elaeocarpus_sylvestris	Engelhardtia_roxburghiana	
1		2
Eurya_loquaiana	Ficus_erecta_var_beecheyana	
1		2
Glochidion_acuminatum	Helicia_formosana	
1		31
Ilex_ficoidea	Lasianthus_fordii	
1		1
Limlia_uraiana	Litsea_acuminata	
13		1
Machilus_japonica_var_kusanoi	Machilus_thunbergii	
1		1
Machilus_zuihoensis	Machilus_zuihoensis_var_mushaensis	
1		2
Melastoma_candida	Melastoma_kochiana	
1		1
Meliosma_squamulata	Myrsine_seguinii	
5		6
Pasania_harlandii	Prunus_phaeosticta	
1		1
Pyrenaria_shinkoensis	Randia_cochinchinensis	
7		4
Schefflera_octophylla	Symplocos_glauca	
1		1
Symplocos_theophrastifolia	Tricalysia_dubia	
1		1

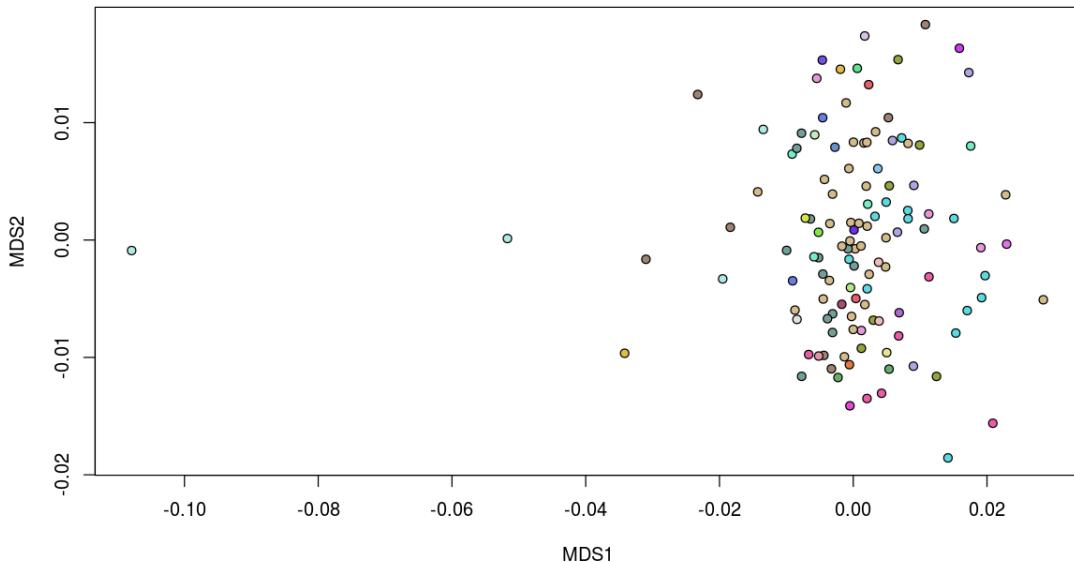
```
In [169]: options(repr.plot.width = 10,repr.plot.height = 6)
par(oma=c(8,0,0,0))
barplot(samps, las=2, cex.names=.75)
```



As with the wood, *Helicia* is by far the most common host, and there are numerous small sample sizes for some of the other hosts. Rerun the NMS without the outlier:

```
In [170]: spp <- names(samps)
hostspp <- sapply(leaf_data$Host_genus_species, FUN= function(x){which(spp == x)})
#spp == sort(unique(leaf_data$Host_genus_species)) ## yup, table function does this
n <- length(spp)
palette <- distinctColorPalette(n)

plot(leafMDS$points,
      col="black",
      pch = 21,
      bg = palette[hostspp]
)
```



```
In [187]: hisamps <- samps[samps > 2] ## only hosts with >2 samples
hihosts <- leaf_data$Host_genus_species %in% names(hisamps)
## make new groups/color palette
hispp <- names(hisamps)
hihostspp <- sapply(leaf_data$Host_genus_species[hihosts], FUN= function(x){which(hisp
n <- length(hispp)
palette <- distinctColorPalette(n)

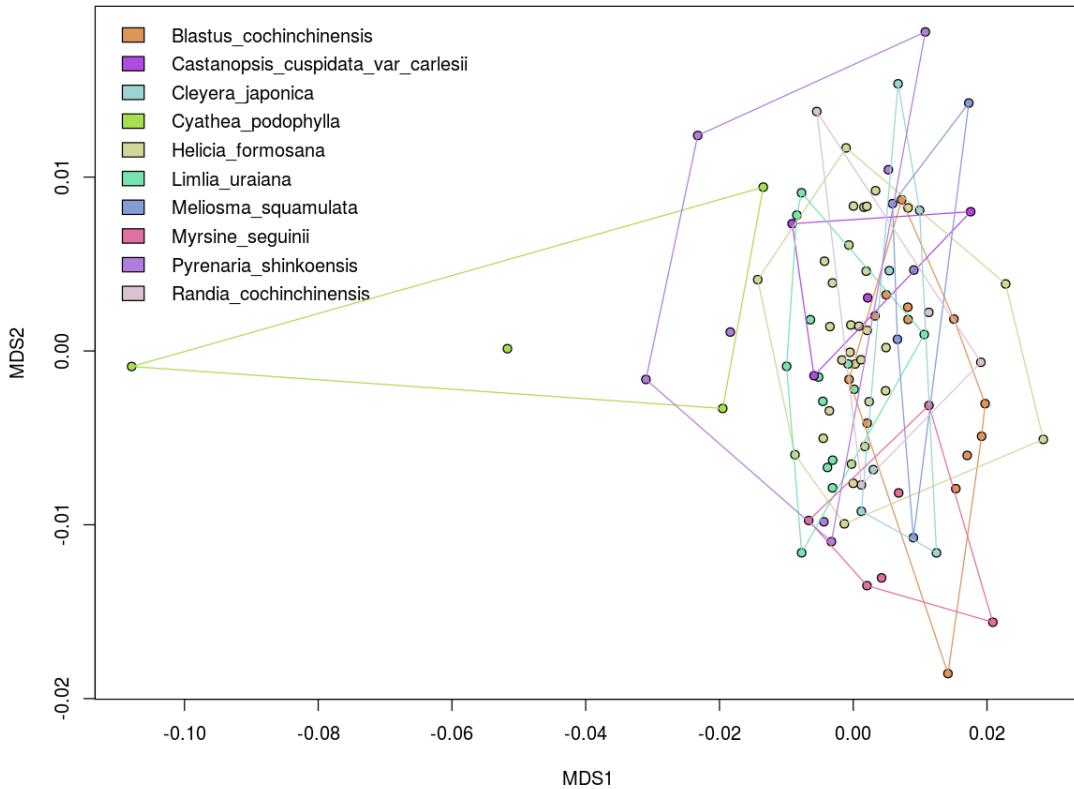
In [188]: options(repr.plot.width = 10,repr.plot.height = 8)
#png('leafhostNMS.png')
plot(leafMDS$points[hihosts,],
      col="black",
      pch = 21,
      bg = palette[hihostspp]
    )
length(hihosts); dim(leafMDS$points)

legend("topleft",
      legend=hispp,
      fill=palette,
      bty='n',
      )

for(i in 1:length(hispp)){
ordihull(leafMDS, groups=leaf_data$Host_genus_species, show.groups=hispp[i], col=palet
```

```
    }
    #dev.off()
```

122  
1. 122 2. 2



Zoom in a little.

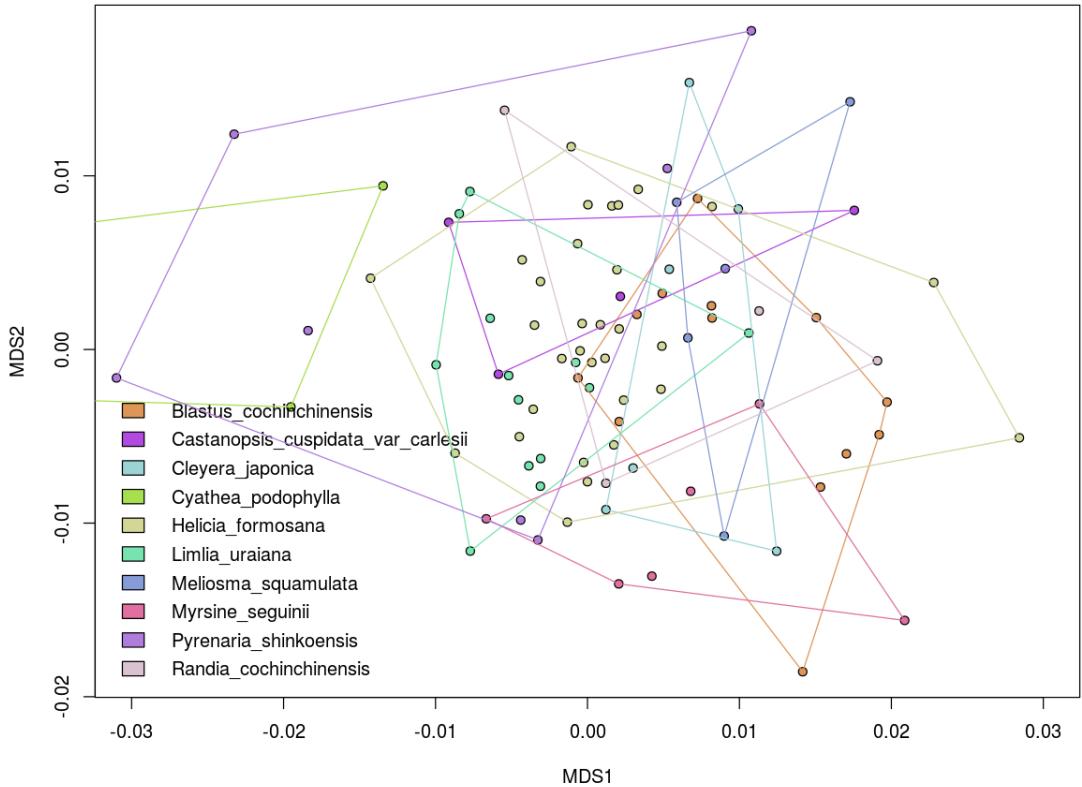
```
In [192]: options(repr.plot.width = 10,repr.plot.height = 8)
#png('leafhostNMS.png')
plot(leafMDS$points[hihosts,],
      col="black",
      pch = 21,
      bg = palette[hihostspp],
      xlim = c(-.03,.03),
      #ylim = c(-.5,.5),
      )
legend("bottomleft",
      legend=hispp,
      fill=palette,
      bty='n',
```

```

)
for(i in 1:length(hispp)){
ordihull(leafMDS, groups=leaf_data$Host_genus_species, show.groups=hispp[i], col=palet
}

#dev.off()

```



```
In [209]: testwoodhost <- adonis(leaf ~ leaf_data$Host_genus_species, permutations=10000)
testwoodhost
```

Call:  
`adonis(formula = leaf ~ leaf_data$Host_genus_species, permutations = 10000)`

Permutation: free  
Number of permutations: 10000

Terms added sequentially (first to last)

```

          Df SumsOfSqs MeanSqs F.Model      R2    Pr(>F)
leaf_data$Host_genus_species  33   21.061 0.63822  1.9528 0.41998 9.999e-05
Residuals                      89   29.087 0.32681
Total                          122   50.148
                                         1.00000

leaf_data$Host_genus_species ***
Residuals
Total
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

```

Quote the old statistical notebook (my words) again:

Okay, so we have unequal sample sizes (an "unbalanced" design), and unequal dispersions. There is clear separation in the NMS ordination among several groups, and stresses are reasonable. The main issue is groups with large dispersion but small sample size may create false positives of difference in a PERMANOVA model, because the centroids from these groups may be arbitrary, and may appear different from other groups when there isn't really enough data to confirm this. Check out Anderson et al 2013.

Assuming this NMS ordination is representative of the true dispersions, I'd say that we can trust our PERMANOVA model to a degree, in the sense that there is a significant difference in the location of the centroids among some of our groups. But I don't think we can get much information from the R2 value as an index of the importance of host.

PERMANOVA models of environmental variables

Environmental data were supplied by Dr. Su and colleagues. It is a little coarse for our study, but let's see if we pick up a signal, using PERMANOVA.

In [193]: `head(sample_data(deseq95), n=2)`

	vegcom	stream_distance	Host_genus	Host_genus_species	Library	Forest_Type	H
1w	2	24.11897	Engelhardtia	Engelhardtia_roxburghiana	W	7	rox
2w	2	23.22664	Pyrenaria	Pyrenaria_shinkoensis	W	7	shi

"Forest\_type" is a composite variable of numerous very localized topographical measurements taken from each 20 m x 20 m quadrat, which Dr. Su et al. have written about [here](#). Vegcom is a four-category classification of the above-ground woody plant community at each site of the Fushan FDP, written about in the [Fushan FDP manual](#). Distance to stream is a variable the Roo and I generated from GIS data. It is a little problematic to use a spatial variable as a continuous, uncorrected term in a pseudo-linear model like PERMANOVA, but we'll look at it anyway. In general, since we haven't explicitly addressed spatial questions or nested these by host effects, this really is a quick look for strong signals from environmental data, that probably also correlates with host and distance...

Wood environmental data

```
In [195]: ## get wood matrix:  
wood95 <- subset_samples(deseq95, Library=="W")  
woodOTU <- otu_table(wood95) ## otus are rows  
wood <- t(woodOTU@.Data) ## transpose so samples are rows  
all(rownames(sample_data(wood95)) == rownames(wood))
```

TRUE

```
In [196]: woodmod_VC <- adonis(wood ~ sample_data(wood95)$vegcom, permutations=10000)  
woodmod_FT <- adonis(wood ~ sample_data(wood95)$Forest_Type, permutations=10000)  
woodmod_SD <- adonis(wood ~ as.numeric(sample_data(wood95)$stream_distance), permutations=10000)
```

Wood endophyte community as a function of plant community:

```
In [198]: woodmod_VC
```

Call:

```
adonis(formula = wood ~ sample_data(wood95)$vegcom, permutations = 10000)
```

Permutation: free

Number of permutations: 10000

Terms added sequentially (first to last)

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
sample_data(wood95)\$vegcom	3	1.990	0.66320	1.7091	0.05565	9.999e-05 ***
Residuals	87	33.760	0.38804		0.94435	
Total	90	35.749			1.00000	

---

Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1 1

---

Wood endophyte community as a function of microtopography:

```
In [199]: woodmod_FT
```

Call:

```
adonis(formula = wood ~ sample_data(wood95)$Forest_Type, permutations = 10000)
```

Permutation: free

Number of permutations: 10000

Terms added sequentially (first to last)

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
--	----	-----------	---------	---------	----	--------

```

sample_data(wood95)$Forest_Type 6      3.168 0.52797 1.3612 0.08861 9.999e-05
Residuals                      84     32.581 0.38787          0.91139
Total                          90     35.749                  1.00000

sample_data(wood95)$Forest_Type ***
Residuals
Total
---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

```

---

Wood endophyte community as a function of distance-to-nearest-stream:

In [201]: woodmod\_SD

Call:

adonis(formula = wood ~ as.numeric(sample\_data(wood95)\$stream\_distance), permutations = 100)

Permutation: free

Number of permutations: 10000

Terms added sequentially (first to last)

	Df	SumsOfSqs	MeanSqs	F.Model
as.numeric(sample_data(wood95)\$stream_distance)	1	0.728	0.72769	1.8493
Residuals	89	35.022	0.39350	
Total	90	35.749		
	R2	Pr(>F)		
as.numeric(sample_data(wood95)\$stream_distance)	0.02036	9.999e-05 ***		
Residuals		0.97964		
Total		1.00000		
---				

Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1 1

So, yes, endophyte community dissimilarity varies somewhat with these variables environmental, but hard to know what exactly is going on, because all the variable being looked at (environmental variables, distance effects, hosts, etc..) are so correlated. By far host is the most suggestive, the above environmental data are giving a weak signal.

Leaf environmental data

In [202]: leaf95 <- subset\_samples(deseq95, Library=="L")  
leafOTU <- otu\_table(leaf95) ## otus are rows  
leaf <- t(leafOTU@.Data) ## transpose so samples are rows

In [203]: leafmod\_VC <- adonis(leaf ~ sample\_data(leaf95)\$vegcom, permutations=10000)  
leafmod\_FT <- adonis(leaf ~ sample\_data(leaf95)\$Forest\_Type, permutations=10000)  
leafmod\_SD <- adonis(leaf ~ as.numeric(sample\_data(leaf95)\$stream\_distance), permutations=10000)

---

Leaf endophyte community as a function of plant community:

In [204]: leafmod\_VC

Call:

```
adonis(formula = leaf ~ sample_data(leaf95)$vegcom, permutations = 10000)
```

Permutation: free

Number of permutations: 10000

Terms added sequentially (first to last)

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
sample_data(leaf95)\$vegcom	3	2.458	0.81930	2.0444	0.04901	9.999e-05 ***
Residuals	119	47.690	0.40076		0.95099	
Total	122	50.148			1.00000	
---						
Signif. codes:	0 ***	0.001 **	0.01 *	0.05 .	0.1	1

---

Leaf endophyte community as a function of microtopography:

In [205]: leafmod\_FT

Call:

```
adonis(formula = leaf ~ sample_data(leaf95)$Forest_Type, permutations = 10000)
```

Permutation: free

Number of permutations: 10000

Terms added sequentially (first to last)

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
sample_data(leaf95)\$Forest_Type	6	3.000	0.50005	1.2303	0.05983	0.005999
Residuals	116	47.147	0.40644		0.94017	
Total	122	50.148			1.00000	
---						
Signif. codes:	0 ***	0.001 **	0.01 *	0.05 .	0.1	1

sample\_data(leaf95)\$Forest\_Type \*\*

Residuals

Total

---

Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1 1

---

Leaf endophyte community as a function of distance-to-nearest-stream:

In [206]: leafmod\_SD

Call:

```
adonis(formula = leaf ~ as.numeric(sample_data(leaf95)$stream_distance), permutations = 100)
```

Permutation: free

Number of permutations: 10000

Terms added sequentially (first to last)

	Df	SumsOfSqs	MeanSqs	F.Model
as.numeric(sample_data(leaf95)\$stream_distance)	1	0.816	0.81642	2.0025
Residuals	121	49.331	0.40770	
Total	122	50.148		
	R2	Pr(>F)		
as.numeric(sample_data(leaf95)\$stream_distance)	0.01628	4e-04	***	
Residuals	0.98372			
Total	1.00000			
---				
Signif. codes:	0 ***	0.001 **	0.01 *	0.05 . 0.1 1

---

As per the wood, pretty weak sauce. Strongest signal remains host.

Cooccurrence network analysis

As with most modern microbial studies, we have 1000s of microbial OTUs to deal with here. Our hypothesis/question behind this study is concerned with the presence and behaviour of a "core" microbiome. How would we find this core, if it exists? How many needles in these haystacks can we find?

The most intuitive way to find "core fungi" that I can think of is to identify those fungi that seem to be most often co-occurring with a particular host, more often than we'd expect by chance. This group of candidate core members can then be observed in greater detail to see if their behaviour corresponds to something that can be called "core".

This is how I view this network analysis - it is sifting our haystacks for us. Once the candidate fungi are found, we take a closer look at their spatial patterns, to see if they behave anything like a "core" mycobiome. But first, we have to find them.

In [6]: `load('/home/daniel/Documents/taiwan/taiwan_combined_stats/tp.rda')`  
`load('deseq95.rda')`

In [288]: tp

Sample_ID	xx	yy	Host_family	Host_genus	Host_species	stream_distance
1	360	220	Juglandaceae	Engelhardtia	roxburghiana	24.118967
2	360	221	Theaceae	Pyrenaria	shinkoensis	23.226639
3	361	221	Proteaceae	Helicia	formosana	22.775251
4	361	220	Theaceae	Pyrenaria	shinkoensis	23.667579
5	363	220	Theaceae	Pyrenaria	shinkoensis	22.764803
6	363	223	Proteaceae	Helicia	formosana	20.087820
7	360	223	Juglandaceae	Engelhardtia	roxburghiana	21.441983
8	360	227	Proteaceae	Helicia	formosana	17.872671
9	360	235	Sabiaceae	Meliosma	squamulata	10.734048
10	360	251	Lauraceae	Machilus	thunbergii	3.543198
11	360	283	Euphorbiaceae	Antidesma	japonicum var. densiflorum	25.900889
12	360	347	Rubiaceae	Randia	cochininchinensis	53.538260
13	367	220	Theaceae	Cleyera	japonica	20.959252
14	375	220	Proteaceae	Helicia	formosana	17.585413
15	391	220	Theaceae	Adinandra	formosana	10.154351
16	423	220	Rubiaceae	Randia	cochininchinensis	4.772709
17	487	220	Rubiaceae	Lasianthus	fordii	32.998484
18	487	347	Proteaceae	Helicia	formosana	23.697306
19	423	283	Rubiaceae	Tricalysia	dubia	10.336297
20	391	251	Cyatheaceae	Cyathea	podophylla	6.134630
21	375	235	Theaceae	Cleyera	japonica	3.963231
22	367	227	Fagaceae	Limlia	uraiana	14.712957
23	360	360	Melastomataceae	Blastus	cochininchinensis	61.456433
24	360	361	Melastomataceae	Blastus	cochininchinensis	62.136425
25	361	361	Proteaceae	Helicia	formosana	62.869333
26	361	360	Verbenaceae	Callicarpa	kochiana	62.197354
27	360	364	Melastomataceae	Melastoma	kochiana	62.061713
28	363	363	Proteaceae	Helicia	formosana	61.359117
29	363	360	Verbenaceae	Callicarpa	kochiana	63.700442
30	367	360	Lauraceae	Machilus	japonica var. kusanoi	60.324927
104	180	51	Proteaceae	Helicia	formosana	1.409998
105	180	83	Fagaceae	Limlia	uraiana	22.467219
106	180	147	Proteaceae	Helicia	formosana	82.497336
107	307	147	Melastomataceae	Blastus	cochininchinensis	64.858760
108	243	83	Theaceae	Cleyera	japonica	19.021132
109	209	49	Sabiaceae	Meliosma	squamulata	13.468151
110	209	49	Fagaceae	Limlia	uraiana	13.468151
111	195	35	Fagaceae	Limlia	uraiana	20.379733
112	167	40	Melastomataceae	Blastus	cochininchinensis	4.250151
113	103	40	Araliaceae	Schefflera	octophylla	12.803846
114	71	40	Proteaceae	Helicia	formosana	6.015914
115	55	40	Cyatheaceae	Cyathea	podophylla	16.758109
116	47	40	Proteaceae	Helicia	formosana	24.211802
117	40	40	Fagaceae	Limlia	uraiana	30.795823
118	40	41	Fagaceae	Limlia	uraiana	30.501391
119	41	41	Fagaceae	Limlia	uraiana	29.542556
120	41	40	Fagaceae	Limlia	uraiana	29.846447
121	43	40	Melastomataceae	Blastus	cochininchinensis	27.958288
122	43	43	Fagaceae	Limlia	uraiana	27.076212
123	40	43	Melastomataceae	Blastus	cochininchinensis	29.970182
124	40	47	Euphorbiaceae	Glochidion	acuminatum	28.916139

## Wood cooccurrence networks

Old notebook:

Let's see if there are any strong cooccurrence relationships in the wood endophytes. First, some reformatting of data is necessary. We are looking to see if any wood endophytes are particularly "loyal" to a given host, so we need a single matrix of both wood endophyte OTUs and hosts, and which sites they occur at. We have this info in a general dataframe which I constructed of our site data oh-so-many-years ago. Load it here, and mess around with it to fit our formatting needs:

Wow. And I wrote that "oh-so-many" years ago... I think I will be working on this project for the rest of my life.

Anyway, reformat the community matrix of wood reads, and site dataframe:

```
In [289]: ## get the necessary data objects
    load('deseq95.rda') ## our variance-stabilized biom
    wood95 <- subset_samples(deseq95, Library=="W") ## subset biom to wood
    woodOTU <- otu_table(wood95) ## taxa are rows.
```

```
In [290]: hostmat <- model.matrix(~0+Host_genus_species+Sample_ID,data=tp)
```

```
In [253]: head(hostmat, n=2); tail(hostmat, n=2)
```

	Host_genus_speciesAdinandra_formosana	Host_genus_speciesAntidesma_japonicum_var_densiflorum
1	0	0
2	0	0
	Host_genus_speciesAdinandra_formosana	Host_genus_speciesAntidesma_japonicum_var_densiflorum
132	0	0
133	0	0

```
In [291]: hostmat <- hostmat[,-35] ## last column is redundant
    colnames(hostmat) <- gsub('Host_genus_species','',colnames(hostmat)) ## clean up host
    hostmat <- t(hostmat) ## transpose so hosts are rows.
```

```
In [292]: head(hostmat)
```

	1	2	3	4	5	6	7	8	9	10	124	125	126	127	128
Adinandra_formosana	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Antidesma_japonicum_var_densiflorum	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Blastus_cochinchinensis	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Callicarpa_kochiana	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Castanopsis_cuspidata_var_carlesii	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Cleyera_japonica	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

```
In [293]: colnames(woodOTU) <- gsub('w','',colnames(woodOTU)) ## drop the "w" in the woodOTU place
```

Not sure if it matters for the cooccurrence package, but let's remove zero-row OTUs (~ the leaf OTUs).

```
In [294]: dim(woodOTU) ## size before zero removal  
woodOTU <- woodOTU[rowSums(woodOTU) != 0,]  
dim(woodOTU) ## size after zero removal
```

1. 3327 2. 91

1. 2025 2. 91

Subset to sites where we have data on wood endophytes.

```
In [295]: colnames(hostmat) %in% colnames(woodOTU)
```

```
1. TRUE 2. TRUE 3. TRUE 4. TRUE 5. TRUE 6. FALSE 7. TRUE 8. FALSE 9. TRUE 10. TRUE  
11. TRUE 12. FALSE 13. TRUE 14. TRUE 15. TRUE 16. TRUE 17. TRUE 18. TRUE 19. TRUE  
20. TRUE 21. TRUE 22. FALSE 23. TRUE 24. FALSE 25. TRUE 26. TRUE 27. TRUE 28. TRUE  
29. TRUE 30. TRUE 31. TRUE 32. TRUE 33. FALSE 34. FALSE 35. TRUE 36. TRUE 37. TRUE  
38. TRUE 39. TRUE 40. FALSE 41. FALSE 42. TRUE 43. FALSE 44. FALSE 45. TRUE 46. TRUE  
47. FALSE 48. FALSE 49. TRUE 50. TRUE 51. FALSE 52. TRUE 53. FALSE 54. FALSE 55. TRUE  
56. TRUE 57. TRUE 58. TRUE 59. TRUE 60. TRUE 61. TRUE 62. FALSE 63. FALSE 64. TRUE  
65. FALSE 66. TRUE 67. TRUE 68. TRUE 69. TRUE 70. TRUE 71. TRUE 72. TRUE 73. TRUE  
74. TRUE 75. TRUE 76. TRUE 77. FALSE 78. FALSE 79. TRUE 80. FALSE 81. TRUE 82. TRUE  
83. FALSE 84. TRUE 85. TRUE 86. TRUE 87. TRUE 88. TRUE 89. TRUE 90. FALSE 91. FALSE  
92. TRUE 93. TRUE 94. TRUE 95. TRUE 96. TRUE 97. TRUE 98. FALSE 99. TRUE 100. TRUE  
101. TRUE 102. TRUE 103. FALSE 104. TRUE 105. FALSE 106. TRUE 107. TRUE 108. TRUE  
109. TRUE 110. FALSE 111. FALSE 112. FALSE 113. FALSE 114. TRUE 115. TRUE 116. FALSE  
117. FALSE 118. FALSE 119. FALSE 120. FALSE 121. TRUE 122. FALSE 123. FALSE 124. TRUE  
125. TRUE 126. FALSE 127. FALSE 128. TRUE 129. TRUE 130. TRUE 131. TRUE 132. FALSE  
133. TRUE
```

```
In [296]: ## subset columns in our host matrix  
hostmat <- hostmat[, colnames(hostmat) %in% colnames(woodOTU)]  
## drop the zero rows:  
hostmat <- hostmat[rowSums(hostmat)>0,]
```

```
In [297]: ## sanity checks  
dim(woodOTU); dim(hostmat);  
all(colnames(hostmat) %in% colnames(woodOTU)); all(colnames(woodOTU) %in% colnames(hos
```

1. 2025 2. 91

1. 30 2. 91

TRUE

TRUE

```
In [298]: head(hostmat, n=2)
```

	1	2	3	4	5	7	9	10	11	13	114	115	121	124
Adinandra_formosana	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Antidesma_japonicum_var_densiflorum	0	0	0	0	0	0	0	0	1	0	0	0	0	0

```
In [299]: head(woodOTU, n=2)
```

	1	2	3	4	5	7	9	10	11	13	114	115	121	124	125	128	129
OTU6797:111leaf	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6.625067
OTU33:Dc-X	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.000000

Stack 'em. And make them presence/absence, I don't trust the read abundances and the host abundances can't be more than one, anyway, so it would be pretty weird not to.

```
In [300]: host_woodOTU_mat <- rbind(hostmat, woodOTU)
           host_woodOTU_mat[host_woodOTU_mat > 0] <- 1
```

```
In [301]: host_woodOTU_mat
```



```
In [304]: #save(host_woodOTU_mat, file="host_woodOTU_mat.rda")
```

```
In [306]: woodcooc <- cooccur(host_woodOTU_mat, spp_names=TRUE)
```

```
|-----| 100%
```

```
In [313]: str(woodcooc)
```

List of 16

```
$ call                  : language cooccur(mat = host_woodOTU_mat, spp_names = TRUE)
$ results               : 'data.frame':      24882 obs. of  11 variables:
..$ sp1                 : num [1:24882] 2 2 2 2 2 2 2 2 2 ...
..$ sp2                 : num [1:24882] 58 61 62 67 72 79 97 124 148 184 ...
..$ sp1_inc              : num [1:24882] 3 3 3 3 3 3 3 3 3 ...
..$ sp2_inc              : num [1:24882] 38 37 58 33 43 77 72 35 64 43 ...
..$ obs_cooccur          : num [1:24882] 1 1 2 1 0 3 3 1 3 2 ...
..$ prob_cooccur         : num [1:24882] 0.014 0.013 0.021 0.012 0.016 0.028 0.026 0.013 0.023 0.016 ...
..$ exp_cooccur          : num [1:24882] 1.3 1.2 1.9 1.1 1.4 2.5 2.4 1.2 2.1 1.4 ...
..$ p_lt                 : num [1:24882] 0.624 0.64 0.746 0.703 0.142 ...
..$ p_gt                 : num [1:24882] 0.807 0.796 0.703 0.746 1 ...
..$ sp1_name              : Factor w/ 2055 levels "Adinandra_formosana",...: 2 2 2 2 2 2 2 2 2 ...
..$ sp2_name              : Factor w/ 2055 levels "Adinandra_formosana",...: 488 1706 1841 828 306 1921 9
$ positive               : int 2006
$ negative               : int 251
$ co_occurrences         : int 2257
$ pairs                  : int 24882
$ random                 : int 22625
$ unclassifiable         : int 0
$ sites                  : int [1:2055, 1:2055] 91 91 91 91 91 91 91 91 91 91 ...
$ species                : int 2055
$ percent_sig             : num 9.07
$ true_rand_classifier: num 0.1
$ spp_key                : 'data.frame':      2055 obs. of  2 variables:
..$ num: int [1:2055] 1 2 3 4 5 6 7 8 9 10 ...
..$ spp: Factor w/ 2055 levels "Adinandra_formosana",...: 1 2 3 4 5 6 7 8 9 10 ...
$ spp.names              : chr [1:2055] "Adinandra_formosana" "Antidesma_japonicum_var_densiflorum"
$ omitted                 : int 2085603
$ pot_pairs               : int 2110485
- attr(*, "class")= chr "cooccur"
```

We have our cooccur object... let's pull out the probabilities and effect sizes:

```
In [318]: woodcoocP <- prob.table(woodcooc)
```

```
Warning message in prob.table(woodcooc):
```

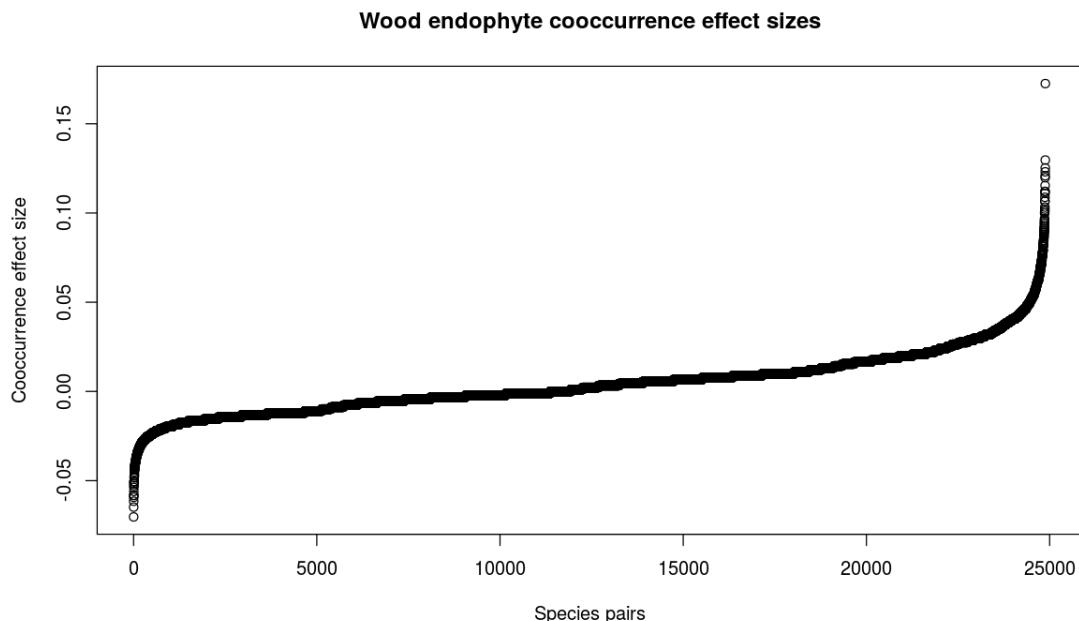
```
The co-occurrence model was run using 'thresh = TRUE.' The probability table may not include all
```

```
In [319]: woodcoocC <- effect.sizes(woodcooc)
```

```
In [320]: woodcoocPC <- cbind(woodcoocP, woodcoocC)
```

This allows us to visualize the cooccurrence relationships in general:

```
In [326]: options(repr.plot.width = 10,repr.plot.height = 6)
plot(sort(woodcoocPC$effect),
      xlab='Species pairs',
      ylab = 'Cooccurrence effect size',
      main='Wood endophyte cooccurrence effect sizes')
```



Pretty standard sigmoidal curve, most of the species don't really seem to have effect on each other, and at either end we have some stronger negative and positive effects.

This is a massive number of tests, ~10,000. So let's correct with a benjamini-hochberg adjustment, and pick out interactions with an fdr of .05 or less.

```
In [328]: ## add a corrected pvalue, benjamini-hochberg
woodcoocPC$p_gt_adj <- p.adjust(woodcoocPC$p_gt, method = "BH")
#save(woodcoocPC,file='woodcoocPC.rda')
## look for only strong effects, sig p vals
strong_woodcooc <- woodcoocPC[woodcoocPC$p_gt_adj <= 0.05,]
```

```
In [330]: ## clean up col names
strong_woodcooc <- strong_woodcooc[,-c(12,13)]
colnames(strong_woodcooc)[10:11] <- c('OTU_A','OTU_B')
```

```
In [333]: head(strong_woodcooc)
```

	sp1	sp2	sp1_inc	sp2_inc	obs_cooccur	prob_cooccur	exp_cooccur	p_lt	p_gt	OTU
294	13	187	22	20	15	0.053	4.8	1.00000	0.00000	Heli
298	13	197	22	17	11	0.045	4.1	0.99999	0.00008	Heli
326	13	248	22	9	9	0.024	2.2	1.00000	0.00000	Heli
374	13	420	22	26	22	0.069	6.3	1.00000	0.00000	Heli
384	13	435	22	24	13	0.064	5.8	0.99998	0.00017	Heli
386	13	456	22	10	9	0.027	2.4	1.00000	0.00001	Heli

In [340]: strong\_woodcooc

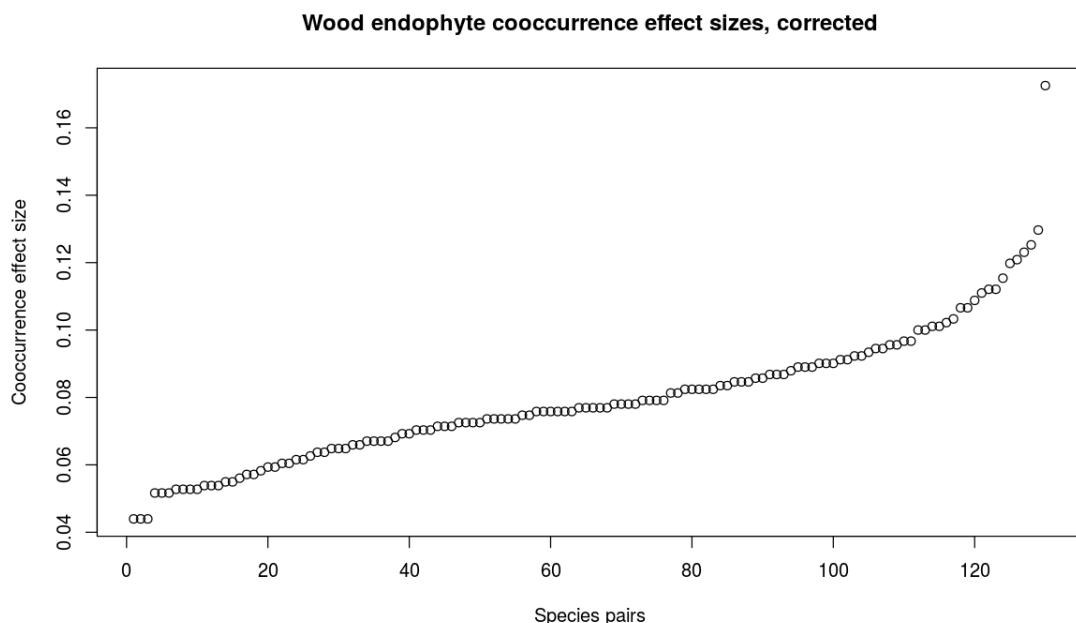
	sp1	sp2	sp1_inc	sp2_inc	obs_cooccur	prob_cooccur	exp_cooccur	p_lt	p_gt	Co
294	13	187	22	20	15	0.053	4.8	1.00000	0.00000	F
298	13	197	22	17	11	0.045	4.1	0.99999	0.00008	F
326	13	248	22	9	9	0.024	2.2	1.00000	0.00000	F
374	13	420	22	26	22	0.069	6.3	1.00000	0.00000	F
384	13	435	22	24	13	0.064	5.8	0.99998	0.00017	F
386	13	456	22	10	9	0.027	2.4	1.00000	0.00001	F
432	13	725	22	15	13	0.040	3.6	1.00000	0.00000	F
618	23	717	5	19	5	0.011	1.0	1.00000	0.00025	M
1498	56	62	25	58	25	0.175	15.9	1.00000	0.00000	C
1609	56	237	25	25	17	0.075	6.9	1.00000	0.00000	C
1616	56	255	25	22	14	0.066	6.0	1.00000	0.00004	C
1909	57	164	23	8	7	0.022	2.0	0.99999	0.00020	C
2191	58	65	38	19	16	0.087	7.9	1.00000	0.00003	C
2193	58	67	38	33	25	0.151	13.8	1.00000	0.00000	C
2194	58	68	38	15	13	0.069	6.3	0.99999	0.00015	C
2196	58	70	38	27	20	0.124	11.3	0.99999	0.00006	C
2197	58	72	38	43	29	0.197	18.0	1.00000	0.00000	C
2243	58	144	38	15	13	0.069	6.3	0.99999	0.00015	C
2248	58	151	38	21	18	0.096	8.8	1.00000	0.00000	C
2251	58	158	38	13	12	0.060	5.4	1.00000	0.00008	C
2286	58	208	38	22	18	0.101	9.2	1.00000	0.00002	C
2429	58	495	38	15	14	0.069	6.3	1.00000	0.00001	C
2560	58	898	38	12	12	0.055	5.0	1.00000	0.00001	C
2676	61	64	37	20	19	0.089	8.1	1.00000	0.00000	C
2757	61	187	37	20	16	0.089	8.1	0.99999	0.00007	C
2815	61	278	37	30	24	0.134	12.2	1.00000	0.00000	C
3163	62	70	58	27	25	0.189	17.2	0.99999	0.00010	C
3247	62	184	58	43	36	0.301	27.4	0.99998	0.00016	C
3299	62	255	58	22	21	0.154	14.0	0.99999	0.00016	C
3479	62	512	58	26	25	0.182	16.6	1.00000	0.00001	C
16777	203	230	24	22	14	0.064	5.8	1.00000	0.00002	C
16839	203	366	24	12	9	0.035	3.2	0.99999	0.00021	C
17438	208	908	22	8	7	0.021	1.9	1.00000	0.00014	C
17474	210	225	30	32	21	0.116	10.5	1.00000	0.00000	C
17521	210	324	30	14	11	0.051	4.6	0.99998	0.00020	C
18470	217	278	15	30	12	0.054	4.9	1.00000	0.00006	C
18608	219	371	20	36	16	0.087	7.9	1.00000	0.00004	C
19499	230	756	22	15	10	0.040	3.6	0.99999	0.00014	C
19949	243	324	22	14	10	0.037	3.4	1.00000	0.00006	C
20154	248	321	9	20	7	0.022	2.0	0.99999	0.00026	C
20166	248	420	9	26	9	0.028	2.6	1.00000	0.00000	C
20337	255	1160	22	10	9	0.027	2.4	1.00000	0.00001	C
20431	261	416	21	20	12	0.051	4.6	1.00000	0.00005	C
20439	261	429	21	8	7	0.020	1.8	1.00000	0.00010	C
22115	306	350	5	19	5	0.011	1.0	1.00000	0.00025	C
22454	328	717	12	19	8	0.028	2.5	0.99998	0.00026	C
22610	345	350	16	19	11	0.037	3.3	1.00000	0.00000	C
23009	366	717	12	19	8 <sub>85</sub>	0.028	2.5	0.99998	0.00026	C
23021	366	934	12	9	6	0.013	1.2	1.00000	0.00010	C
23426	416	434	20	20	12	0.048	4.4	1.00000	0.00002	C
23431	416	459	20	14	10	0.034	3.1	1.00000	0.00002	C

```
In [334]: dim(strong_woodcooc)
```

1. 130 2. 13

Okay, now that we have reduced our interactions to those with an fdr below .05, what's left standing? Looking at the effect sizes:

```
In [337]: options(repr.plot.width = 10,repr.plot.height = 6)
plot(sort(strong_woodcooc$effect),
     xlab='Species pairs',
     ylab = 'Cooccurrence effect size',
     main='Wood endophyte cooccurrence effect sizes, corrected')
```



Most interactions drop off, including all negative ones (because I used the p-value associated with positive interactions, "p\_gt"). Let's graph these remaining cooccurrences, using the Section ?? package.

```
In [338]: aa <- strong_woodcooc[,c(10,11,12,13)]
graphcooc_wood <- graph_from_data_frame(aa, directed=FALSE) ## quick igraph object
```

```
In [339]: #save(graphcooc_wood, file = 'graphcooc_wood.rda')
```

```
In [106]: #load('graphcooc_wood.rda')
```

```
In [109]: vcols <- vector(length = length(V(graphcooc_wood)))
vcols[] <- 'gray'
vcols[which(names(V(graphcooc_wood)) == "Helicia_formosana")] <- "red"
vcols[which(names(V(graphcooc_wood)) == "Myrsine_seguinii")] <- "darkgreen"
```

```

aa <- adjacent_vertices(graphcooc_wood, "Helicia_formosana", mode = c("all"))[[1]]
vcols[V(graphcooc_wood) %in% aa] <- "lightpink" ## color first-degrees purple
bb <- adjacent_vertices(graphcooc_wood, "Myrsine_seguinii", mode = c("all"))[[1]]
vcols[V(graphcooc_wood) %in% bb] <- "lightgreen" ## color first-degrees purple
helwoodfung <- names(aa) ## useful below, in mapping of core mycobiomes

```

In [204]: helwoodfung

1. 'OTU352:1w'
2. 'OTU726:72w'
3. 'OTU269:1w'
4. 'OTU287:3w'
5. 'OTU250:4w'
6. 'OTU257:3w'
7. 'OTU84:38w'

In [200]: `#save(helwoodfung, file="helwoodfung.rda")`

And take a look. The purple nodes are first degree associates of the host tree *Helicia formosana*, what we will call the core mycobiome of *H. formosana*.

```

In [112]: svg('woodCoccurrence.svg')
plot(graphcooc_wood,
      vertex.color = vcols,
      vertex.size = 8,
      vertex.label = NA,
      edge.color = 'black')
dev.off()

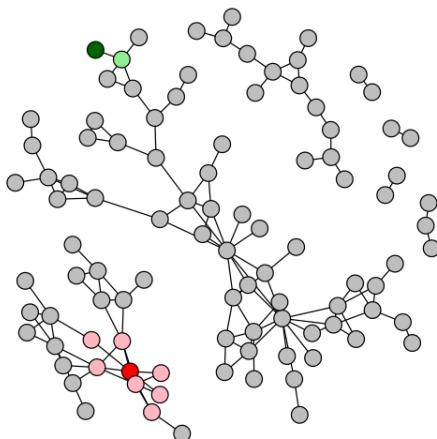
```

**png:** 2

```

In [378]: plot(graphcooc_wood,
              vertex.color = vcols,
              vertex.size = 8,
              vertex.label = NA,
              edge.color = 'black')

```



## Leaf cooccurrence networks

```
In [72]: leaf95 <- subset_samples(deseq95, Library=="L") ## subset biom to leaf
leafOTU <- otu_table(leaf95) ## taxa are rows.

hostmat <- model.matrix(~0+Host_genus_species+Sample_ID,data=tp)
hostmat <- hostmat[,-35]
colnames(hostmat) <- gsub('Host_genus_species',' ',colnames(hostmat))
hostmat <- t(hostmat)

In [73]: ## get rid of underscores:
colnames(leafOTU) <- gsub('leaf.?', ' ', colnames(leafOTU))

In [74]: head(hostmat, n=1)



|                     | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 124 | 125 | 126 | 127 | 128 | 129 | 130 | 1 |
|---------------------|---|---|---|---|---|---|---|---|---|----|-----|-----|-----|-----|-----|-----|-----|---|
| Adinandra_formosana | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0  | 0   | 0   | 0   | 0   | 0   | 0   | 0   |   |



In [75]: head(leafOTU, n=1)



|               | 100      | 101      | 102 | 103      | 104 | 105 | 107 | 108 | 109 | 110 | 90 | 92 | 93 |
|---------------|----------|----------|-----|----------|-----|-----|-----|-----|-----|-----|----|----|----|
| OTU19:100leaf | 13.61471 | 4.479497 | 0   | 9.567495 | 0   | 0   | 0   | 0   | 0   | 0   | 0  | 0  | 0  |



In [76]: ## order:
aa <- order(as.numeric(colnames(leafOTU)))
leafOTU <- leafOTU[,aa]
## subset columns in our host matrix
hostmat <- hostmat[,colnames(hostmat) %in% colnames(leafOTU)]
## worked?
all(colnames(hostmat) == colnames(leafOTU))

TRUE

In [77]: #Stack them, reduce leaf OTU reads to presence/absence:
host_leafOTU_mat <- rbind(hostmat, leafOTU)
host_leafOTU_mat[host_leafOTU_mat > 0] <- 1

In [79]: save(host_leafOTU_mat, file="host_leafOTU_mat.rda")

In [78]: leafcooc <- cooccur(host_leafOTU_mat, spp_names=TRUE)

| ====== | 100%

```

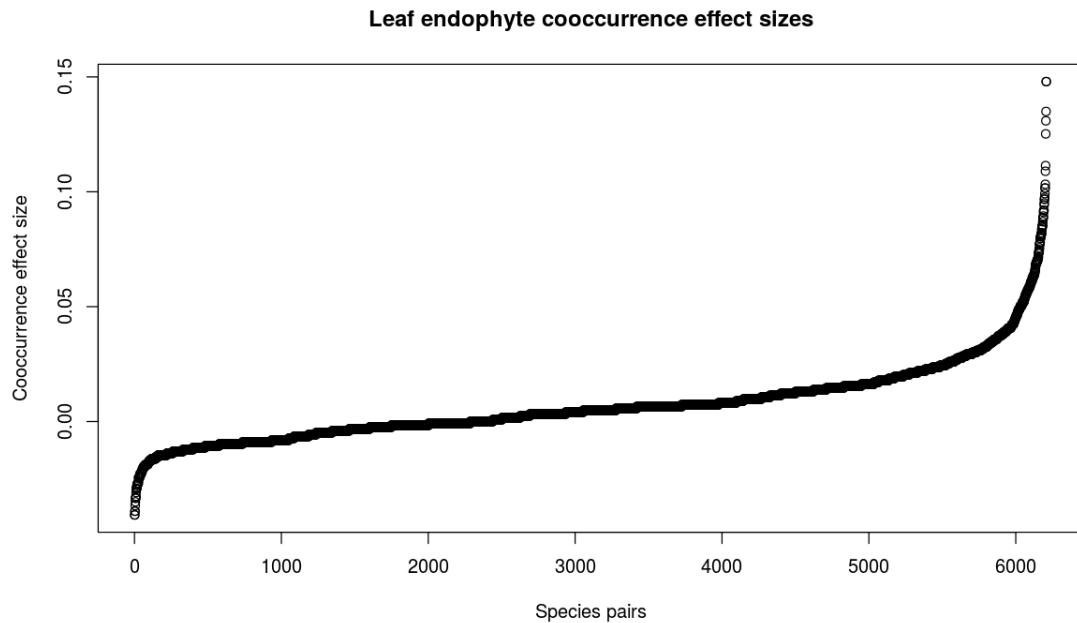
```
In [80]: save(leafcooc, file='leafcooc.rda')

In [81]: leafcoocP <- prob.table(leafcooc)
leafcoocC <- effect.sizes(leafcooc)
leafcoocPC <- cbind(leafcoocP, leafcoocC)
```

```
Warning message in prob.table(leafcooc):
```

```
The co-occurrence model was run using 'thresh = TRUE.' The probability table may not include all
```

```
In [82]: options(repr.plot.width = 10,repr.plot.height = 6)
plot(sort(leafcoocPC$effect),
      xlab='Species pairs',
      ylab = 'Cooccurrence effect size',
      main='Leaf endophyte cooccurrence effect sizes')
```



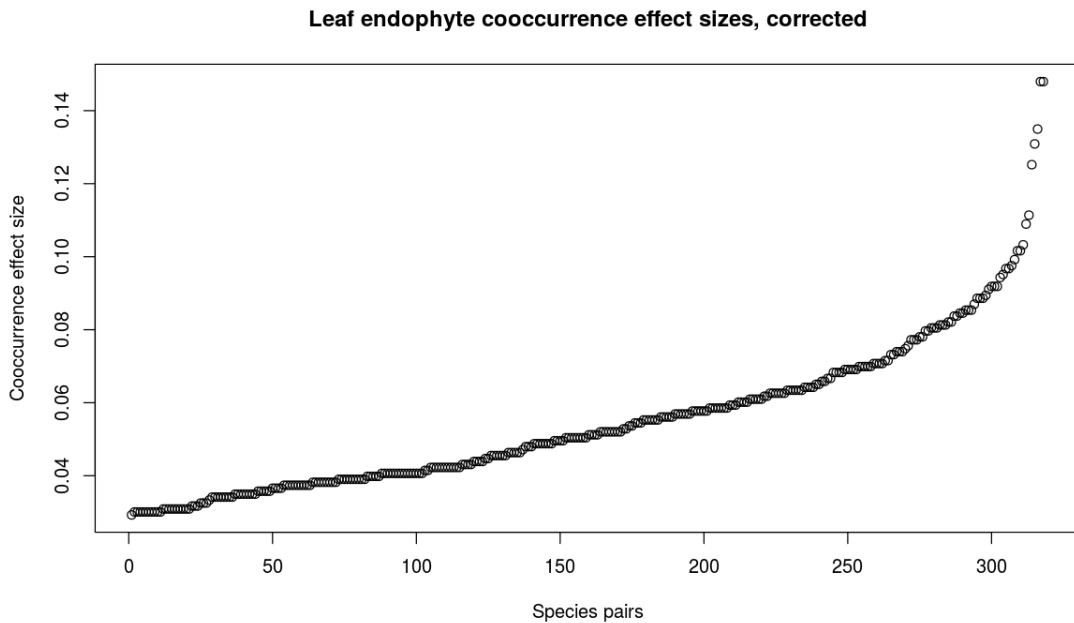
```
In [83]: ## add a corrected pvalue, benjamini-hochberg
leafcoocPC$p_gt_adj <- p.adjust(leafcoocPC$p_gt, method = "BH")
#save(leafcoocPC,file='leafcoocPC.rda')
## look for only strong effects, sig p vals
strong_leafcooc <- leafcoocPC[leafcoocPC$p_gt_adj <= 0.05,]

## clean up col names
strong_leafcooc <- strong_leafcooc[,-c(12,13)]
colnames(strong_leafcooc)[10:11] <- c('OTU_A','OTU_B')
```

```
In [86]: dim(strong_leafcooc)
```

```
1. 318 2. 13
```

```
In [87]: options(repr.plot.width = 10,repr.plot.height = 6)
plot(sort(strong_leafcooc$effect),
      xlab='Species pairs',
      ylab = 'Cooccurrence effect size',
      main='Leaf endophyte cooccurrence effect sizes, corrected')
```



```
In [89]: aa <- strong_leafcooc[,c(10,11,12,13)]
graphcooc_leaf <- graph_from_data_frame(aa, directed=FALSE) ## quick igraph object

In [90]: #save(graphcooc_leaf, file = 'graphcooc_leaf.rda')

In [184]: #load(file = 'graphcooc_leaf.rda')

In [185]: vcols <- vector(length = length(V(graphcooc_leaf)))
vcols[] <- 'gray'

In [186]: names(V(graphcooc_leaf))

1. 'Blastus_cochinchinensis' 2. 'Helicia_formosana' 3. 'Limlia_uriana' 4. 'OTU19:100leaf'
5. 'OTU1:100leaf' 6. 'OTU202:100leaf' 7. 'OTU10:100leaf' 8. 'OTU17:100leaf' 9. 'OTU15:100leaf'
10. 'OTU27:100leaf' 11. 'OTU355:100leaf' 12. 'OTU22:101leaf' 13. 'OTU181:100leaf'
14. 'OTU73:100leaf' 15. 'OTU159:100leaf' 16. 'OTU297:100leaf' 17. 'OTU252:117leaf'
18. 'OTU78:100leaf' 19. 'OTU226:100leaf' 20. 'OTU209:101leaf' 21. 'OTU368:100leaf'
22. 'OTU883:107leaf' 23. 'OTU1938:130leaf' 24. 'OTU93:100leaf' 25. 'OTU263:100leaf'
26. 'OTU319:100leaf' 27. 'OTU360:100leaf' 28. 'OTU247:17leaf' 29. 'OTU45:100leaf'
30. 'OTU53:107leaf' 31. 'OTU280:100leaf' 32. 'OTU97:102leaf' 33. 'OTU94:114leaf'
34. 'OTU340:103leaf' 35. 'OTU462:101leaf' 36. 'OTU124:101leaf' 37. 'OTU405:101leaf'
38. 'OTU436:102leaf' 39. 'OTU115:101leaf' 40. 'OTU343:103leaf' 41. 'OTU546:101leaf'
42. 'OTU116:117leaf' 43. 'OTU218:102leaf' 44. 'OTU663:110leaf' 45. 'OTU489:125leaf'
46. 'OTU1025:3leaf' 47. 'OTU208:103leaf' 48. 'OTU65:102leaf' 49. 'OTU309:117leaf'
50. 'OTU86:102leaf' 51. 'OTU14:102leaf' 52. 'OTU479:104leaf' 53. 'OTU605:103leaf'
54. 'OTU398:103leaf' 55. 'OTU25:103leaf' 56. 'OTU606:103leaf' 57. 'OTU39:103leaf'
```

```

58. 'OTU75:104leaf' 59. 'OTU170:105leaf' 60. 'OTU483:111leaf' 61. 'OTU51:109leaf'
62. 'OTU47:110leaf' 63. 'OTU416:110leaf' 64. 'OTU369:110leaf' 65. 'OTU215:110leaf'
66. 'OTU80:110leaf' 67. 'OTU8:105leaf' 68. 'OTU91:100leaf' 69. 'OTU239:100leaf'
70. 'OTU16:100leaf' 71. 'OTU199:101leaf' 72. 'OTU678:131leaf' 73. 'OTU183:110leaf'
74. 'OTU749:116leaf' 75. 'OTU613:100leaf' 76. 'OTU11:103leaf' 77. 'OTU878:96leaf'
78. 'OTU371:110leaf' 79. 'OTU26:110leaf' 80. 'OTU457:101leaf' 81. 'OTU454:101leaf'
82. 'OTU509:102leaf' 83. 'OTU242:103leaf' 84. 'OTU244:115leaf' 85. 'OTU1005:81leaf'
86. 'OTU303:111leaf' 87. 'OTU470:103leaf' 88. 'OTU557:116leaf' 89. 'OTU440:102leaf'
90. 'OTU48:102leaf' 91. 'OTU223:104leaf' 92. 'OTU472:111leaf' 93. 'OTU31:105leaf'
94. 'OTU135:117leaf' 95. 'OTU464:111leaf' 96. 'OTU23:119leaf'

```

```

In [187]: vcols[which(names(V(graphcooc_leaf)) == "Helicia_formosana")] <- "red"
vcols[which(names(V(graphcooc_leaf)) == "Blastus_cochinchinensis")] <- "blue"
vcols[which(names(V(graphcooc_leaf)) == "Limlia_uraiana")] <- "orange"

aa <- adjacent_vertices(graphcooc_leaf, "Helicia_formosana", mode = c("all"))[[1]]
vcols[V(graphcooc_leaf) %in% aa] <- "lightpink" ## color first-degrees
bb <- adjacent_vertices(graphcooc_leaf, "Blastus_cochinchinensis", mode = c("all"))[[1]]
vcols[V(graphcooc_leaf) %in% bb] <- "lightblue" ## color first-degrees
cc <- adjacent_vertices(graphcooc_leaf, "Limlia_uraiana", mode = c("all"))[[1]]
vcols[V(graphcooc_leaf) %in% cc] <- "yellow" ## color first-degrees

helleaffung <- names(aa)
#save(helleaffung, file= 'helleaffung.rda') ## useful below, in mapping of core mycobacteria

```

```
In [102]: helleaffung
```

```

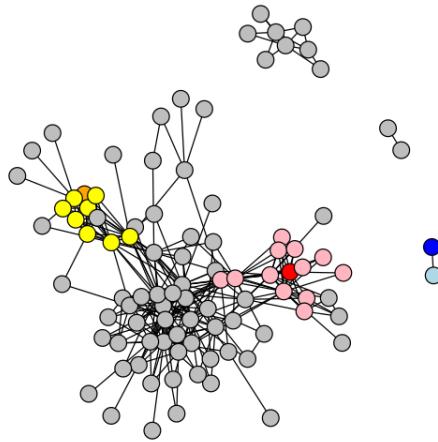
1. 'OTU19:100leaf' 2. 'OTU202:100leaf' 3. 'OTU10:100leaf' 4. 'OTU17:100leaf'
5. 'OTU15:100leaf' 6. 'OTU27:100leaf' 7. 'OTU94:114leaf' 8. 'OTU91:100leaf' 9. 'OTU239:100leaf'
10. 'OTU16:100leaf' 11. 'OTU199:101leaf' 12. 'OTU678:131leaf'

```

```

In [101]: plot(graphcooc_leaf,
            vertex.color = vcols,
            vertex.size = 8,
            vertex.label = NA,
            edge.color = 'black')

```



## Community composition

Let's look at what's in our leaf and wood endophyte communities.

```
In [4]: load('deseq95.rda')
woodPA <- subset_samples(deseq95, Library == 'W')
leafPA <- subset_samples(deseq95, Library == 'L')
leafOTU <- otu_table(leafPA) ## extract otu table matrix
leafOTU[leafOTU > 0] <- 1 ## convert to P/A
leafOTU -> otu_table(leafPA) ## put it back in
woodOTU <- otu_table(woodPA) ## same with wood
woodOTU[woodOTU > 0] <- 1
woodOTU -> otu_table(woodPA)
leafPA <- prune_taxa(rowSums(otu_table(leafPA)) > 0, leafPA) ## get rid of empty taxa
woodPA <- prune_taxa(rowSums(otu_table(woodPA)) > 0, woodPA)
```

```
In [106]: aa <- names(taxa_sums(leafPA))
bb <- names(taxa_sums(woodPA))
venn.diagram(list(Leaf = aa, Wood = bb),
             fill = c("green", "brown"),
             alpha = c(0.3, 0.7),
             cex = 2,
             cat.fontface = 4,
             fontfamily =3,
             imagetype = 'png',
             filename='overallvenn.png',
             )
```

```
1
<img src='overallvenn.png', width=400, height=400>
No shared species. But more species in general. Not sure why, there are couple places in the pipeline that could have caused this.
```

We can look at how many of our OTUs in each belong to Ascomycota, how many to basidiomycota, etc. Note that we are making no ecological statements here, doesn't matter if an OTU was observed from one site or all sites, they are all weighted equally as present in the study in wood or leaf tissue

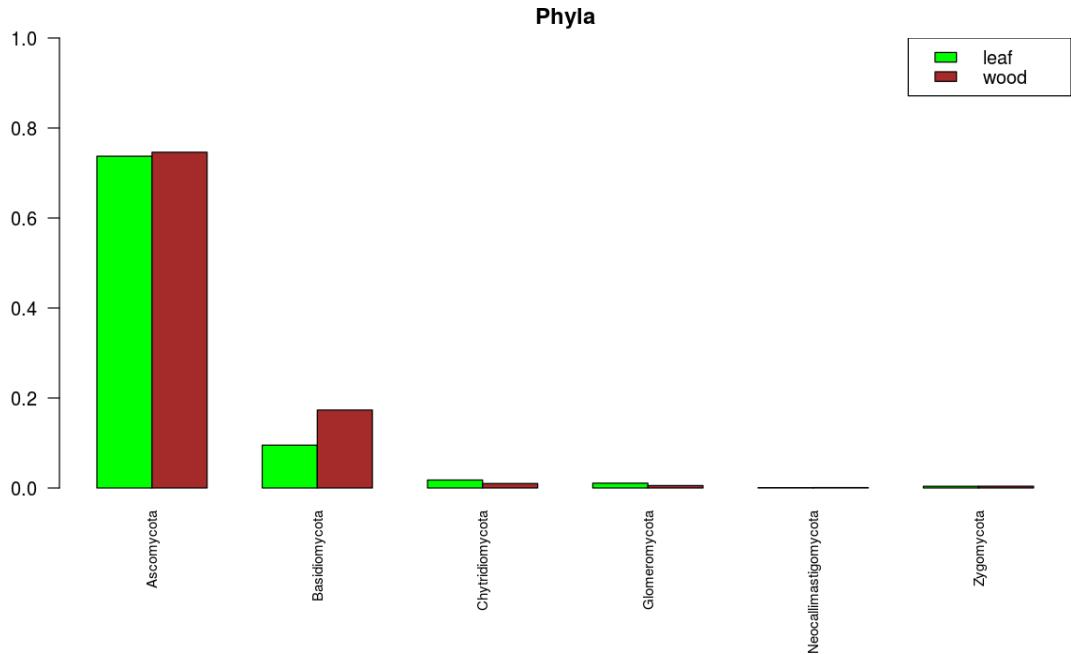
```
In [124]: leafphyplot <- table(tax_table(leafPA)[,"Phylum"])/nrow(otu_table(leafPA))
          woodphyplot <- table(tax_table(woodPA)[,"Phylum"])/nrow(otu_table(woodPA))
          bothphyplot <- as.table(rbind(leafphyplot, woodphyplot))
```

```
In [125]: bothphyplot
```

	Ascomycota	Basidiomycota	Chytridiomycota	Glomeromycota
leafphyplot	0.7373271889	0.0952380952	0.0176651306	0.0107526882
woodphyplot	0.7461728395	0.1733333333	0.0098765432	0.0054320988
	Neocallimastigomycota	Zygomycota		
leafphyplot	0.0007680492	0.0038402458		
woodphyplot	0.0009876543	0.0039506173		

```
In [126]: options(repr.plot.width = 10,repr.plot.height = 6)
par(mar = c(8,3,2,2))
barplot(bothphyplot,
        las=2,
        cex.names=0.75,
        ylim = c(0,1),
        main='Phyla',
        beside=TRUE,
        col = c('green','brown'),
        )

legend('topright',
       fill = c('green','brown'),
       legend= c('leaf','wood')
       )
```



Class level:

```
In [5]: leafclasstable <- table(tax_table(leafPA)[, "Class"])/nrow(otu_table(leafPA))
woodclasstable <- table(tax_table(woodPA)[, "Class"])/nrow(otu_table(woodPA))
leafclassvector <- as.vector(leafclasstable); names(leafclassvector) <- names(leafclasstable)
woodclassvector <- as.vector(woodclasstable); names(woodclassvector) <- names(woodclasstable)
## which classes that are in leaves are not observed in wood? and vice-versa:
notinwoodclassvector <- names(leafclassvector)[!(names(leafclassvector) %in% names(woodclassvector))]
notinleafclassvector <- names(woodclassvector)[!(names(woodclassvector) %in% names(leafclassvector))]
## match up the membership and order of these vectors so we can rbind them
notinwood <- vector(length=length(notinwoodclassvector))
notinwood[] <- 0; names(notinwood) <- notinwoodclassvector
fullwoodclassvector <- c(woodclassvector,notinwood)
notinleaf <- vector(length=length(notinleafclassvector))
notinleaf[] <- 0; names(notinleaf) <- notinleafclassvector
fullleafclassvector <- c(leafclassvector,notinleaf)
fullwoodclassvector <- fullwoodclassvector[names(fullleafclassvector)] ## match order
## put them in a matrix together
bothclass <- rbind(fullleafclassvector, fullwoodclassvector)
```

```
In [6]: bothclass
```

	Agaricomycetes	Arthoniomycetes	Chytridiomycetes	Dothideomycetes	Entomophthoromycetes
fullleafclassvector	0.08602151	0.0007680492	0.002304147	0.2004608	0.0007680492
fullwoodclassvector	0.09679012	0.0004938272	0.004938272	0.2706173	0.0004938272

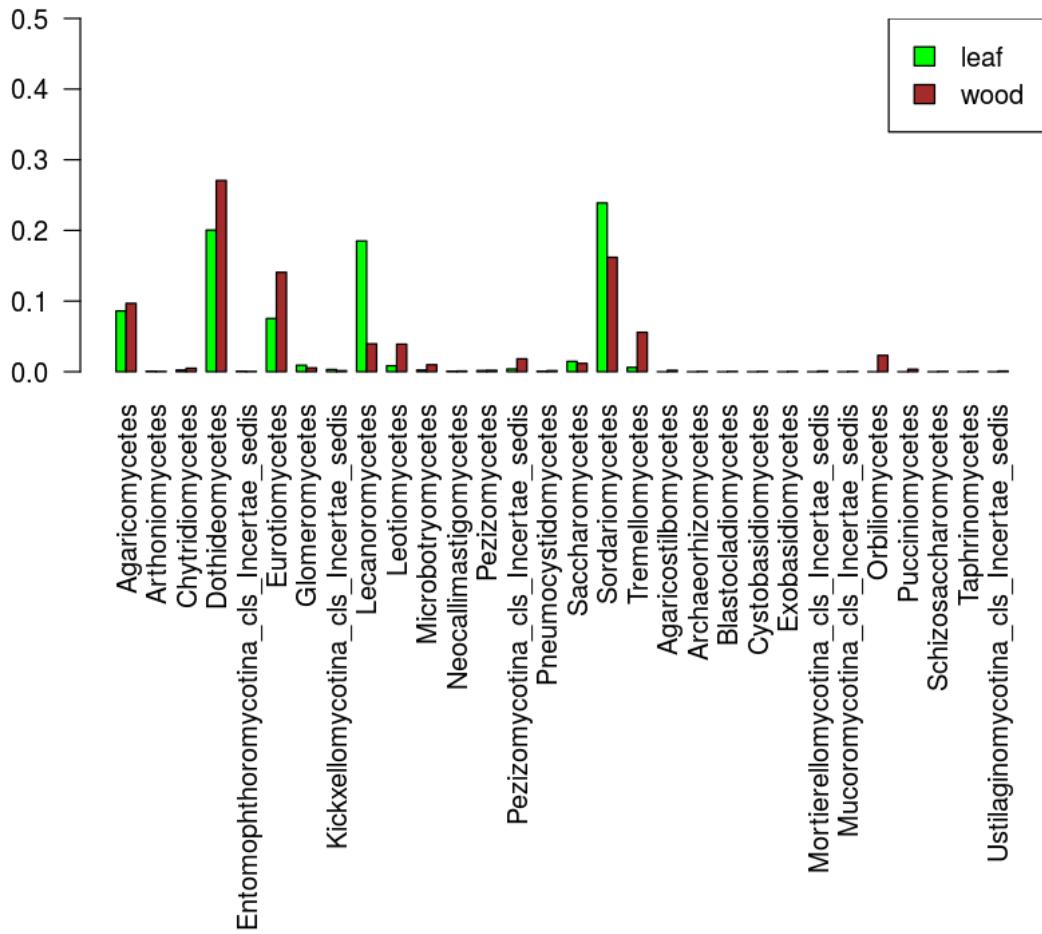
```
In [8]: #svg("Class_bargraph.svg")
par(mar=c(20,3,4,2)) # increase y-axis margin.
```

```

#barplot(bothclasseable,
barplot(bothclass,
  las=2,
  ylim = c(0,0.5),
  #main='Marxist critique: Class differences',
  beside=TRUE,
  col = c('green','brown'),
  )
legend('topright',
  fill = c('green','brown'),
  legend= c('leaf','wood')
  )
#dev.off()

```

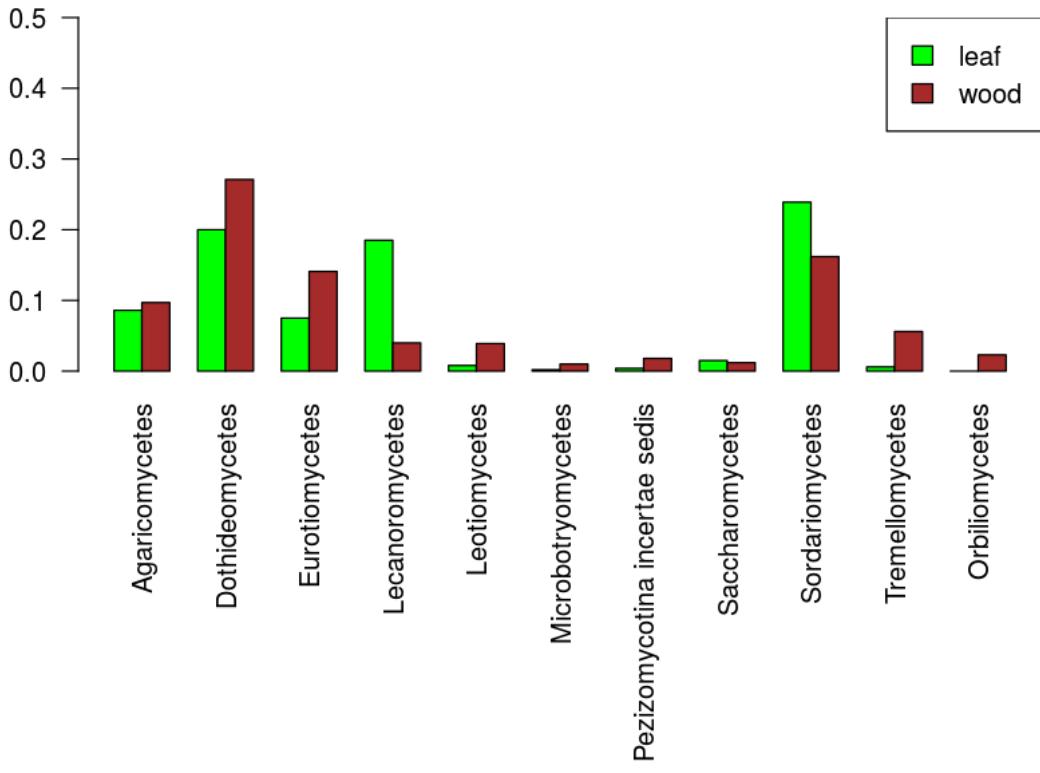
### Marxist critique: Class differences



Okay, but I'm not sure we need these extremely small numbers. They're not really informative, let's simplify.

```
In [14]: ## put them in a matrix together
bothclass <- rbind(fullleafclassvector, fullwoodclassvector)
## round
bothclass <- round(bothclass, digits = 3)
## drop the zeros, and any column that lacks
## a value with at least 1% in either leaf or wood:
bothclass <- bothclass[, colSums(bothclass < 0.01) < 2]
colnames(bothclass)[7] <- "Pezizomycotina incertae sedis"

In [15]: #svg("Class_bargraph.svg")
par(mar=c(20,3,4,2)) # increase y-axis margin.
#barplot(bothclasstable,
barplot(bothclass,
        las=2,
        ylim = c(0,0.5),
        #main='Marxist critique: Class differences',
        beside=TRUE,
        col = c('green','brown'),
        )
legend('topright',
       fill = c('green','brown'),
       legend= c('leaf','wood')
       )
#dev.off()
```



## All-host spatial analysis Mantel tests

```
In [17]: ## wood endophytes
wood95 <- subset_samples(deseq95, Library=="W") ## subset biom to wood
## make our community distance matrix
aa <- t(otu_table(wood95))
aa[aa > 0] <- 1 ## P/A
wood_comdist <- vegdist(aa, method = "bray")
## make our physical distance matrix
cc <- as.matrix(sample_data(wood95)[,c('X', 'Y')])
class(cc) <- "numeric"
physdist <- vegdist(cc, method = "euclidean")
```

```

woodmgram <- mgram(wood_comdist, physdist) ## correlogram object
wood_mant_test <- ecodist::mantel(wood_comdist ~ physdist, nperm = 10000) ## overall test

Warning message in class(X) <- NULL:
Setting class(x) to NULL;    result will no longer be an S4 object

In [18]: wood_mant_test

mantelr      0.10203574049288 pval1      0.0048 pval2      0.9953 pval3      0.0067 llim.2.5\%
0.0652767540073569 ulim.97.5\%                      0.140376492155294

In [19]: ## leaf endophytes
leaf95 <- subset_samples(deseq95, Library=="L")
aa <- t(otu_table(leaf95))
aa[aa > 0] <- 1
leaf_comdist <- vegdist(aa, method = "bray")
cc <- as.matrix(sample_data(leaf95)[,c('X', 'Y')])
class(cc) <- "numeric"
physdist <- vegdist(cc, method = "euclidean")
leafmgram <- mgram(leaf_comdist, physdist)
leaf_mant_test <- ecodist::mantel(leaf_comdist ~ physdist, nperm = 10000)

Warning message in class(X) <- NULL:
Setting class(x) to NULL;    result will no longer be an S4 object

In [20]: leaf_mant_test

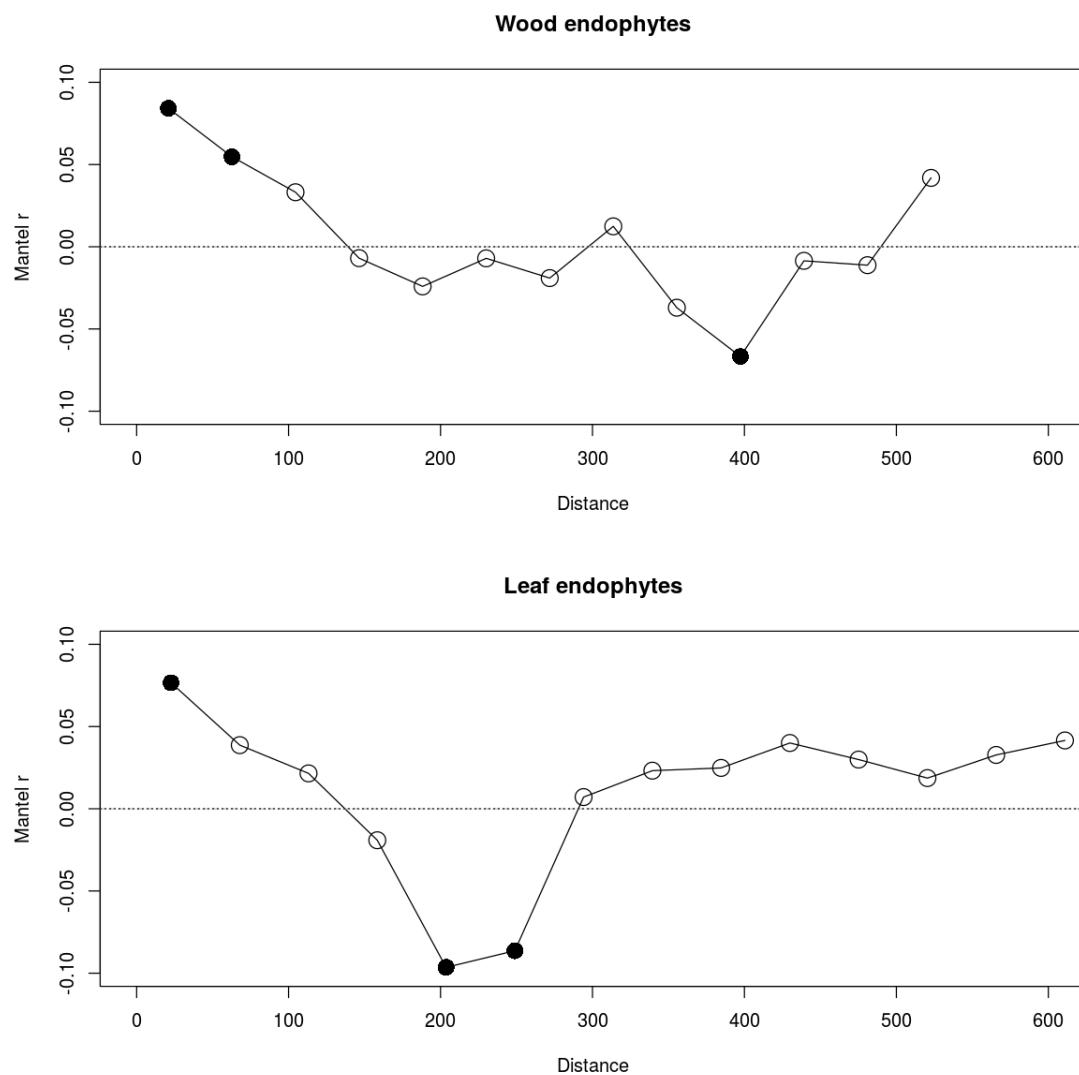
mantelr     -0.00581543149591255 pval1      0.5744 pval2      0.4257 pval3      0.8281 llim.2.5\%
-0.0329528157499163 ulim.97.5\%                      0.0229370938009279

In [22]: svg('allHostMantel.svg')
options(repr.plot.height = 10)
par(mfrow=c(2,1))
plot(woodmgram,
      main="Wood endophytes",
      ylim=c(-.10,.10),
      xlim=c(0,600)
)
abline(h=0,lty=3)
plot(leafmgram,
      main="Leaf endophytes",
      ylim=c(-.10,.10),
      xlim=c(0,600)
)
abline(h=0,lty=3)
dev.off()

```

**png:** 2

```
In [140]: options(repr.plot.height = 10)
par(mfrow=c(2,1))
plot(woodmgram,
      main="Wood endophytes",
      ylim=c(-.10,.10),
      xlim=c(0,600)
    )
abline(h=0,lty=3)
plot(leafmgram,
      main="Leaf endophytes",
      ylim=c(-.10,.10),
      xlim=c(0,600)
    )
abline(h=0,lty=3)
```



Distance-based MEMs

Leaf dbMEMs

See our previous statistical notebook for an explanation and references.

```
In [2]: ## make the leaf phyloseq object
    load('deseq95.rda')
    leaf95 <- subset_samples(deseq95, Library=="L") ## subset biom to leaf
```

```
In [129]: ## get a dataframe of the positions of points:
    leafxy <- sample_data(leaf95)[,c('X', 'Y')]
    rownames(leafxy) <- gsub("leaf.*","", rownames(leafxy) )
    leafxy <- data.frame(leafxy[order(as.numeric(rownames(leafxy))),])
    leafxy$X <- as.numeric(leafxy$X)
    leafxy$Y <- as.numeric(leafxy$Y)
    #save(leafxy, file="leafxy.rda")
```

```
In [3]: #load('leafxy.rda')
```

```
In [4]: ## make spanning tree, get farthest "nearest neighbor":
    ptd <- dist(leafxy)
```

```
    span.ptd <- spantree(ptd)
    dmin <- max(span.ptd$dist)
```

```
dmin
```

```
91.9238815542512
```

```
In [5]: ## truncate our distance matrix using this distance.
```

```
    ptd[ptd > dmin] <- 4*dmin
    ptd.PCoA <- cmdscale(ptd, k=nrow(leafxy)-1, eig = TRUE)
    nb.ev <- length(which(ptd.PCoA$eig > 0.0000001)) ## keep only the positive eigenvalues
    ptd.PCNM <- data.frame(ptd.PCoA$points[1:nrow(leafxy), 1:nb.ev])
```

Warning message in cmdscale(ptd, k = nrow(leafxy) - 1, eig = TRUE):  
only 59 of the first 122 eigenvalues are > 0

Let's see if any of them are useful, by seeing if they explain any variation in our species matrix with an RDA:

```
In [30]: leafcom <- t(otu_table(leaf95)) ## get community matrix out of phyloseq
    rownames(leafcom ) <- gsub("leaf.*","", rownames(leafxy) ) ## clean up names
    leafcom <- data.frame(leafcom[order(as.numeric(rownames(leafcom))),]) ## order
    leafcom[leafcom > 0] <- 1 ## P/A
    leafcom.hel <- decostand(leafcom,'hellinger') ## transform
    leaf.pcnm.rda <- rda(leafcom.hel, ptd.PCNM) ## rda of our leaf community by the PCNMs
    #save(leafcom.hel, file='leafcom.hel.rda') ## need this for later...
```

```
In [14]: #load('leafcom.hel.rda')
```

```
In [31]: leaf.pcnm.sigtest <- anova.cca(leaf.pcnm.rda) ## significance test
```

```
In [135]: leaf.pcnm.sigtest
```

	Df	Variance	F	Pr(>F)
Model	36	0.2715140	1.055838	0.005
Residual	86	0.6143143	NA	NA

```
In [33]: head(ptd.PCNM)
```

X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12	X13	X14	X15	X16	X17	X18	X19	X20	X21	X22	X23	X24	X25	X26	X27	X28	X29	X30	X31	X32	X33	X34	X35	X36
-12.24390	-16.40882	-192.3169	132.6053	-63.30822	6.643841	38.55102	13.856463	45.68720	-9																										
-12.24272	-16.40939	-192.3333	132.6180	-63.31489	6.645941	38.43778	13.826450	45.59288	-9																										
-12.24178	-16.41082	-192.3569	132.6372	-63.32377	6.633360	38.27449	13.769214	45.47777	-9																										
-12.24296	-16.41026	-192.3404	132.6244	-63.31710	6.631260	38.38773	13.799227	45.57208	-9																										
-11.79384	-15.69972	-189.6346	130.7484	-62.45369	10.247207	-15.57217	-3.092479	-11.20009	23																										
-11.79146	-15.70329	-189.6854	130.7873	-62.47368	10.239083	-15.72366	-3.124420	-11.28586	23																										

Lots of them. Do a model selection to find the important ones.

```
In [34]: mod0 <- rda(leafcom.hel ~ 1, ptd.PCNM) ## make a model with no terms  
mod1 <- rda(leafcom.hel ~ ., ptd.PCNM) ## and one with all terms
```

```
In [35]: mod0
```

```
Call: rda(formula = leafcom.hel ~ 1, data = ptd.PCNM)
```

```
Inertia Rank  
Total 0.8858  
Unconstrained 0.8858 122  
Inertia is variance  
  
Eigenvalues for unconstrained axes:  
PC1 PC2 PC3 PC4 PC5 PC6 PC7 PC8  
0.04139 0.03765 0.02669 0.02327 0.02290 0.02085 0.01813 0.01756  
(Showed only 8 of all 122 unconstrained eigenvalues)
```

```
In [36]: mod1
```

```
Call: rda(formula = leafcom.hel ~ X1 + X2 + X3 + X4 + X5 + X6 + X7 + X8  
+ X9 + X10 + X11 + X12 + X13 + X14 + X15 + X16 + X17 + X18 + X19 + X20  
+ X21 + X22 + X23 + X24 + X25 + X26 + X27 + X28 + X29 + X30 + X31 + X32  
+ X33 + X34 + X35 + X36, data = ptd.PCNM)
```

```
Inertia Proportion Rank  
Total 0.8858 1.0000  
Constrained 0.2715 0.3065 36  
Unconstrained 0.6143 0.6935 86
```

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2	RDA3	RDA4	RDA5	RDA6	RDA7	RDA8
0.020544	0.018466	0.014712	0.012691	0.011540	0.010673	0.009853	0.009606
RDA9	RDA10	RDA11	RDA12	RDA13	RDA14	RDA15	RDA16
0.008947	0.008709	0.008507	0.007775	0.007618	0.007235	0.007070	0.006934
RDA17	RDA18	RDA19	RDA20	RDA21	RDA22	RDA23	RDA24
0.006670	0.006610	0.006163	0.006073	0.005990	0.005832	0.005698	0.005446
RDA25	RDA26	RDA27	RDA28	RDA29	RDA30	RDA31	RDA32
0.005182	0.005104	0.004958	0.004845	0.004677	0.004588	0.004335	0.004085
RDA33	RDA34	RDA35	RDA36				
0.004083	0.003795	0.003482	0.003018				

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.030089	0.025824	0.019905	0.019071	0.017080	0.015499	0.014328	0.013585

(Showed only 8 of all 86 unconstrained eigenvalues)

In [37]: `step.res <- ordiR2step(mod0, mod1, perm.max = 1000) ## use them to compare the various`

Step: R2.adj= 0

Call: leafcom.hel ~ 1

	R2.adjusted
<All variables>	1.620984e-02
+ X8	2.620069e-03
+ X3	2.165231e-03
+ X34	2.039849e-03
+ X7	1.917903e-03
+ X9	1.718135e-03
+ X5	1.634330e-03
+ X29	1.608591e-03
+ X4	9.534867e-04
+ X1	7.976764e-04
+ X31	7.431715e-04
+ X36	5.931890e-04
+ X35	5.848153e-04
+ X33	5.778134e-04
+ X2	5.457400e-04
+ X13	4.846803e-04
+ X12	3.433390e-04
+ X14	2.848280e-04
+ X6	2.683434e-04
+ X24	2.244751e-04
+ X16	1.656106e-04

```

+ X26      7.449907e-05
<none>    0.000000e+00
+ X17     -3.125392e-05
+ X28     -6.246177e-05
+ X19     -2.736884e-04
+ X22     -3.101883e-04
+ X10     -3.482506e-04
+ X23     -3.731518e-04
+ X25     -4.337967e-04
+ X18     -4.374499e-04
+ X27     -4.604699e-04
+ X15     -5.472573e-04
+ X20     -5.632760e-04
+ X32     -6.238695e-04
+ X11     -1.052366e-03
+ X21     -1.244011e-03
+ X30     -2.063238e-03

```

	Df	AIC	F	Pr(>F)
--	----	-----	---	--------

```
+ X8  1 -13.251 1.3205  0.016 *
```

---

Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1 1

Step: R2.adj= 0.002620069

Call: leafcom.hel ~ X8

	R2.adjusted
<All variables>	0.0162098442
+ X3	0.0048251774
+ X34	0.0046987499
+ X7	0.0045757885
+ X9	0.0043743554
+ X5	0.0042898523
+ X29	0.0042638980
+ X4	0.0036033349
+ X1	0.0034462263
+ X31	0.0033912671
+ X36	0.0032400348
+ X35	0.0032315913
+ X33	0.0032245310
+ X2	0.0031921904
+ X13	0.0031306219
+ X12	0.0029881027
+ X14	0.0029291041
+ X6	0.0029124822
+ X24	0.0028682483
+ X16	0.0028088932
+ X26	0.0027170224

```

<none>          0.0026200686
+ X17          0.0026103882
+ X28          0.0025789203
+ X19          0.0023659334
+ X22          0.0023291293
+ X10          0.0022907499
+ X23          0.0022656411
+ X25          0.0022044909
+ X18          0.0022008072
+ X27          0.0021775954
+ X15          0.0020900847
+ X20          0.0020739326
+ X32          0.0020128342
+ X11          0.0015807670
+ X21          0.0013875245
+ X30          0.0005614706

```

```

      Df      AIC      F Pr(>F)
+ X3   1 -12.544 1.2681 0.026 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

```

```

Step: R2.adj= 0.004825177
Call: leafcom.hel ~ X8 + X3

```

```

R2.adjusted
<All variables> 0.016209844
+ X34          0.006939857
+ X7           0.006815862
+ X9           0.006612736
+ X5           0.006527523
+ X29          0.006501351
+ X4           0.005835237
+ X1           0.005676808
+ X31          0.005621387
+ X36          0.005468884
+ X35          0.005460369
+ X33          0.005453250
+ X2           0.005420637
+ X13          0.005358551
+ X12          0.005214835
+ X14          0.005155340
+ X6           0.005138579
+ X24          0.005093973
+ X16          0.005034119
+ X26          0.004941476
+ X17          0.004833946
<none>          0.004825177

```

```

+ X28      0.004802214
+ X19      0.004587437
+ X22      0.004550324
+ X10      0.004511622
+ X23      0.004486302
+ X25      0.004424638
+ X18      0.004420923
+ X27      0.004397516
+ X15      0.004309270
+ X20      0.004292982
+ X32      0.004231370
+ X11      0.003795672
+ X21      0.003600806
+ X30      0.002767811

```

```

Df      AIC      F Pr(>F)
+ X34   1 -11.835 1.2555  0.024 *

```

---

```
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

Step: R2.adj= 0.006939857

Call: leafcom.hel ~ X8 + X3 + X34

```

R2.adjusted
<All variables> 0.016209844
+ X7      0.008965333
+ X9      0.008760486
+ X5      0.008674550
+ X29     0.008648156
+ X4      0.007976397
+ X1      0.007816626
+ X31     0.007760735
+ X36     0.007606939
+ X35     0.007598353
+ X33     0.007591173
+ X2      0.007558284
+ X13     0.007495672
+ X12     0.007350737
+ X14     0.007290739
+ X6      0.007273835
+ X24     0.007228851
+ X16     0.007168490
+ X26     0.007075062
+ X17     0.006966621
<none>    0.006939857
+ X28     0.006934619
+ X19     0.006718023
+ X22     0.006680595

```

```

+ X10          0.006641565
+ X23          0.006616030
+ X25          0.006553844
+ X18          0.006550098
+ X27          0.006526493
+ X15          0.006437499
+ X20          0.006421073
+ X32          0.006358939
+ X11          0.005919548
+ X21          0.005723031
+ X30          0.004882976

      Df      AIC      F Pr(>F)
+ X7    1 -11.124 1.2432  0.032 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Step: R2.adj= 0.008965333
Call: leafcom.hel ~ X8 + X3 + X34 + X7

                    R2.adjusted
<All variables> 0.016209844
+ X9            0.010818834
+ X5            0.010732165
+ X29           0.010705545
+ X4             0.010028044
+ X1             0.009866907
+ X31            0.009810539
+ X36            0.009655429
+ X35            0.009646769
+ X33            0.009639527
+ X2             0.009606358
+ X13            0.009543210
+ X12            0.009397037
+ X14            0.009336525
+ X6             0.009319477
+ X24            0.009274109
+ X16            0.009213232
+ X26            0.009119006
+ X17            0.009009637
+ X28            0.008977363
<none>          0.008965333
+ X19            0.008758914
+ X22            0.008721167
+ X10            0.008681803
+ X23            0.008656051
+ X25            0.008593332
+ X18            0.008589554

```

```

+ X27          0.008565747
+ X15          0.008475993
+ X20          0.008459426
+ X32          0.008396761
+ X11          0.007953616
+ X21          0.007755418
+ X30          0.006908183

      Df      AIC      F Pr(>F)
+ X9   1 -10.401 1.2211  0.036 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Step: R2.adj= 0.01081883
Call: leafcom.hel ~ X8 + X3 + X34 + X7 + X9

                    R2.adjusted
<All variables> 0.016209844
+ X5          0.012616876
+ X29          0.012590027
+ X4           0.011906686
+ X1           0.011744159
+ X31          0.011687305
+ X36          0.011530858
+ X35          0.011522123
+ X33          0.011514819
+ X2           0.011481364
+ X13          0.011417672
+ X12          0.011270238
+ X14          0.011209205
+ X6            0.011192010
+ X24          0.011146251
+ X16          0.011084849
+ X26          0.010989811
+ X17          0.010879499
+ X28          0.010846946
<none>        0.010818834
+ X19          0.010626615
+ X22          0.010588542
+ X10          0.010548839
+ X23          0.010522864
+ X25          0.010459606
+ X18          0.010455795
+ X27          0.010431783
+ X15          0.010341254
+ X20          0.010324545
+ X32          0.010261340
+ X11          0.009814374

```

```

+ X21          0.009614468
+ X30          0.008759929

      Df      AIC      F Pr(>F)
+ X5   1 -9.6804 1.2131  0.044 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Step: R2.adj= 0.01261688
Call: leafcom.hel ~ X8 + X3 + X34 + X7 + X9 + X5

```

	R2.adjusted
<All variables>	0.01620984
+ X29	0.01441910
+ X4	0.01372982
+ X1	0.01356588
+ X31	0.01350853
+ X36	0.01335073
+ X35	0.01334192
+ X33	0.01333455
+ X2	0.01330080
+ X13	0.01323656
+ X12	0.01308784
+ X14	0.01302628
+ X6	0.01300893
+ X24	0.01296277
+ X16	0.01290084
+ X26	0.01280497
+ X17	0.01269370
+ X28	0.01266087
<none>	0.01261688
+ X19	0.01243862
+ X22	0.01240022
+ X10	0.01236017
+ X23	0.01233397
+ X25	0.01227016
+ X18	0.01226631
+ X27	0.01224209
+ X15	0.01215078
+ X20	0.01213392
+ X32	0.01207017
+ X11	0.01161932
+ X21	0.01141767
+ X30	0.01055570

```

      Df      AIC      F Pr(>F)
+ X29   1 -8.97 1.2121  0.02 *
---

```

```

Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Step: R2.adj= 0.0144191
Call: leafcom.hel ~ X8 + X3 + X34 + X7 + X9 + X5 + X29

          R2.adjusted
<All variables>  0.01620984
+ X4            0.01555762
+ X1            0.01539224
+ X31           0.01533439
+ X36           0.01517520
+ X35           0.01516631
+ X33           0.01515888
+ X2            0.01512484
+ X13           0.01506003
+ X12           0.01491001
+ X14           0.01484791
+ X6            0.01483041
+ X24           0.01478385
+ X16           0.01472137
+ X26           0.01462466
+ X17           0.01451242
+ X28           0.01447929
<none>          0.01441910
+ X19           0.01425509
+ X22           0.01421635
+ X10           0.01417595
+ X23           0.01414952
+ X25           0.01408515
+ X18           0.01408128
+ X27           0.01405684
+ X15           0.01396473
+ X20           0.01394773
+ X32           0.01388341
+ X11           0.01342860
+ X21           0.01322519
+ X30           0.01235566

      Df      AIC      F Pr(>F)
+ X4    1 -8.1865 1.133   0.12

```

```

In [38]: attributes(step.res$terms)$term.labels
1. 'X8' 2. 'X3' 3. 'X34' 4. 'X7' 5. 'X9' 6. 'X5' 7. 'X29'

In [41]: sigPCNM <- attributes(step.res$terms)$term.labels

```

```

sigPCNM <- sigPCNM[c(2,6,4,1,5,7,3)]  

In [42]: sigPCNM  

1. 'X3' 2. 'X5' 3. 'X7' 4. 'X8' 5. 'X9' 6. 'X29' 7. 'X34'  

In [7]: sigPCNM <- c('X3', 'X5', 'X7', 'X8', 'X9', 'X29', 'X34')  

In [10]: leafPCNM <- ptd.PCNM[,sigPCNM]  

         save(leafPCNM, file='leafPCNM.rda')  

In [11]: head(leafPCNM)

```

	X3	X5	X7	X8	X9	X29	X34
-192.3169	-63.30822	38.55102	13.856463	45.68720	-0.08749394	1.0041332	
-192.3333	-63.31489	38.43778	13.826450	45.59288	-0.08574080	1.1295285	
-192.3569	-63.32377	38.27449	13.769214	45.47777	-0.09419784	1.0706720	
-192.3404	-63.31710	38.38773	13.799227	45.57208	-0.09595098	0.9452767	
-189.6346	-62.45369	-15.57217	-3.092479	-11.20009	0.06989570	-0.8895464	
-189.6854	-62.47368	-15.72366	-3.124420	-11.28586	0.07476702	-0.5075401	

These are the important ones. But how important?

```
In [15]: leaf.pcnm.rda2 <- rda(leafcom.hel ~ ., leafPCNM)
```

```
In [16]: leaf.pcnm.rda2
```

```
Call: rda(formula = leafcom.hel ~ X3 + X5 + X7 + X8 + X9 + X29 + X34,
data = leafPCNM)
```

	Inertia	Proportion	Rank
Total	0.88583	1.00000	
Constrained	0.06287	0.07097	7
Unconstrained	0.82296	0.92903	115
Inertia is variance			

Eigenvalues for constrained axes:

RDA1	RDA2	RDA3	RDA4	RDA5	RDA6	RDA7
0.012774	0.011118	0.009905	0.008368	0.007642	0.006620	0.006441

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.03844	0.03281	0.02383	0.02233	0.02065	0.01928	0.01762	0.01653

(Showed only 8 of all 115 unconstrained eigenvalues)

Let's look at the variance explained by each ecologically important axis:

```
In [88]: evs <- c(0.012774, 0.011118, 0.009905, 0.008368, 0.007642, 0.006620, 0.006441)
      names(evs) <- sigPCNM
      cbind(evs, round(evs/0.88583, digits=3))
```

	evs	
X3	0.012774	0.014
X5	0.011118	0.013
X7	0.009905	0.011
X8	0.008368	0.009
X9	0.007642	0.009
X29	0.006620	0.007
X34	0.006441	0.007

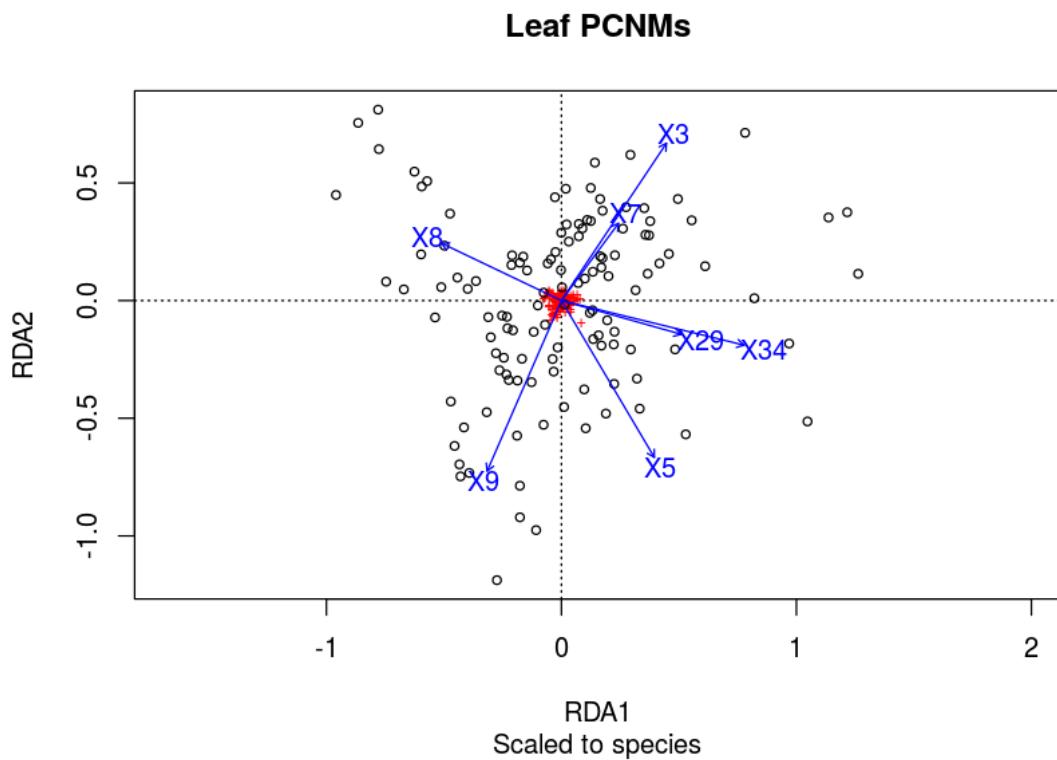
```
In [50]: anova.cca(leaf.pcnm.rda2)
```

	Df	Variance	F	Pr(>F)
Model	7	0.0628662	1.254981	0.001
Residual	115	0.8229621	NA	NA

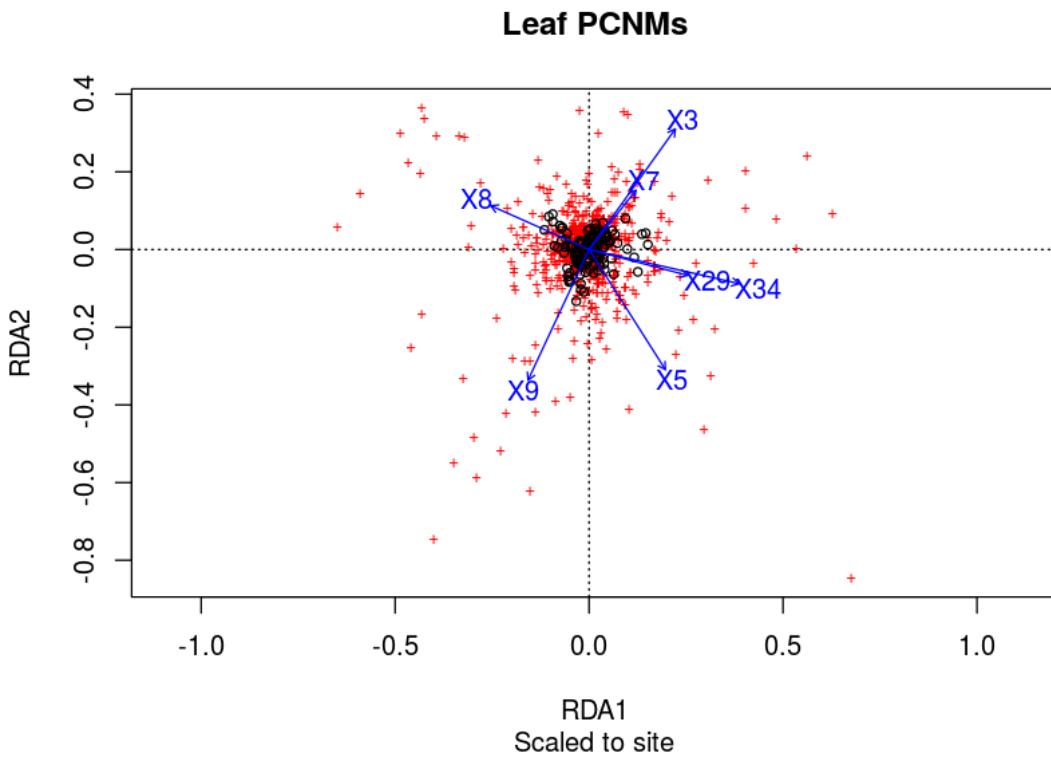
```
In [ ]: axes.test <- anova.cca(leaf.pcnm.rda2, by='axis') ## takes a minute
```

```
In [ ]: axes.test
```

```
In [17]: options(repr.plot.height = 5)
        plot(leaf.pcnm.rda2,
              display = c('sp', 'wa', 'bp'),
              scaling = 2,
              sub='Scaled to species',
              main='Leaf PCNMs'
        )
```



```
In [18]: options(repr.plot.height = 5)
plot(leaf.pcnm.rda2,
      display = c('sp', 'wa', 'bp'),
      scaling = 1,
      sub='Scaled to site',
      main='Leaf PCNMs'
)
```



Here we'll use a function that modifies the s.value function. It's in the supporting material for Borcard's book, [source code available here](#)

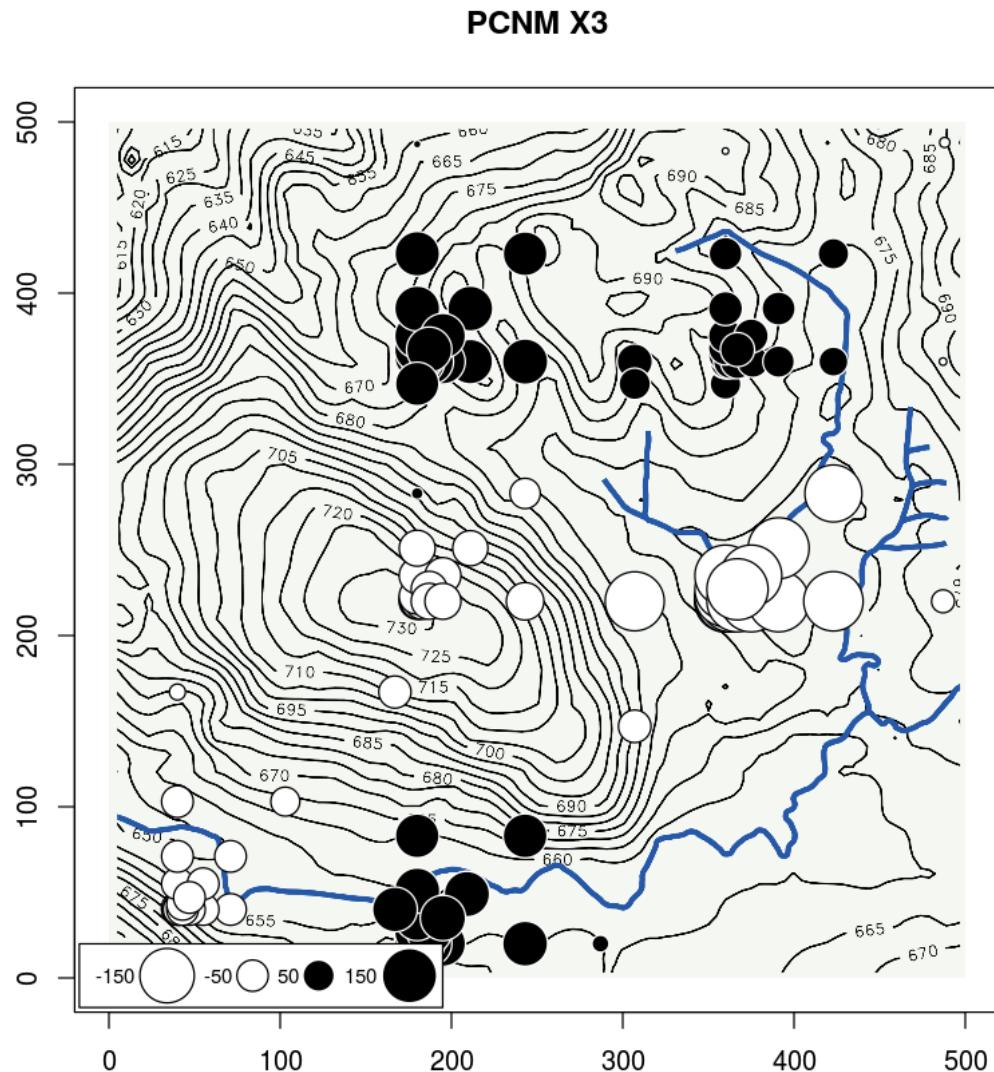
```
In [19]: source("/home/daniel/Documents/taiwan/taiwan_combined_stats/CSrev/NEwR-2ed_code_data/NEwR.R")
```

Make a function for viewing these.

```
In [80]: mapP <- function(PCNM, P, bkg){
  require('png')
  topo <- readPNG(bkg) ## load scanned image of Fushan Map
  plot(1, type='n',
    xlim=c(0,500),
    ylim=c(0,500),
    xlab = '',
    ylab = '',
    main = paste('PCNM', colnames(PCNM)[P], sep = ' '))
  ## blank plot
  rasterImage(topo, 0, 0, 500, 500)
  sr.value(dfxy=leafxy,
  z=leafPCNM[,P],
  # clegend = 0, ## gets rid of legends, they can get in the way
  add.plot = TRUE,
)
}
```

```
In [12]: #save(mapP, file='mapPleaf.R')
```

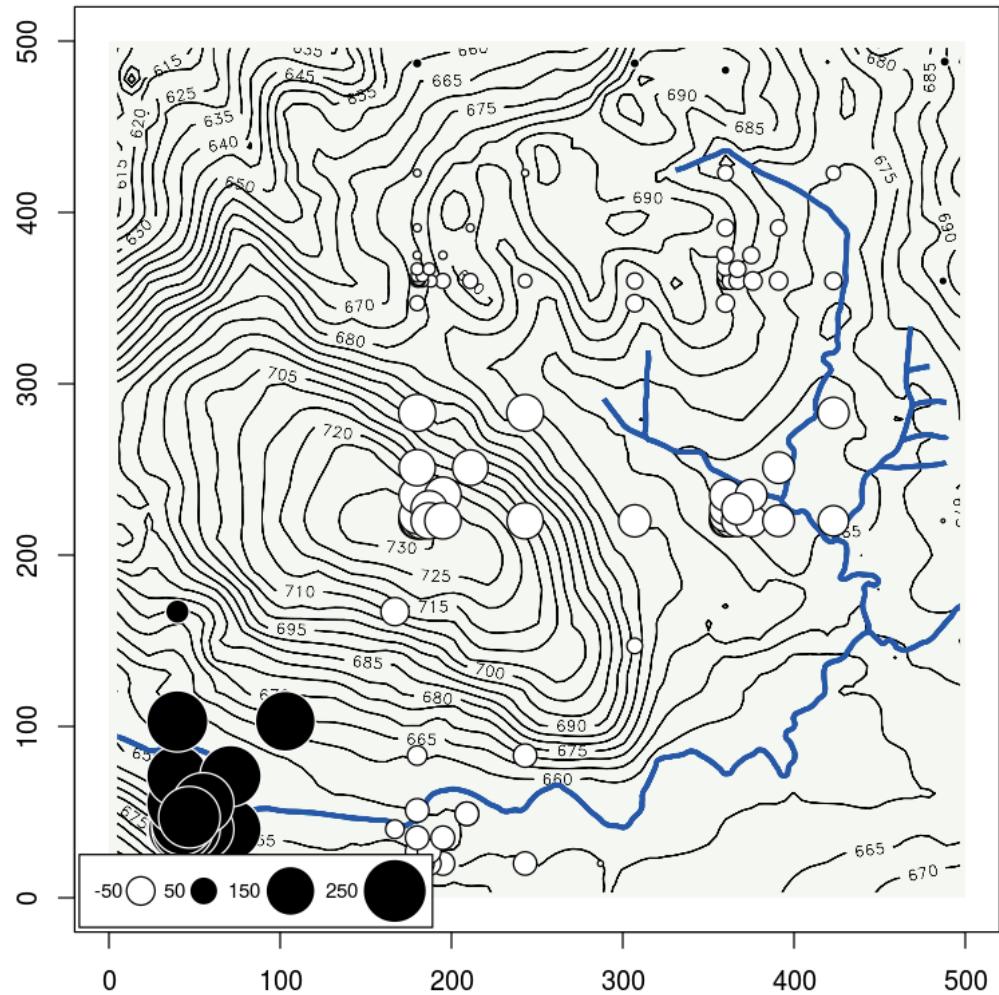
```
In [54]: options(repr.plot.height = 8)
par(pty="s")
mapP(leafPCNM,1,'/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png')
```



```
In [55]: options(repr.plot.height = 8)
```

```
par(pty="s")
mapP(leafPCNM,2,'/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png')
```

## PCNM X5



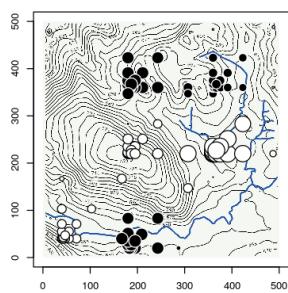
```
In [96]: ## make svgs
  par(pty="s")
  par(mfrow=c(1,1))
  for (i in 1:ncol(leafPCNM)){
    svg(paste0('allHostLeafPCNMs','_',i,'.svg'))
    mapP(leafPCNM,i,'/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png')
    dev.off()
  }
```

```
In [88]: evs <- c(0.012774, 0.011118, 0.009905, 0.008368, 0.007642, 0.006620, 0.006441)
      names(evs) <- sigPCNM
      cbind(evs, round(evs/0.88583, digits=3))
```

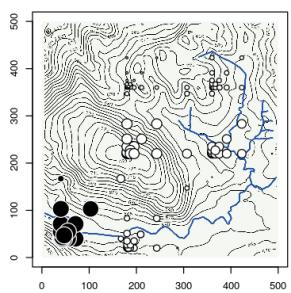
	evs	
X3	0.012774	0.014
X5	0.011118	0.013
X7	0.009905	0.011
X8	0.008368	0.009
X9	0.007642	0.009
X29	0.006620	0.007
X34	0.006441	0.007

```
In [79]: #svg('allHostLeafPCNMs.svg')
options(repr.plot.height = 20)
par(pty="s")
par(mfrow=c(4,2))
for (i in 1:ncol(leafPCNM)){mapP(leafPCNM,i, '/home/daniel/Documents/taiwan/taiwan_combi
par(mfrow=c(1,1))
```

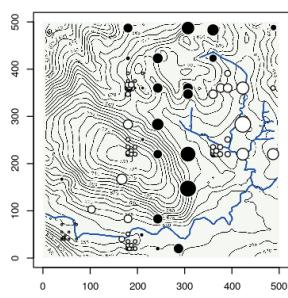
PCNM X3



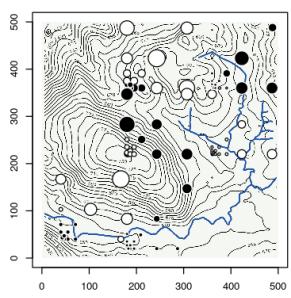
PCNM X5



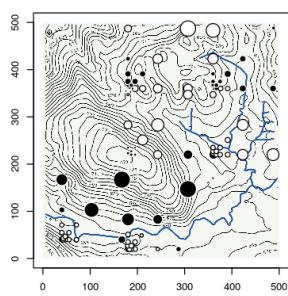
PCNM X7



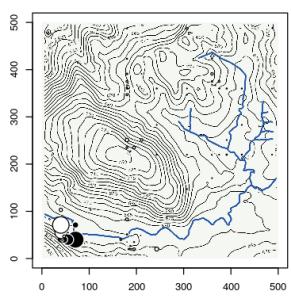
PCNM X8



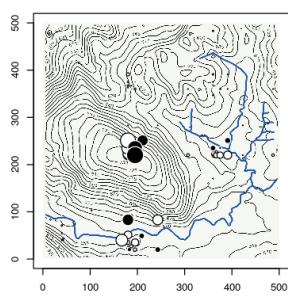
PCNM X9



PCNM X29



PCNM X34



It seems like our hill top PCNM has decomposed into several separate PCNM vectors. If we were to follow Borcard 2011's example, I think we'd try reordering the vectors that are acting on similar scales. If we did a PCA on these we might find our old hilltop vector. But that is a different direction than my original analysis, seems a little like fishing for results. Let's stick to repeating what has been done.

Check environmental correlations

```
In [160]: leaf.env <- sample_data(leaf95)[,c('Forest_Type', 'vegcom')] ## fix weird rownames and
           rownames(leaf.env) <- gsub("leaf.*","",rownames(leaf.env))

In [165]: leaf.env <- data.frame(leaf.env[order(as.numeric(rownames(leaf.env))),])
           leaf.env$Forest_Type <- as.factor(leaf.env$Forest_Type)
           leaf.env$vegcom <- as.factor(leaf.env$vegcom)

           all(rownames(leaf.env) == rownames(leafPCNM)) ## check order of rows match
           #save(leaf.env, file="leaf.env.rda")
```

TRUE

```
In [23]: #load('leaf.env.rda')
```

```
In [24]: head(leafPCNM)
```

X3	X5	X7	X8	X9	X29	X34
-192.3169	-63.30822	38.55102	13.856463	45.68720	-0.08749394	1.0041332
-192.3333	-63.31489	38.43778	13.826450	45.59288	-0.08574080	1.1295285
-192.3569	-63.32377	38.27449	13.769214	45.47777	-0.09419784	1.0706720
-192.3404	-63.31710	38.38773	13.799227	45.57208	-0.09595098	0.9452767
-189.6346	-62.45369	-15.57217	-3.092479	-11.20009	0.06989570	-0.8895464
-189.6854	-62.47368	-15.72366	-3.124420	-11.28586	0.07476702	-0.5075401

```
In [39]: leafPCNM.X3 <- lm(leafPCNM[,1] ~ ., data=leaf.env)
           leafPCNM.X5 <- lm(leafPCNM[,2] ~ ., data=leaf.env)
           leafPCNM.X7 <- lm(leafPCNM[,3] ~ ., data=leaf.env)
           leafPCNM.X8 <- lm(leafPCNM[,4] ~ ., data=leaf.env)
           leafPCNM.X9 <- lm(leafPCNM[,5] ~ ., data=leaf.env)
           leafPCNM.X29 <- lm(leafPCNM[,6] ~ ., data=leaf.env)
           leafPCNM.X34 <- lm(leafPCNM[,7] ~ ., data=leaf.env)
```

```
In [40]: summary(leafPCNM.X3)
```

Call:

```
lm(formula = leafPCNM[, 1] ~ ., data = leaf.env)
```

Residuals:

Min	1Q	Median	3Q	Max
-----	----	--------	----	-----

```
-163.639 -58.465 -0.024 84.381 139.293
```

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-66.020	63.465	-1.040	0.300
Forest_Type2	8.471	69.280	0.122	0.903
Forest_Type3	24.829	76.363	0.325	0.746
Forest_Type4	-66.397	98.734	-0.672	0.503
Forest_Type5	60.173	101.867	0.591	0.556
Forest_Type6	76.013	95.696	0.794	0.429
Forest_Type7	-33.133	94.126	-0.352	0.725
vegcom2	70.436	69.198	1.018	0.311
vegcom3	112.875	69.356	1.627	0.106
vegcom4	16.416	94.226	0.174	0.862

Residual standard error: 89.75 on 113 degrees of freedom

Multiple R-squared: 0.2679, Adjusted R-squared: 0.2096

F-statistic: 4.594 on 9 and 113 DF, p-value: 3.52e-05

In [41]: `summary(leafPCNM.X5)`

Call:

```
lm(formula = leafPCNM[, 2] ~ ., data = leaf.env)
```

Residuals:

Min	1Q	Median	3Q	Max
-156.813	-38.032	0.036	23.493	259.857

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-82.136	54.916	-1.496	0.1375
Forest_Type2	4.088	59.947	0.068	0.9457
Forest_Type3	26.852	66.077	0.406	0.6852
Forest_Type4	87.015	85.434	1.019	0.3106
Forest_Type5	20.561	88.145	0.233	0.8160
Forest_Type6	-4.559	82.806	-0.055	0.9562
Forest_Type7	23.193	81.446	0.285	0.7763
vegcom2	33.663	59.877	0.562	0.5751
vegcom3	131.645	60.013	2.194	0.0303 *
vegcom4	192.922	81.534	2.366	0.0197 *

---

Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1 1

Residual standard error: 77.66 on 113 degrees of freedom

Multiple R-squared: 0.3796, Adjusted R-squared: 0.3302

F-statistic: 7.681 on 9 and 113 DF, p-value: 9.642e-09

In [42]: `summary(leafPCNM.X7)`

Call:

`lm(formula = leafPCNM[, 3] ~ ., data = leaf.env)`

Residuals:

Min	1Q	Median	3Q	Max
-155.002	-10.425	-5.251	3.292	163.341

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-11.149	29.367	-0.380	0.705
Forest_Type2	6.033	32.058	0.188	0.851
Forest_Type3	58.521	35.336	1.656	0.100
Forest_Type4	-5.073	45.688	-0.111	0.912
Forest_Type5	22.347	47.137	0.474	0.636
Forest_Type6	-24.648	44.282	-0.557	0.579
Forest_Type7	-15.676	43.555	-0.360	0.720
vegcom2	24.202	32.020	0.756	0.451
vegcom3	22.431	32.093	0.699	0.486
vegcom4	43.617	43.602	1.000	0.319

Residual standard error: 41.53 on 113 degrees of freedom

Multiple R-squared: 0.1148, Adjusted R-squared: 0.04426

F-statistic: 1.628 on 9 and 113 DF, p-value: 0.1156

In [43]: `summary(leafPCNM.X8)`

Call:

`lm(formula = leafPCNM[, 4] ~ ., data = leaf.env)`

Residuals:

Min	1Q	Median	3Q	Max
-158.553	-2.282	3.794	8.804	129.318

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-12.722	28.085	-0.453	0.651446
Forest_Type2	-3.221	30.659	-0.105	0.916516
Forest_Type3	19.080	33.793	0.565	0.573449
Forest_Type4	130.103	43.693	2.978	0.003555 **

```

Forest_Type5 104.186      45.080   2.311 0.022638 *
Forest_Type6 123.447      42.349   2.915 0.004290 **
Forest_Type7 115.246      41.654   2.767 0.006616 **
vegcom2     -103.666      30.623   -3.385 0.000979 ***
vegcom3     -97.530       30.692   -3.178 0.001915 **
vegcom4     -82.874       41.698   -1.987 0.049289 *
---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 39.72 on 113 degrees of freedom
Multiple R-squared: 0.1532, Adjusted R-squared: 0.08573
F-statistic: 2.271 on 9 and 113 DF, p-value: 0.02232

```

In [44]: `summary(leafPCNM.X9)`

```

Call:
lm(formula = leafPCNM[, 5] ~ ., data = leaf.env)

Residuals:
```

Min	1Q	Median	3Q	Max
-125.599	-11.637	0.032	14.946	122.657

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	6.1097	23.8001	0.257	0.79787
Forest_Type2	-2.4392	25.9807	-0.094	0.92537
Forest_Type3	82.8740	28.6370	2.894	0.00457 **
Forest_Type4	-5.8950	37.0265	-0.159	0.87379
Forest_Type5	0.9476	38.2013	0.025	0.98025
Forest_Type6	35.3143	35.8873	0.984	0.32720
Forest_Type7	25.0797	35.2982	0.711	0.47885
vegcom2	-34.8875	25.9501	-1.344	0.18151
vegcom3	-31.4230	26.0092	-1.208	0.22951
vegcom4	-36.8152	35.3360	-1.042	0.29970

```

---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```

Residual standard error: 33.66 on 113 degrees of freedom
Multiple R-squared: 0.2828, Adjusted R-squared: 0.2257
F-statistic: 4.951 on 9 and 113 DF, p-value: 1.315e-05
```

In [45]: `summary(leafPCNM.X29)`

```
Call:  
lm(formula = leafPCNM[, 6] ~ ., data = leaf.env)
```

Residuals:

Min	1Q	Median	3Q	Max
-20.6469	-0.3274	-0.1813	0.3202	22.4760

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	0.3151	2.3223	0.136	0.892
Forest_Type2	-0.3020	2.5351	-0.119	0.905
Forest_Type3	-0.0890	2.7942	-0.032	0.975
Forest_Type4	-0.3599	3.6128	-0.100	0.921
Forest_Type5	-0.7320	3.7275	-0.196	0.845
Forest_Type6	-0.8302	3.5017	-0.237	0.813
Forest_Type7	-0.8161	3.4442	-0.237	0.813
vegcom2	0.7464	2.5321	0.295	0.769
vegcom3	-0.2084	2.5378	-0.082	0.935
vegcom4	1.9315	3.4479	0.560	0.576

Residual standard error: 3.284 on 113 degrees of freedom  
Multiple R-squared: 0.02191, Adjusted R-squared: -0.05599  
F-statistic: 0.2812 on 9 and 113 DF, p-value: 0.9787

```
In [46]: summary(leafPCNM.X34)
```

```
Call:  
lm(formula = leafPCNM[, 7] ~ ., data = leaf.env)
```

Residuals:

Min	1Q	Median	3Q	Max
-8.2077	-0.2429	0.0475	0.3386	10.9976

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	0.16925	1.49630	0.113	0.910
Forest_Type2	0.11849	1.63340	0.073	0.942
Forest_Type3	-0.92724	1.80040	-0.515	0.608
Forest_Type4	-0.88119	2.32785	-0.379	0.706
Forest_Type5	-0.32198	2.40170	-0.134	0.894
Forest_Type6	-0.26347	2.25622	-0.117	0.907
Forest_Type7	-0.33554	2.21919	-0.151	0.880
vegcom2	0.09221	1.63147	0.057	0.955
vegcom3	0.37872	1.63519	0.232	0.817
vegcom4	0.15983	2.22156	0.072	0.943

```

Residual standard error: 2.116 on 113 degrees of freedom
Multiple R-squared:  0.01447, Adjusted R-squared: -0.06402
F-statistic: 0.1844 on 9 and 113 DF,  p-value: 0.9954

```

The only real evidence of environmental correlation are X5, X8 and X9, which seem to be related to the microtopography composite variable and the vegetation zones. X5 is correlated with the presence of vegetative community types 3 and 4, with a fair amount of variance explained ( $R^2 = .33$ ). This seems like the most important result here.

X8 looks like it compares the southern hillsides with the rest of the plot, but is poorly predicted by this ( $R^2 = .085$ ). X9 looks like a comparison of the northern hillside with the rest of the plots. X9 is much better explained ( $R^2 = 0.22$ ), but we can only track it to one type of topography, type 3. This is the steepest habitat, present on the central hill, mostly.

Not sure what happened to the general hilltop PCNM that was important when we ran this analysis last time. It may have decomposed into these two (X8 and X9). We seem to have unintentionally imposed a stricter cleanup and perhaps lost some patterns observed before. But better to be cautious, leave them off. At least one reviewer seem to think the hilltop argument was overemphasized, anyway.

We can look at X5, X8 and X9 again on a map of the microtopographic and vegetation variables:

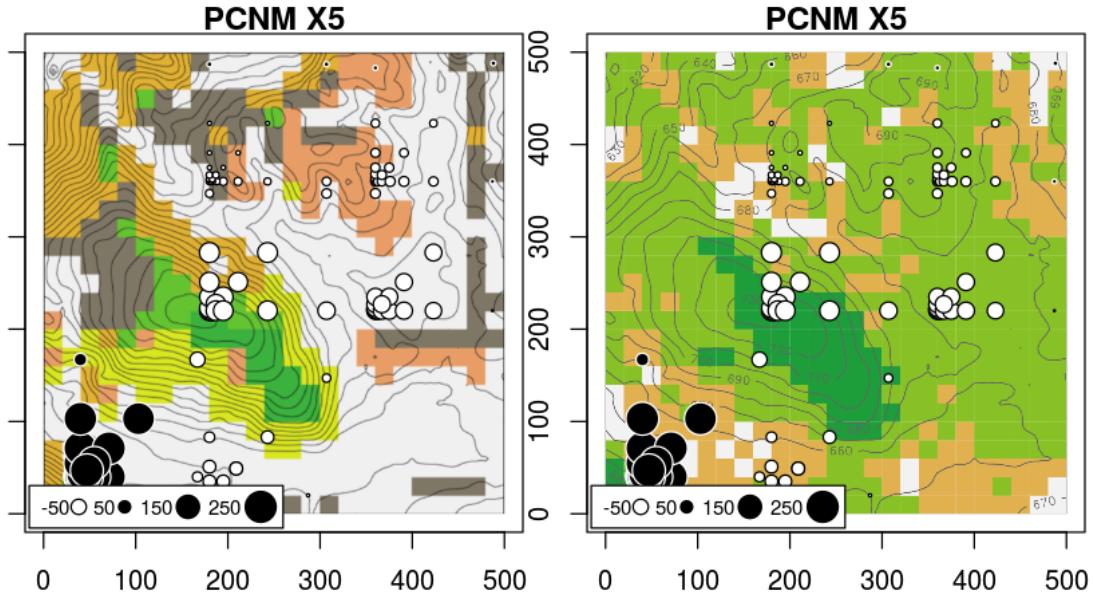
```
In [178]: head(leafPCNM)
```

X3	X7	X8	X9	X34
192.3169	38.55102	13.856463	45.68720	1.0041332
192.3333	38.43778	13.826450	45.59288	1.1295285
192.3569	38.27449	13.769214	45.47777	1.0706720
192.3404	38.38773	13.799227	45.57208	0.9452767
189.6346	-15.57217	-3.092479	-11.20009	-0.8895464
189.6854	-15.72366	-3.124420	-11.28586	-0.5075401

```
In [27]: load('leafPCNM.rda')
load('mapPleaf.R')
load("leafxy.rda")
source("/home/daniel/Documents/taiwan/taiwan_combined_stats/CSrev/NEWR-2ed_code_data/NE
```

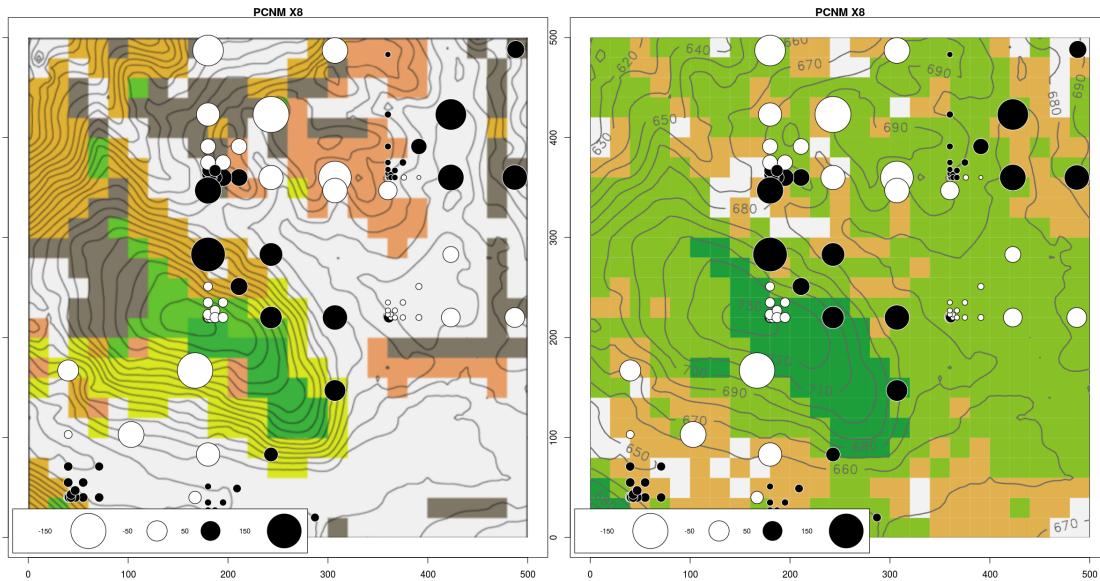
PCNM X5:

```
In [58]: options(repr.plot.height=5) #, repr.plot.height=3)
par(mar=c(1,1,1,1))
par(pty="s")
par(mfrow=c(1,2))
mapP(leafPCNM,2,'/home/daniel/Documents/taiwan/taiwan_combined_stats/forestmap.png')
mapP(leafPCNM,2,'/home/daniel/Documents/taiwan/taiwan_combined_stats/fushveg.png')
par(mfrow=c(1,1))
```



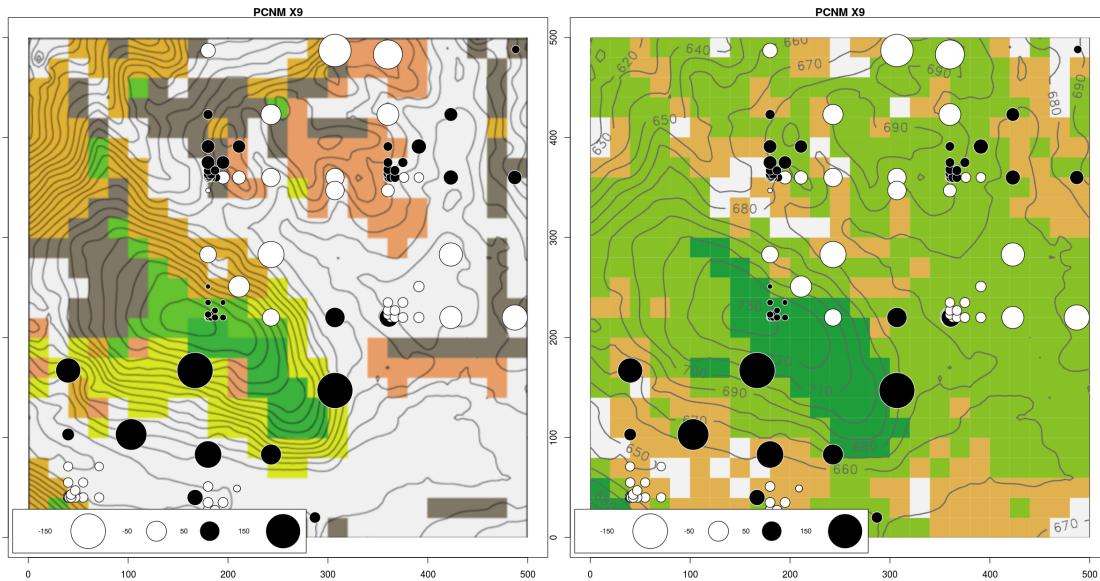
PCNM X8:

```
In [54]: options(repr.plot.height=20) #, repr.plot.height=3)
par(mar=c(1,1,1,1))
par(pty="s")
par(mfrow=c(1,2))
mapP(leafPCNM,3,'/home/daniel/Documents/taiwan/taiwan_combined_stats/forestmap.png')
mapP(leafPCNM,3,'/home/daniel/Documents/taiwan/taiwan_combined_stats/fushveg.png')
par(mfrow=c(1,1))
```



## PCNM X9

```
In [52]: options(repr.plot.height=20) #, repr.plot.height=3)
par(mar=c(1,1,1,1))
par(pty="s")
par(mfrow=c(1,2))
mapP(leafPCNM,4,'/home/daniel/Documents/taiwan/taiwan_combined_stats/forestmap.png')
mapP(leafPCNM,4,'/home/daniel/Documents/taiwan/taiwan_combined_stats/fushveg.png')
par(mfrow=c(1,1))
```



## Wood dbMEMs

```
In [82]: ## generate possible PCNMs for points where wood endophytes were sampled:
load('deseq95.rda')
wood95 <- subset_samples(deseq95, Library=="W")
woodxy <- sample_data(wood95)[,c('X', 'Y')]
rownames(woodxy) <- gsub("w.*","", rownames(woodxy) )
woodxy$X <- as.numeric(woodxy$X)
woodxy$Y <- as.numeric(woodxy$Y)
#save(woodxy, file="woodxy.rda")
```

```
In [89]: ptd <- dist(woodxy)
span.ptd <- spantree(ptd)
dmin <- max(span.ptd$dist) ## 90.51 m
```

```

ptd[ptd > dmin] <- 4*dmin
ptd.PCoA <- cmdscale(ptd, k=nrow(woodxy)-1, eig = TRUE)
nb.ev <- length(which(ptd.PCoA$eig > 0.0000001))
ptd.PCNM <- data.frame(ptd.PCoA$points[1:nrow(woodxy), 1:nb.ev])

```

Warning message in class(X) <- NULL:

Setting class(x) to NULL; result will no longer be an S4 object  
 Warning message in cmdscale(ptd  
 only 51 of the first 90 eigenvalues are > 0

In [90]: *## get our community matrix, check for redundancy with our new PCNMs.*

```

woodcom <- t(otu_table(wood95))
rownames(woodcom) <- gsub("w.*","", rownames(woodcom) )
woodcom <- data.frame(woodcom)
woodcom[woodcom > 0] <- 1
woodcom.hel <- decostand(woodcom,'hellinger')
wood.pcnm.rda <- rda(woodcom.hel, ptd.PCNM)
wood.pcnm.sigtest <- anova.cca(wood.pcnm.rda)
wood.pcnm.sigtest

```

	Df	Variance	F	Pr(>F)
Model	31	0.3126333	1.067526	0.001
Residual	59	0.5573744	NA	NA

In [91]: `save(woodcom.hel, file='woodcom.hel.rda')`

In [92]: *## select important components of the model:*

```

mod0 <- rda(woodcom.hel ~ 1, ptd.PCNM)
mod1 <- rda(woodcom.hel ~ ., ptd.PCNM)
step.res <- ordiR2step(mod0, mod1, perm.max = 1000) ## 4 pcns
step.res$anova

```

Step: R2.adj= 0

Call: woodcom.hel ~ 1

	R2.adjusted
<All variables>	2.273028e-02
+ X3	7.446086e-03
+ X1	6.634900e-03
+ X20	2.333609e-03
+ X5	2.009092e-03
+ X30	1.448755e-03
+ X29	1.381123e-03
+ X18	1.343061e-03
+ X4	1.233672e-03
+ X2	1.120416e-03
+ X27	8.618764e-04
+ X24	5.939652e-04
+ X26	4.844883e-04
+ X22	4.443681e-04

```

+ X17      3.821384e-04
+ X16      1.433522e-04
<none>    0.000000e+00
+ X6      -9.384242e-05
+ X12     -3.149634e-04
+ X25     -3.337257e-04
+ X28     -3.491988e-04
+ X21     -4.286406e-04
+ X7      -4.287189e-04
+ X13     -4.399449e-04
+ X19     -5.682180e-04
+ X15     -8.672504e-04
+ X10     -9.373336e-04
+ X11     -1.088913e-03
+ X14     -1.183240e-03
+ X31     -1.212176e-03
+ X23     -1.221067e-03
+ X9      -1.345341e-03
+ X8      -1.979941e-03

```

```

Df      AIC      F Pr(>F)
+ X3   1 -11.374 1.6752 0.002 **
---
```

```
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1
```

```

Step: R2.adj= 0.007446086
Call: woodcom.hel ~ X3
```

```

R2.adjusted
<All variables> 0.022730285
+ X1      0.014240998
+ X20     0.009890828
+ X5      0.009562624
+ X30     0.008995919
+ X29     0.008927519
+ X18     0.008889024
+ X4      0.008778392
+ X2      0.008663849
+ X27     0.008402371
+ X24     0.008131416
+ X26     0.008020695
+ X22     0.007980119
+ X17     0.007917182
+ X16     0.007675682
<none>   0.007446086
+ X6      0.007435792
+ X12     0.007212159
+ X25     0.007193183
```

```

+ X28      0.007177534
+ X21      0.007097189
+ X7       0.007097110
+ X13      0.007085757
+ X19      0.006956026
+ X15      0.006653595
+ X10      0.006582716
+ X11      0.006429414
+ X14      0.006334016
+ X31      0.006304750
+ X23      0.006295758
+ X9       0.006170072
+ X8       0.005528261

```

```

Df      AIC      F Pr(>F)
+ X1    1 -11.028 1.6135 0.002 **

---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

```

```

Step: R2.adj= 0.014241
Call: woodcom.hel ~ X3 + X1

```

```

R2.adjusted
<All variables> 0.02273028
+ X20      0.01679194
+ X5       0.01645997
+ X30      0.01588675
+ X29      0.01581756
+ X18      0.01577862
+ X4       0.01566672
+ X2       0.01555086
+ X27      0.01528638
+ X24      0.01501231
+ X26      0.01490031
+ X22      0.01485927
+ X17      0.01479561
+ X16      0.01455134
+ X6       0.01430869
<none>     0.01424100
+ X12      0.01408248
+ X25      0.01406329
+ X28      0.01404746
+ X21      0.01396619
+ X7       0.01396611
+ X13      0.01395463
+ X19      0.01382341
+ X15      0.01351750
+ X10      0.01344581

```

```

+ X11          0.01329074
+ X14          0.01319425
+ X31          0.01316465
+ X23          0.01315555
+ X9           0.01302842
+ X8           0.01237923

      Df      AIC      F Pr(>F)
+ X20   1 -10.304 1.2283  0.016 *
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1    1

```

Step: R2.adj= 0.01679194  
Call: woodcom.hel ~ X3 + X1 + X20

	R2.adjusted
<All variables>	0.02273028
+ X5	0.01906638
+ X30	0.01848649
+ X29	0.01841650
+ X18	0.01837711
+ X4	0.01826391
+ X2	0.01814670
+ X27	0.01787914
+ X24	0.01760188
+ X26	0.01748859
+ X22	0.01744707
+ X17	0.01738267
+ X16	0.01713555
+ X6	0.01689008
<none>	0.01679194
+ X12	0.01666125
+ X25	0.01664183
+ X28	0.01662582
+ X21	0.01654360
+ X7	0.01654352
+ X13	0.01653191
+ X19	0.01639916
+ X15	0.01608969
+ X10	0.01601717
+ X11	0.01586030
+ X14	0.01576268
+ X31	0.01573274
+ X23	0.01572354
+ X9	0.01559493
+ X8	0.01493819

Df AIC F Pr(>F)

```
+ X5 1 -9.5664 1.2017 0.014 *
---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1
```

```
Step: R2.adj= 0.01906638
Call: woodcom.hel ~ X3 + X1 + X20 + X5
```

```
R2.adjusted
<All variables> 0.02273028
+ X30          0.02080762
+ X29          0.02073680
+ X18          0.02069695
+ X4           0.02058241
+ X2           0.02046383
+ X27          0.02019312
+ X24          0.01991260
+ X26          0.01979797
+ X22          0.01975597
+ X17          0.01969081
+ X16          0.01944078
+ X6           0.01919243
<none>        0.01906638
+ X12          0.01896090
+ X25          0.01894126
+ X28          0.01892505
+ X21          0.01884187
+ X7           0.01884179
+ X13          0.01883004
+ X19          0.01869573
+ X15          0.01838262
+ X10          0.01830924
+ X11          0.01815053
+ X14          0.01805176
+ X31          0.01802147
+ X23          0.01801216
+ X9           0.01788203
+ X8           0.01721757
```

```
Df      AIC      F Pr(>F)
+ X30 1 -8.7925 1.1529 0.044 *
---
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1
```

```
Step: R2.adj= 0.02080762
Call: woodcom.hel ~ X3 + X1 + X20 + X5 + X30
```

```
R2.adjusted
<All variables> 0.02273028
```

```

+ X29          0.02251866
+ X18          0.02247833
+ X4           0.02236243
+ X2           0.02224243
+ X27          0.02196851
+ X24          0.02168465
+ X26          0.02156865
+ X22          0.02152615
+ X17          0.02146021
+ X16          0.02120721
+ X6           0.02095590
<none>         0.02080762
+ X12          0.02072162
+ X25          0.02070174
+ X28          0.02068534
+ X21          0.02060117
+ X7           0.02060109
+ X13          0.02058919
+ X19          0.02045329
+ X15          0.02013645
+ X10          0.02006220
+ X11          0.01990160
+ X14          0.01980166
+ X31          0.01977100
+ X23          0.01976158
+ X9           0.01962991
+ X8           0.01895753

```

	Df	AIC	F	Pr(>F)
+ X29	1	-8.0286	1.1488	0.054 .
---				
Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1				

	R2.adj	Df	AIC	F	Pr(>F)
+ X3	0.007446086	1	-11.374479	1.675175	0.002
+ X1	0.014240998	1	-11.027857	1.613484	0.002
+ X20	0.016791943	1	-10.303663	1.228317	0.016
+ X5	0.019066375	1	-9.566450	1.201722	0.014
+ X30	0.020807617	1	-8.792466	1.152929	0.044
<All variables>	0.022730285	NA	NA	NA	NA

```

In [94]: ## subset to these, check the model again:
          attributes(step.res$terms)$term.labels

1. 'X3' 2. 'X1' 3. 'X20' 4. 'X5' 5. 'X30'

In [95]: ## subset to these, check the model again:

```

```

sigPCNM <- attributes(step.res$terms)$term.labels
sigPCNM <- sigPCNM[c(2,1,4,3,5)]

In [96]: woodPCNM <- ptd.PCNM[,sigPCNM]
          wood.pcnm.rda2 <- rda(woodcom.hel ~ ., woodPCNM)
          axes.test <- anova.cca(wood.pcnm.rda2, by='axis') ## takes a minute

```

In [97]: axes.test

	Df	Variance	F	Pr(>F)
RDA1	1	0.021677372	2.2901188	0.001
RDA2	1	0.013171046	1.3914630	0.014
RDA3	1	0.011633696	1.2290487	0.105
RDA4	1	0.010018368	1.0583964	0.557
RDA5	1	0.008930355	0.9434526	0.748
Residual	85	0.804576845	NA	NA

In [98]: wood.pcnm.rda2

```
Call: rda(formula = woodcom.hel ~ X1 + X3 + X5 + X20 + X30, data =
woodPCNM)
```

	Inertia	Proportion	Rank
Total	0.87001	1.00000	
Constrained	0.06543	0.07521	5
Unconstrained	0.80458	0.92479	85

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2	RDA3	RDA4	RDA5
0.021677	0.013171	0.011634	0.010018	0.008930

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.031096	0.026215	0.025264	0.021274	0.017924	0.017047	0.016374	0.014673

(Showed only 8 of all 85 unconstrained eigenvalues)

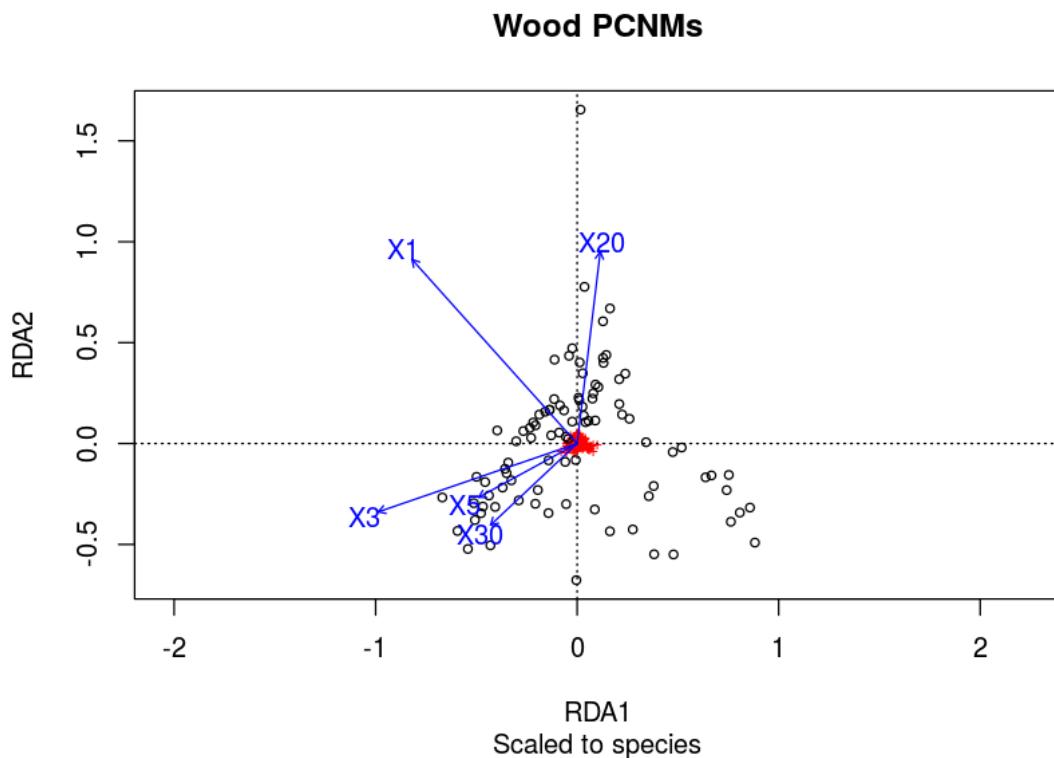
In [99]: woodRDA2.sig <- anova.cca(wood.pcnm.rda2)

In [100]: woodRDA2.sig

	Df	Variance	F	Pr(>F)
Model	5	0.06543084	1.382496	0.001
Residual	85	0.80457685	NA	NA

In [104]: #save(woodPCNM, file='woodPCNM.rda')

```
In [101]: options(repr.plot.height = 5)
plot(wood.pcnm.rda2,
      display = c('sp','wa','bp'),
      scaling = 2,
      sub='Scaled to species',
      main='Wood PCNMs'
)
```



Here is our map plotter again, tweaked for wood:

```
In [98]: source("/home/daniel/Documents/taiwan/taiwan_combined_stats/CSrev/NEWR-2ed_code_data/NEWR.R")
In [99]: mapP <- function(PCNM, P, bkg){
  require('png')
  topo <- readPNG(bkg)
  plot(1, type='n',
       xlim=c(0,500),
       ylim=c(0,500),
       xlab = '',
       ylab = '',
       main = paste('PCNM', colnames(PCNM)[P], sep = ' '))
  rasterImage(topo,0,0,500,500)
```

```

        sr.value(dfxy=woodxy, z=PCNM[,P], add.plot = TRUE) ## changed for woodXY
    }
#save(mapP, file='mapPwood.R')

```

In [111]: `head(woodPCNM)`

	X1	X3	X5	X20	X30
1	116.3280	-63.82517	-17.26254	-4.966470	1.4163441
2	115.6676	-65.31071	-18.74772	-10.672158	-0.2971716
3	115.6859	-65.32292	-18.75275	-9.964486	-0.2800219
4	115.6604	-65.30261	-18.74324	-10.661146	-0.2841266
5	115.6917	-65.32323	-18.75155	-9.246396	-0.2498117
7	115.7133	-65.34755	-18.76499	-9.279430	-0.2889467

In [103]: `load('woodPCNM.rda')`

`load('mapPwood.R')`

`load("woodxy.rda")`

`source("/home/daniel/Documents/taiwan/taiwan_combined_stats/CSrev/NEwR-2ed_code_data/NEwR.R")`

In [105]: `## make svgs`

`par(pty="s")`

`par(mfrow=c(1,1))`

`for (i in 1:ncol(woodPCNM)){`

`svg(paste0('allHostWoodPCNMs','_',i,'.svg'))`

`mapP(woodPCNM,i,'/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png')`

`dev.off()`

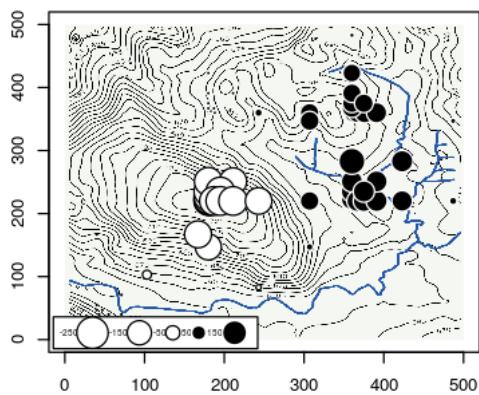
`}`

In [113]: `options(repr.plot.height = 10)`

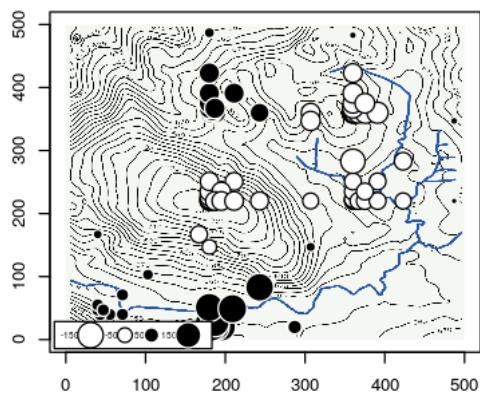
`par(mfrow=c(3,2))`

`for (i in 1:ncol(woodPCNM)) mapP(woodPCNM, i, '/home/daniel/Documents/taiwan/taiwan_co`

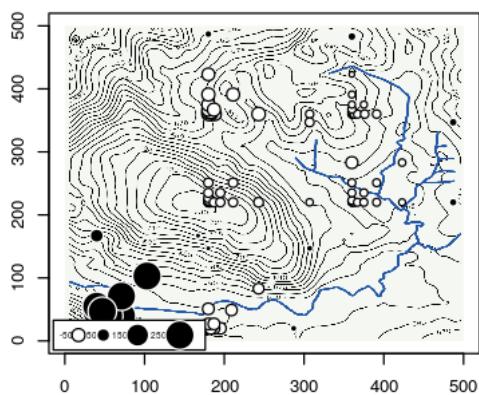
PCNM X1



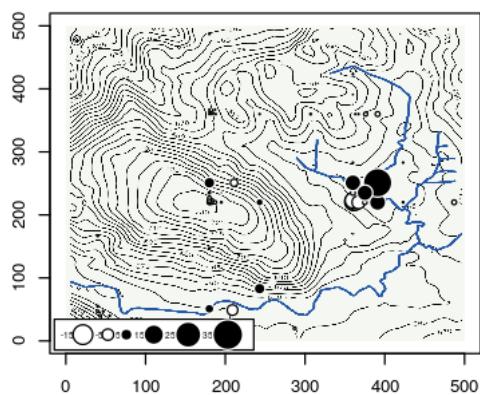
PCNM X3



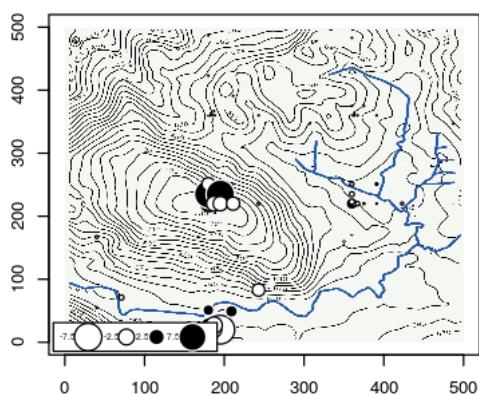
PCNM X5



PCNM X20



PCNM X30



Check environmental correlations

```
In [114]: ## get wood environmental data
    wood.env <- data.frame(sample_data(wood95)[,c('Forest_Type', 'vegcom')])
    rownames(wood.env) <- gsub("w.*","", rownames(wood.env) )
    wood.env$Forest_Type <- as.factor(wood.env$Forest_Type)
    wood.env$vegcom <- as.factor(wood.env$vegcom)
    #save(wood.env, file='wood.env.rda')
```

```
In [120]: ## run linear models
    woodPCNM.X1 <- lm(woodPCNM[,1] ~ ., data=wood.env)
    woodPCNM.X3 <- lm(woodPCNM[,2] ~ ., data=wood.env)
    woodPCNM.X5 <- lm(woodPCNM[,3] ~ ., data=wood.env)
    woodPCNM.X20 <- lm(woodPCNM[,4] ~ ., data=wood.env)
    woodPCNM.X30 <- lm(woodPCNM[,5] ~ ., data=wood.env)
```

```
In [121]: summary(woodPCNM.X1)
```

Call:

```
lm(formula = woodPCNM[, 1] ~ ., data = wood.env)
```

Residuals:

Min	1Q	Median	3Q	Max
-141.871	-37.025	0.213	29.077	109.758

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )		
(Intercept)	-209.1236	37.1697	-5.626	2.56e-07 ***		
Forest_Type2	0.1747	40.4077	0.004	0.996561		
Forest_Type3	119.1983	45.5234	2.618	0.010541 *		
Forest_Type4	131.2801	63.3136	2.073	0.041305 *		
Forest_Type5	252.6981	63.4404	3.983	0.000148 ***		
Forest_Type6	202.4282	60.6131	3.340	0.001270 **		
Forest_Type7	264.4201	59.3824	4.453	2.68e-05 ***		
vegcom2	21.3763	46.0719	0.464	0.643909		
vegcom3	-13.6048	46.0719	-0.295	0.768525		
vegcom4	-55.7532	70.0561	-0.796	0.428453		
---						
Signif. codes:	0 ***	0.001 **	0.01 *	0.05 .	0.1	1

Residual standard error: 52.57 on 81 degrees of freedom

Multiple R-squared: 0.8017, Adjusted R-squared: 0.7797

F-statistic: 36.39 on 9 and 81 DF, p-value: < 2.2e-16

```
In [123]: summary(woodPCNM.X3)
```

```

Call:
lm(formula = woodPCNM[, 2] ~ ., data = wood.env)

Residuals:
    Min      1Q  Median      3Q     Max 
-107.951 -49.454   -6.155  18.365 228.408 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -80.24273  51.40126 -1.561  0.12240  
Forest_Type2  0.07185  55.87907  0.001  0.99898  
Forest_Type3  96.26621  62.95343  1.529  0.13012  
Forest_Type4  89.81588  87.55515  1.026  0.30803  
Forest_Type5 142.29148  87.73057  1.622  0.10871  
Forest_Type6 254.49730  83.82073  3.036  0.00322 ** 
Forest_Type7 133.25377  82.11884  1.623  0.10854  
vegcom2       -92.91954  63.71202 -1.458  0.14859  
vegcom3        33.31424  63.71202  0.523  0.60248  
vegcom4       -16.28437  96.87928 -0.168  0.86693  
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1    1

Residual standard error: 72.69 on 81 degrees of freedom
Multiple R-squared:  0.5385, Adjusted R-squared:  0.4872 
F-statistic: 10.5 on 9 and 81 DF,  p-value: 1.358e-10

```

In [124]: `summary(woodPCNM.X5)`

```

Call:
lm(formula = woodPCNM[, 3] ~ ., data = wood.env)

Residuals:
    Min      1Q  Median      3Q     Max 
-130.341 -10.321   -6.484  10.532 256.505 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -2.449e+01  4.461e+01 -0.549  0.58455  
Forest_Type2  4.093e-04  4.849e+01  0.000  0.99999  
Forest_Type3  1.495e+01  5.463e+01  0.274  0.78506  
Forest_Type4  1.158e+02  7.598e+01  1.524  0.13141  
Forest_Type5 -2.738e+01  7.613e+01 -0.360  0.72010  
Forest_Type6 -5.305e+01  7.274e+01 -0.729  0.46791  
Forest_Type7  1.572e+00  7.126e+01  0.022  0.98245  

```

```

vegcom2      1.267e+01  5.529e+01   0.229  0.81934
vegcom3      6.843e+01  5.529e+01   1.238  0.21945
vegcom4      2.692e+02  8.407e+01   3.202  0.00195 **

---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 63.08 on 81 degrees of freedom
Multiple R-squared:  0.4286, Adjusted R-squared:  0.3651
F-statistic: 6.749 on 9 and 81 DF,  p-value: 3.645e-07

```

In [125]: `summary(woodPCNM.X20)`

Call:  
`lm(formula = woodPCNM[, 4] ~ ., data = wood.env)`

Residuals:

Min	1Q	Median	3Q	Max
-10.643	-0.447	0.017	0.211	32.011

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-0.5998	3.4892	-0.172	0.864
Forest_Type2	0.2655	3.7931	0.070	0.944
Forest_Type3	2.4430	4.2733	0.572	0.569
Forest_Type4	3.0389	5.9433	0.511	0.611
Forest_Type5	2.6886	5.9552	0.451	0.653
Forest_Type6	2.4007	5.6898	0.422	0.674
Forest_Type7	2.4655	5.5743	0.442	0.659
vegcom2	-1.8953	4.3248	-0.438	0.662
vegcom3	-2.0967	4.3248	-0.485	0.629
vegcom4	-1.2670	6.5762	-0.193	0.848

Residual standard error: 4.934 on 81 degrees of freedom
Multiple R-squared: 0.009097, Adjusted R-squared: -0.101
F-statistic: 0.08263 on 9 and 81 DF, p-value: 0.9998

In [122]: `summary(woodPCNM.X30)`

Call:  
`lm(formula = woodPCNM[, 5] ~ ., data = wood.env)`

Residuals:

Min	1Q	Median	3Q	Max
-----	----	--------	----	-----

```

-8.0828 -0.2426 0.1045 0.1631 7.5708

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -2.26915   1.28311 -1.768   0.0807 .
Forest_Type2 2.82912   1.39488  2.028   0.0458 *
Forest_Type3 1.86740   1.57148  1.188   0.2382
Forest_Type4 1.50467   2.18560  0.688   0.4931
Forest_Type5 2.02773   2.18998  0.926   0.3572
Forest_Type6 2.13547   2.09238  1.021   0.3105
Forest_Type7 2.00299   2.04990  0.977   0.3314
vegcom2       0.13274   1.59041  0.083   0.9337
vegcom3       0.67775   1.59041  0.426   0.6711
vegcom4       0.00778   2.41835  0.003   0.9974
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1.815 on 81 degrees of freedom
Multiple R-squared:  0.06772, Adjusted R-squared:  -0.03587
F-statistic: 0.6537 on 9 and 81 DF,  p-value: 0.7478

```

The first PCNM seems heavily correlated with the microtopography variable ("forest type").  
Comparisons of all samples to hilltop  
Leaf endophytes compared to hilltop  
Leaf comparison to hilltop in euclidean space (NMS)

In [273]: `load('deseq95.rda')`

```

## make our dataframe of spatial coordinates and BC index:
leaf95 <- subset_samples(deseq95, Library == 'L') ## general leaf phyloseq obj
leafHel95 <- subset_samples(leaf95, Host_genus_species == 'Helicia_formosana') ## helicia

bb <- t(otu_table(leafHel95)) ## get otu table for Helicia leaves
bb[bb > 0] <- 1 ## presence/absence
cc <- vegdist(bb, method='bray') ## makes a triangular association matrix
dd <- as.matrix(cc) ## convert to full, symmetric matrix
leafheltopBC <- dd[, '72leaf'] ## extract only the comparisons to our hilltop Helicia samples

all(names(leafheltopBC) == rownames(sample_data(leafHel95))) ## check, compatible with sample data

```

TRUE

In [229]: `## map a dataframe with the info we want to plot:`  
`mapBC <- cbind(sample_data(leafHel95)[,c('X','Y')], leafheltopBC)`  
`colnames(mapBC)[3] <- 'BC'`  
`## order by BC dissimilarity`

```

mapBC <- mapBC[order(mapBC$BC),]
head(mapBC)



|         | X   | Y   | BC        |
|---------|-----|-----|-----------|
| 72leaf  | 183 | 223 | 0.0000000 |
| 99leaf  | 183 | 23  | 0.6097561 |
| 127leaf | 40  | 103 | 0.6226415 |
| 97leaf  | 180 | 21  | 0.6470588 |
| 8leaf   | 360 | 227 | 0.6875000 |
| 33leaf  | 423 | 360 | 0.6923077 |



In [230]: mapBCrev <- mapBC[rev(rownames(mapBC)),]

In [235]: ## make a heat map palette useing colorbrewer
my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)
## need an "adapter", for our BC values to this heat map color scheme:
BCroundup <- rev(round(mapBC$BC*100+1))

options(repr.plot.width = 10, repr.plot.height = 7)

#png('hilltopmap.png')

topo <- readPNG('/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png') ## load
plot(1, type='n',
      xlim=c(0,500),
      ylim=c(0,500),
      xlab = '',
      ylab = '',
      asp=1,
      main='Helicia formosana leaves, comparison to Hilltop'
) ## blank plot
rasterImage(topo,0,0,500,500) ## add raster of our plot

## add Helicia points, colored by similarity
points(mapBCrev[,c('X','Y')],
      pch=21,
      cex=1.5,
      bg = my_palette2[BCroundup],
      lwd=1.5,
      )

#draw.circle(x=183, y=223, radius=200, border="red", lwd=2, lty=2) ## use the plotrix pac

## make a legend:
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'

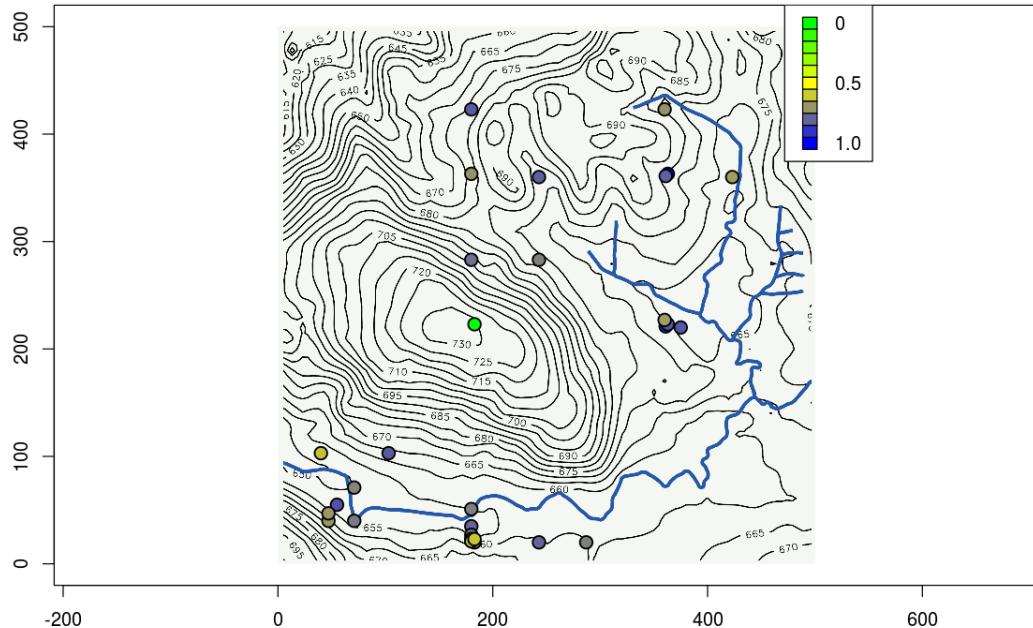
```

```

        legend("topright",
               fill = my_palette2[seq(1,101,10)],
               legend = heatlegend,
               bg = "white",
               y.intersp = 0.5,
)
#dev.off()

```

**Helicia formosana leaves, comparison to Hilltop**



Leaf comparison to hilltop in BC dissimilarity space (NMS)

Where does this hilltop tend to be when we do an NMS ordination of the BC distances?

```
In [274]: leafOTU <- t(otu_table(leaf95))
leafOTU[leafOTU > 0] <- 1
leafMDS <- metaMDS(leafOTU)
```

```
Run 0 stress 0.1260773
Run 1 stress 0.126222
... Procrustes: rmse 0.008143772  max resid 0.03413517
Run 2 stress 0.1260959
... Procrustes: rmse 0.006410951  max resid 0.03108768
Run 3 stress 0.1259856
... New best solution
... Procrustes: rmse 0.006827669  max resid 0.0181532
Run 4 stress 0.1265765
```

```

Run 5 stress 0.1262299
... Procrustes: rmse 0.008884247 max resid 0.02344429
Run 6 stress 0.1259681
... New best solution
... Procrustes: rmse 0.002355039 max resid 0.01238553
Run 7 stress 0.1260302
... Procrustes: rmse 0.006796167 max resid 0.0193296
Run 8 stress 0.126221
... Procrustes: rmse 0.008384131 max resid 0.01884438
Run 9 stress 0.1261614
... Procrustes: rmse 0.008310365 max resid 0.02453497
Run 10 stress 0.1262036
... Procrustes: rmse 0.008942904 max resid 0.02465052
Run 11 stress 0.1262609
... Procrustes: rmse 0.006660805 max resid 0.01839853
Run 12 stress 0.1261559
... Procrustes: rmse 0.006421822 max resid 0.03125817
Run 13 stress 0.126263
... Procrustes: rmse 0.007207525 max resid 0.02365601
Run 14 stress 0.126007
... Procrustes: rmse 0.004006621 max resid 0.0165842
Run 15 stress 0.12611
... Procrustes: rmse 0.005868089 max resid 0.02346685
Run 16 stress 0.1260058
... Procrustes: rmse 0.002647116 max resid 0.01497188
Run 17 stress 0.126162
... Procrustes: rmse 0.007435341 max resid 0.02237596
Run 18 stress 0.1262586
... Procrustes: rmse 0.008244743 max resid 0.03657024
Run 19 stress 0.1260916
... Procrustes: rmse 0.006237917 max resid 0.03297871
Run 20 stress 0.1260675
... Procrustes: rmse 0.004404558 max resid 0.03062487
*** No convergence -- monoMDS stopping criteria:
 20: stress ratio > sratmax

```

```

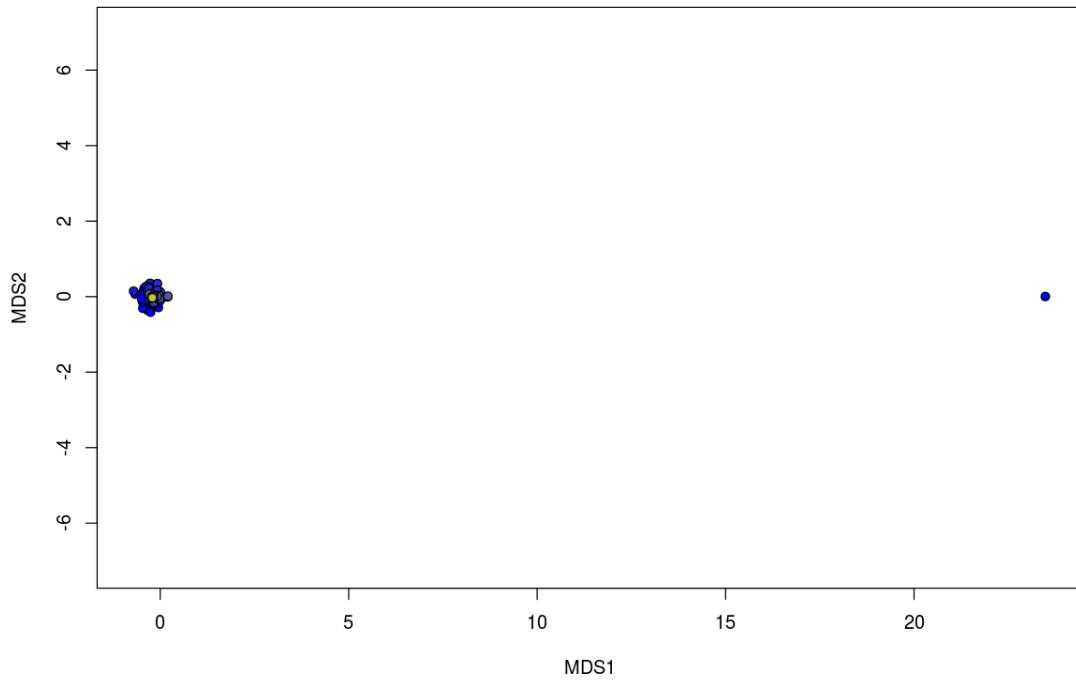
In [276]: allBC <- as.matrix(vegdist(leafOTU))[, "72leaf", drop=FALSE]
          allBCroundup <- round(allBC*100+1) ## turn our allBC values into heatmap values
          my_palette2 <- colorRampPalette(c("green", "yellow", "blue"))(n = 101)

```

```

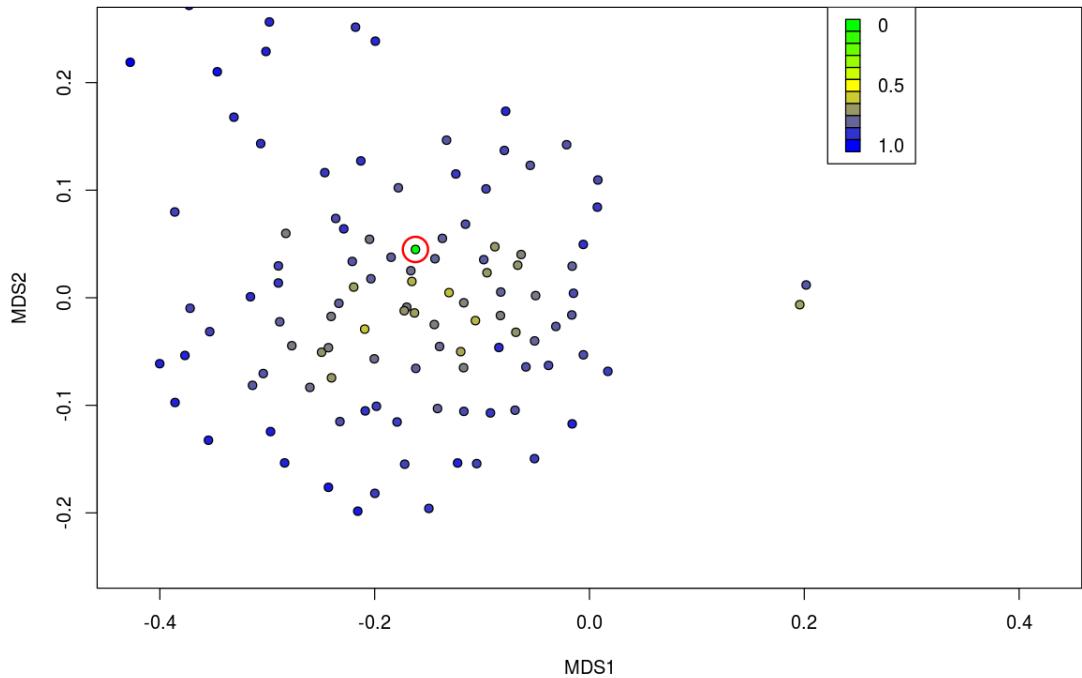
In [278]: plot(leafMDS$points,
           pch=21,
           bg = my_palette2[allBCroundup[,1]],
           asp=1
         )

```



Zoom in, without the outlier:

```
In [281]: #png('hilltopNMS.png')
plot(leafMDS$points,
      pch=21,
      bg = my_palette2[allBCroundup[,1]],
      xlim = c(-0.25,0.25),
      ylim = c(-0.25,0.25),
      asp=1
)
legend("topright",
      fill = my_palette2[seq(1,101,10)],
      legend = heatlegend,
      bg = "white",
      y.intersp = 0.5,
)
points(leafMDS$points['72leaf',1],
      leafMDS$points['72leaf',2],
      pch = 1,
      cex = 3,
      col = 'red',
      lwd=2,
)
#dev.off()
```



```
In [285]: #identify(leafMDS$points) ## ID the outlier point
#leafMDS$points[90,drop=FALSE] ## get its dataframe position/rowname

leafNoOut <- leafOTU[row.names(leafOTU) != "67leaf",] ## get rid of ourlier row

In [286]: leafMDSnoOut <- metaMDS(leafNoOut)

Run 0 stress 0.228814
Run 1 stress 0.2231124
... New best solution
... Procrustes: rmse 0.07646505 max resid 0.3547096
Run 2 stress 0.2248534
Run 3 stress 0.2254021
Run 4 stress 0.224686
Run 5 stress 0.2286426
Run 6 stress 0.2233411
... Procrustes: rmse 0.03047058 max resid 0.1988328
Run 7 stress 0.225404
Run 8 stress 0.2244533
Run 9 stress 0.2254235
Run 10 stress 0.2243584
Run 11 stress 0.2265294
Run 12 stress 0.2241343
```

```

Run 13 stress 0.2255818
Run 14 stress 0.2233027
... Procrustes: rmse 0.03012185 max resid 0.2012423
Run 15 stress 0.2246742
Run 16 stress 0.2256866
Run 17 stress 0.2248967
Run 18 stress 0.2258309
Run 19 stress 0.2241649
Run 20 stress 0.2249329
*** No convergence -- monoMDS stopping criteria:
  4: no. of iterations >= maxit
  16: stress ratio > sratmax

```

```

In [276]: allBC <- as.matrix(vegdist(leafNoOut))[, "72leaf", drop=FALSE]
           allBCroundup <- round(allBC*100+1) ## turn our allBC values into heatmap values
           my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)

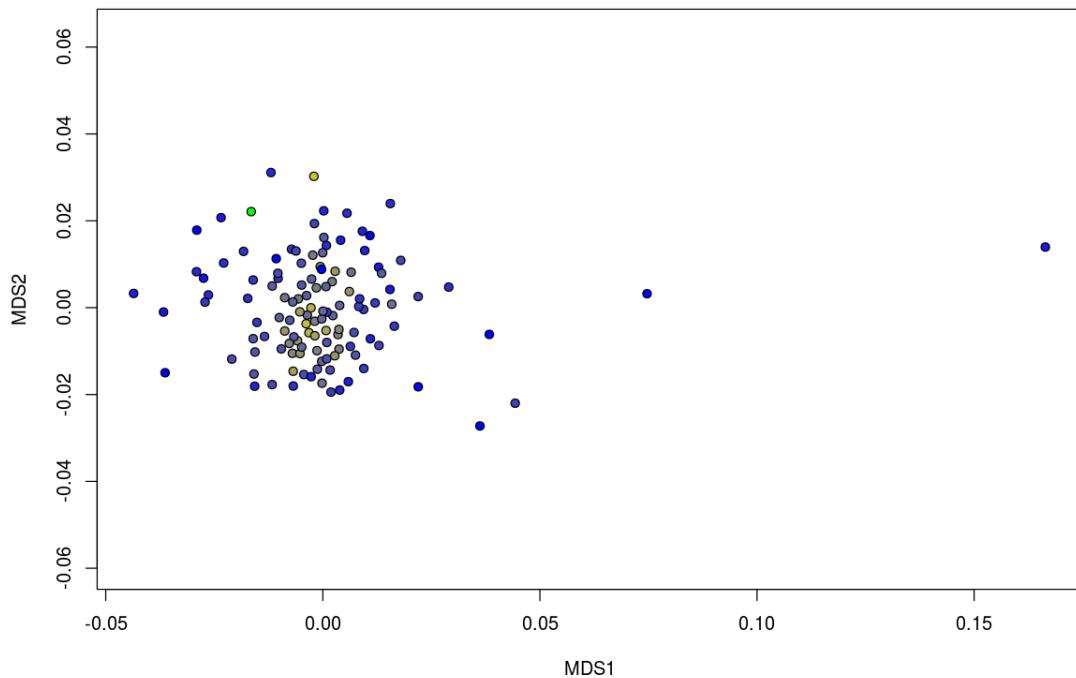
```

```

In [289]: plot(leafMDSnOut$points,
            pch=21,
            bg = my_palette2[allBCroundup[,1]],
            asp=1,
            main='Leaf comparison to hilltop, without outliers'
        )

```

**Leaf comparison to hilltop, without outliers**



As mentioned above, this newer analysis doesn't repeat our results from last analysis about the hilltop being central in the dissimilarity space (NMS). Drop this from the manuscript.

Wood endophytes compared to hilltop

Wood comparison to hilltop in euclidean space (NMS)

```
In [264]: woodOTU <- t(otu_table(wood95))
woodOTU[woodOTU > 0] <- 1

woodMDS <- metaMDS(woodOTU)

Run 0 stress 0.2752839
Run 1 stress 0.2772215
Run 2 stress 0.2731556
... New best solution
... Procrustes: rmse 0.08576075 max resid 0.3949623
Run 3 stress 0.2717249
... New best solution
... Procrustes: rmse 0.03333857 max resid 0.1799176
Run 4 stress 0.2841524
Run 5 stress 0.275739
Run 6 stress 0.2719552
... Procrustes: rmse 0.008702442 max resid 0.06319884
Run 7 stress 0.2831439
Run 8 stress 0.2774238
Run 9 stress 0.2758084
Run 10 stress 0.279804
Run 11 stress 0.2743058
Run 12 stress 0.2816734
Run 13 stress 0.2783536
Run 14 stress 0.286105
Run 15 stress 0.2771506
Run 16 stress 0.2769533
Run 17 stress 0.2726776
Run 18 stress 0.2741197
Run 19 stress 0.2743129
Run 20 stress 0.2755739
*** No convergence -- monoMDS stopping criteria:
20: stress ratio > sratmax
```

```
In [259]: ## map a dataframe with the info we want to plot:
mapBC <- cbind(sample_data(woodHel95)[,c('X','Y')], woodheltopBC)
colnames(mapBC)[3] <- 'BC'
## order by BC dissimilarity
mapBC <- mapBC[order(mapBC$BC),]
head(mapBC)
```

	X	Y	BC
72w	183	223	0.0000000
28w	363	363	0.7076923
25w	361	361	0.7404580
96w	181	21	0.7678571
14w	375	220	0.7964072
99w	183	23	0.7966102

```
In [260]: mapBCrev <- mapBC[rev(rownames(mapBC)),]
```

```
In [262]: ## make a heat map palette useing colorbrewer
my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)
## need an "adapter", for our BC values to this heat map color scheme:
BCroundup <- rev(round(mapBC$BC*100+1))
```

```
options(repr.plot.width = 10, repr.plot.height = 7)
```

```
#png('hilltopmap.png')
```

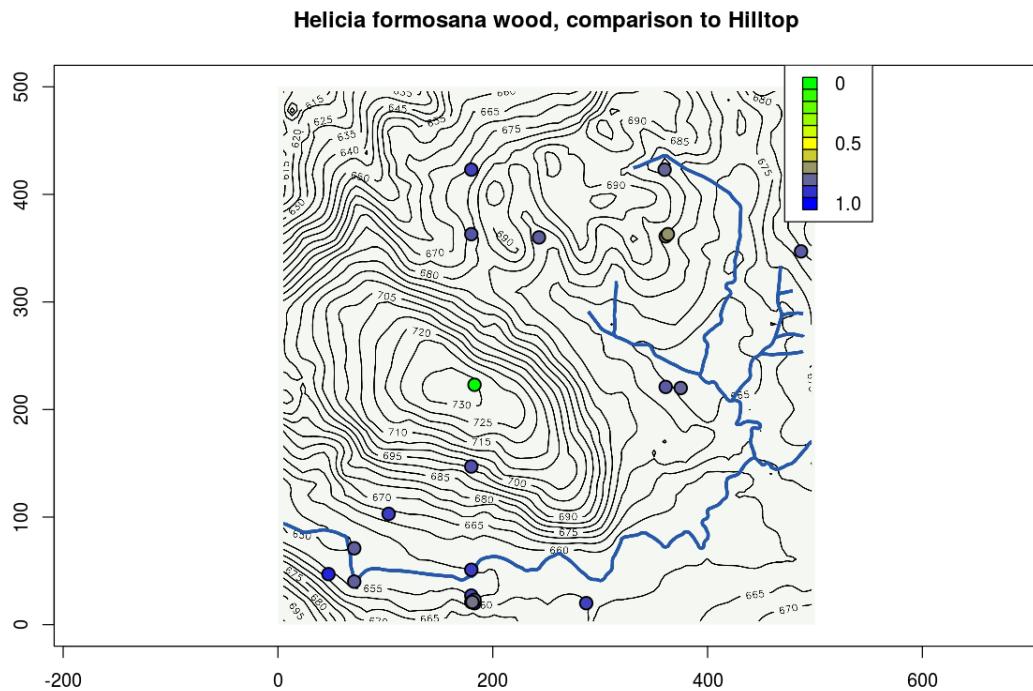
```
topo <- readPNG('/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png') ## load topo
plot(1, type='n',
      xlim=c(0,500),
      ylim=c(0,500),
      xlab = '',
      ylab = '',
      asp=1,
      main='Helicia formosana wood, comparison to Hilltop'
) ## blank plot
rasterImage(topo,0,0,500,500) ## add raster of our plot
```

```
## add Helicia points, colored by similarity
points(mapBCrev[,c('X','Y')],
      pch=21,
      cex=1.5,
      bg = my_palette2[BCroundup],
      lwd=1.5,
      )
```

```
#draw.circle(x=183, y=223, radius=200, border="red", lwd=2, lty=2) ## use the plotrix package
```

```
## make a legend:
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'
legend("topright",
      fill = my_palette2[seq(1,101,10)],
```

```
    legend = heatlegend,  
    bg = "white",  
    y.intersp = 0.5,  
)  
#dev.off()
```



Map wood comparison to hilltop in BC dissimilarity space (NMS)

```
In [269]: woodOTU <- t(otu_table(wood95))
woodOTU[woodOTU > 0] <- 1

woodMDS <- metaMDS(woodOTU)

allBC <- as.matrix(vegdist(woodOTU))[, "72w", drop=FALSE]
allBCroundup <- round(allBC*100+1) ## turn our allBC values into heatmap values
my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)

Run 0 stress 0.2752839
Run 1 stress 0.2749867
... New best solution
... Procrustes: rmse 0.05399962 max resid 0.3481395
Run 2 stress 0.2791709
Run 3 stress 0.2768793
Run 4 stress 0.2742886
```

```

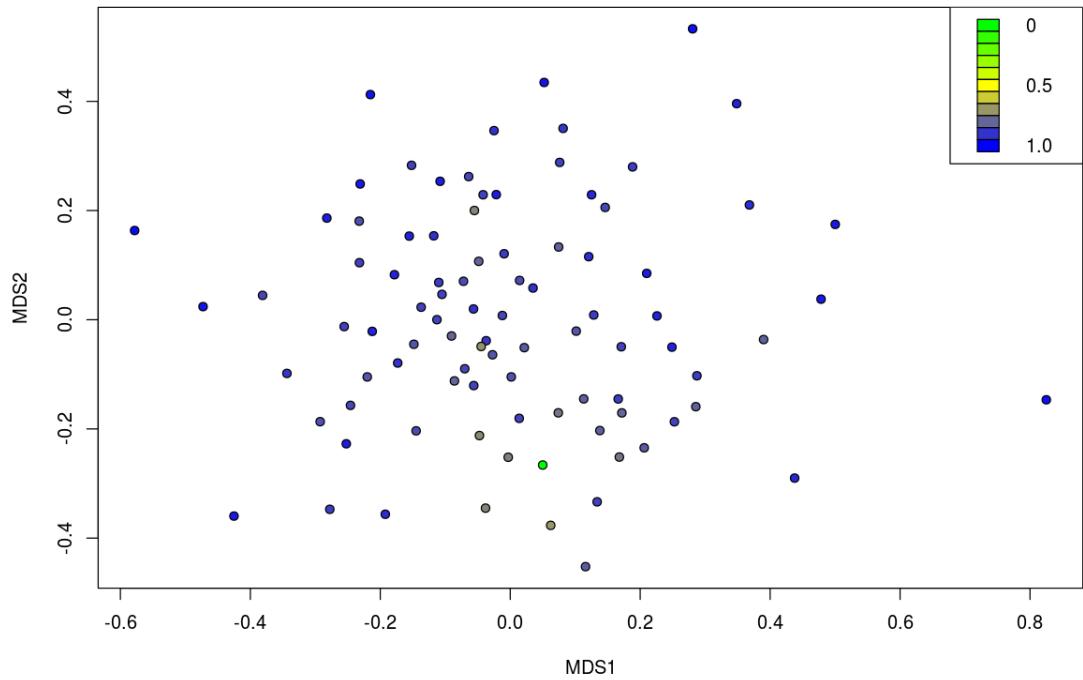
... New best solution
... Procrustes: rmse 0.08608061  max resid 0.3459764
Run 5 stress 0.2734657
... New best solution
... Procrustes: rmse 0.01634504  max resid 0.08382424
Run 6 stress 0.2764642
Run 7 stress 0.2736337
... Procrustes: rmse 0.04964461  max resid 0.334079
Run 8 stress 0.2779225
Run 9 stress 0.2891091
Run 10 stress 0.2746194
Run 11 stress 0.2731145
... New best solution
... Procrustes: rmse 0.02257999  max resid 0.1305221
Run 12 stress 0.2806462
Run 13 stress 0.2805967
Run 14 stress 0.2769614
Run 15 stress 0.2937179
Run 16 stress 0.2744863
Run 17 stress 0.2733944
... Procrustes: rmse 0.02474187  max resid 0.1829851
Run 18 stress 0.2972708
Run 19 stress 0.2782096
Run 20 stress 0.2958267
*** No convergence -- monoMDS stopping criteria:
  1: no. of iterations >= maxit
  19: stress ratio > sratmax

```

```

In [270]: plot(woodMDS$points,
            pch=21,
            bg = my_palette2[allBCroundup[,1]],
            #      asp=1,
            #ylim=c(-0.2, 0.2),
            #xlim=c(-0.5, 0.5),
            )
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'
legend("topright",
       fill = my_palette2[seq(1,101,10)],
       legend = heatlegend,
       bg = "white",
       y.intersp = 0.5,
       )

```



### Spatial patterns of Helicia mycobiome

Let's look at the spatial patterns of the mycobiome of just one host, *Helicia formosana*.

```
In [63]: load('deseq95.rda')
leafHel95 <- subset_samples(deseq95, Library == 'L' & Host_genus_species == "Helicia_formosana")
woodHel95 <- subset_samples(deseq95, Library == 'W' & Host_genus_species == "Helicia_formosana")
```

How many OTUs are present in just *Helicia*?

```
In [67]: ## leaves:
sum(rowSums(otu_table(leafHel95)) > 0)
```

426

```
In [68]: ## wood:
sum(rowSums(otu_table(woodHel95)) > 0)
```

731

Check the environmental variables we have:

```
In [188]: leafHelOTU <- t(otu_table(leafHel95))
leafHelOTU[leafHelOTU > 0] <- 1

ff <- as.factor(sample_data(leafHel95)$Forest_Type)
```

```
gg <- as.factor(sample_data(leafHel95)$vegcom)

lf <- adonis(leafHelOTU ~ ff, permutations=10000) ## topography
lg <- adonis(leafHelOTU ~ gg, permutations=10000) ## vegetative community
```

In [189]: lf

Call:  
adonis(formula = leafHelOTU ~ ff, permutations = 10000)

Permutation: free  
Number of permutations: 10000

Terms added sequentially (first to last)

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
ff	4	1.2156	0.30389	0.92587	0.12468	0.657
Residuals	26	8.5337	0.32822		0.87532	
Total	30	9.7493			1.00000	

In [190]: lg

Call:  
adonis(formula = leafHelOTU ~ gg, permutations = 10000)

Permutation: free  
Number of permutations: 10000

Terms added sequentially (first to last)

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
gg	3	1.0377	0.34589	1.072	0.10644	0.2717
Residuals	27	8.7116	0.32265		0.89356	
Total	30	9.7493			1.00000	

```
In [191]: woodHelOTU <- t(otu_table(woodHel95))
woodHelOTU[woodHelOTU > 0] <- 1

ff <- as.factor(sample_data(woodHel95)$Forest_Type)
gg <- as.factor(sample_data(woodHel95)$vegcom)

lf <- adonis(woodHelOTU ~ ff, permutations=10000) ## topography
lg <- adonis(woodHelOTU ~ gg, permutations=10000) ## vegetative community
```

In [193]: lf; lg

```

Call:
adonis(formula = woodHelOTU ~ ff, permutations = 10000)

Permutation: free
Number of permutations: 10000

Terms added sequentially (first to last)

      Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
ff        4    1.3655 0.34137  1.0643 0.20028 0.2022
Residuals 17    5.4525 0.32074           0.79972
Total      21    6.8180                   1.00000

```

```

Call:
adonis(formula = woodHelOTU ~ gg, permutations = 10000)

Permutation: free
Number of permutations: 10000

Terms added sequentially (first to last)

      Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
gg        3    1.0394 0.34648  1.0793 0.15245 0.188
Residuals 18    5.7786 0.32103           0.84755
Total      21    6.8180                   1.00000

```

Neither *Helicia* wood or leaf endos are displaying direct responses to our environmental variables.

#### Helicia mantel tests

```

In [3]: ## wood endophytes
## make our community distance matrix
aa <- t(otu_table(woodHel95))
aa[aa > 0] <- 1 ## P/A
woodHel_comdist <- vegdist(aa, method = "bray")

In [4]: ## make our physical distance matrix
cc <- as.matrix(sample_data(woodHel95)[,c('X','Y')])
class(cc) <- "numeric"
physdist <- vegdist(cc, method = "euclidean")
woodHelmgram <- mgram(woodHel_comdist, physdist) ## correlogram object
woodHel_mant_test <- ecodist::mantel(woodHel_comdist ~ physdist, nperm = 10000) ## overa

Warning message in class(X) <- NULL:
Setting class(x) to NULL; result will no longer be an S4 object

```

```
In [6]: woodHel_mant_test
```

```
mantelr      0.18782127561416 pval1      0.0152 pval2      0.9849 pval3      0.0172 llim.2.5\%
0.0874457207155239 ulim.97.5\%                      0.307731867632012
```

```
In [7]: ## leaf endophytes
```

```
## make our community distance matrix
aa <- t(otu_table(leafHel95))
aa[aa > 0] <- 1 ## P/A
leafHel_comdist <- vegdist(aa, method = "bray")
## make our physical distance matrix
cc <- as.matrix(sample_data(leafHel95)[,c('X', 'Y')])
class(cc) <- "numeric"
physdist <- vegdist(cc, method = "euclidean")
leafHelmgram <- mgram(leafHel_comdist, physdist) ## correlogram object
leafHel_mant_test <- ecodist::mantel(leafHel_comdist ~ physdist, nperm = 10000) ## overa
```

Warning message in class(X) <- NULL:

Setting class(x) to NULL; result will no longer be an S4 object

```
In [8]: leafHel_mant_test
```

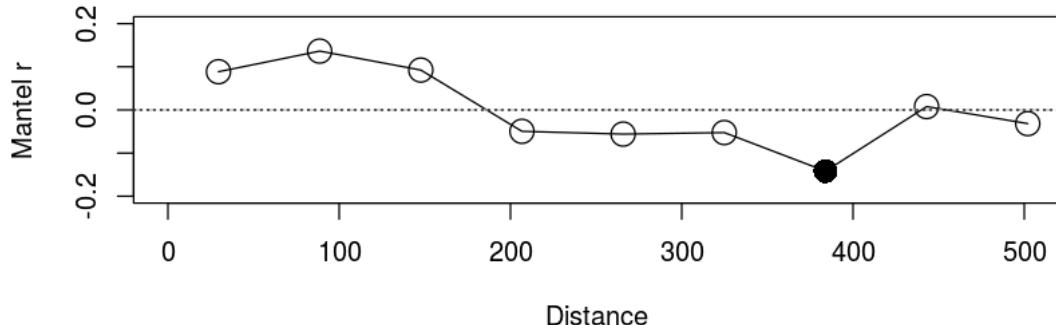
```
mantelr      0.237606273004194 pval1      0.0015 pval2      0.9986 pval3      0.0015 llim.2.5\%
0.149400750450931 ulim.97.5\%                      0.323412489883924
```

```
In [9]: options(repr.plot.height = 6)
```

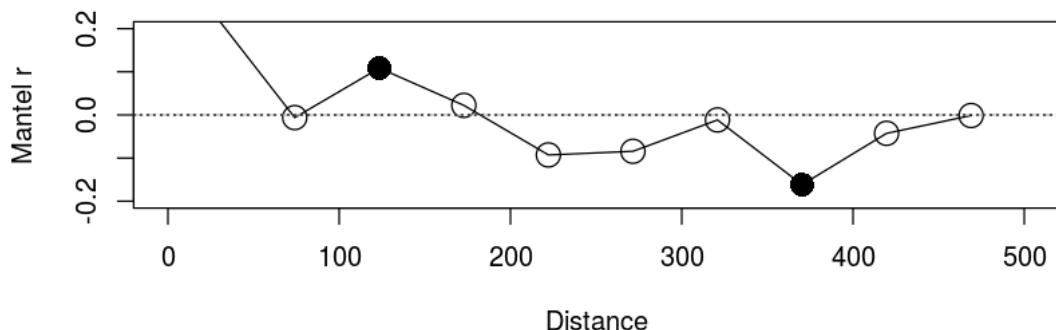
```
par(mfrow=c(2,1))
plot(woodHelmgram,
      main="Wood endophytes",
      ylim=c(-.20,.20),
      xlim=c(0,500)
    )
abline(h=0,lty=3)

plot(leafHelmgram,
      main="Leaf endophytes",
      ylim=c(-.20,.20),
      xlim=c(0,500)
    )
abline(h=0,lty=3)
```

### Wood endophytes



### Leaf endophytes



Helicia dbMEMs

Helicia leaf dbMEMs

```
In [25]: leafHelxy <- sample_data(leafHel95)[,c('X', 'Y')]
rownames(leafHelxy) <- gsub("leaf.*","",rownames(leafHelxy))
leafHelxy <- data.frame(leafHelxy[order(as.numeric(rownames(leafHelxy))),])
leafHelxy$X <- as.numeric(leafHelxy$X)
leafHelxy$Y <- as.numeric(leafHelxy$Y)

In [27]: ptd <- dist(leafHelxy)
span.ptd <- spantree(ptd)
dmin <- max(span.ptd$dist) ## 144.2 m

In [29]: ## truncate our distance matrix using this distance.
ptd[ptd > dmin] <- 4*dmin
ptd.PCoA <- cmdscale(ptd, k=nrow(leafHelxy)-1, eig = TRUE)
nb.ev <- length(which(ptd.PCoA$eig > 0.0000001)) ## keep only the positive eigenvalues
ptd.PCNM <- data.frame(ptd.PCoA$points[1:nrow(leafHelxy), 1:nb.ev])
leafHelcom <- t(otu_table(leafHel95)) ## get community matrix out of phyloseq
```

```
Warning message in cmdscale(ptd, k = nrow(leafHelxy) - 1, eig = TRUE):
only 17 of the first 30 eigenvalues are > 0
```

```
In [38]: head(ptd.PCNM)
```

	X1	X2	X3	X4	X5	X6	X7	X8	X9
3	-281.8947	-221.1453	26.57923	178.1044	32.33650	-4.309998	-5.885088	-21.95259	-9.4710
6	-281.9868	-221.3037	26.59104	177.9283	32.35275	-4.337738	-5.743736	-21.74552	-9.3852
8	-282.3043	-221.4742	26.60132	177.8339	32.02336	-5.346045	-6.165988	-22.02304	-9.8791
14	-251.4270	-240.1965	28.00565	123.2325	94.80062	163.354497	111.037212	86.39302	30.3632
25	-343.1403	-188.8784	11.58567	-128.6972	-18.62933	-57.486706	7.467029	12.49429	-0.5126
28	-342.8798	-188.6956	11.55936	-129.2759	-18.87787	-56.991101	7.657137	12.24018	-0.3814

```
In [30]: rownames(leafHelcom) <- gsub("leaf.*","",rownames(leafHelcom)) ## fix sample names
leafHelcom <- data.frame(leafHelcom[order(as.numeric(rownames(leafHelcom))),]) ## order
leafHelcom[leafHelcom > 0] <- 1 ## P/A
leafHelcom.hel <- decostand(leafHelcom,'hellinger') ## transform
leafHel.pcnm.rda <- rda(leafHelcom.hel, ptd.PCNM) ## rda of our leaf community by the PTD
leafHel.pcnm.sigtest <- anova.cca(leafHel.pcnm.rda) ## significance test
```

```
In [34]: leafHel.pcnm.rda
```

```
Call: rda(X = leafHelcom.hel, Y = ptd.PCNM)
```

	Inertia	Proportion	Rank
Total	0.7807	1.0000	
Constrained	0.4107	0.5261	15
Unconstrained	0.3699	0.4739	15

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2	RDA3	RDA4	RDA5	RDA6	RDA7	RDA8	RDA9	RDA10
0.04984	0.04027	0.03499	0.03224	0.03101	0.03010	0.02724	0.02619	0.02482	0.02293
RDA11	RDA12	RDA13	RDA14	RDA15					
0.02184	0.01989	0.01884	0.01756	0.01296					

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
0.03856	0.03579	0.03307	0.03042	0.02825	0.02693	0.02518	0.02332	0.02271	0.02138
PC11	PC12	PC13	PC14	PC15					
0.02026	0.01761	0.01727	0.01598	0.01321					

```
In [35]: leafHel.pcnm.sigtest
```

	Df	Variance	F	Pr(>F)
Model	15	0.4107370	1.110306	0.024
Residual	15	0.3699312	NA	NA

Huh, so lots of variance is described by these eigenvectors. But when we try to home in on which are the most important, we lose most of our variance-explained:

```
In [31]: mod0 <- rda(leafHelcom.hel ~ 1, ptd.PCNM)
      mod1 <- rda(leafHelcom.hel ~ ., ptd.PCNM)
      step.res <- ordiR2step(mod0, mod1, perm.max = 1000)
```

Step: R2.adj= 0

Call: leafHelcom.hel ~ 1

	R2.adjusted
<All variables>	0.0522703476
+ X1	0.0229218580
+ X15	0.0084925083
+ X6	0.0060445539
+ X11	0.0046342416
+ X14	0.0034423569
+ X8	0.0019905930
+ X2	0.0009470103
+ X13	0.0008225454
+ X3	0.0004068255
<none>	0.0000000000
+ X10	-0.0006274055
+ X9	-0.0010631516
+ X4	-0.0012739732
+ X5	-0.0017769749
+ X7	-0.0047112587
+ X12	-0.0132133424

Df	AIC	F	Pr(>F)
+ X1	1	-6.462	1.7038 0.002 **

---

Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1 1

Step: R2.adj= 0.02292186

Call: leafHelcom.hel ~ X1

	R2.adjusted
<All variables>	0.05227035
+ X15	0.03253631
+ X6	0.03000093
+ X11	0.02854025
+ X14	0.02730579
+ X8	0.02580218
+ X2	0.02472133
+ X13	0.02459242
+ X3	0.02416185
+ X10	0.02309068
<none>	0.02292186
+ X9	0.02263937
+ X4	0.02242102

```

+ X5          0.02190006
+ X7          0.01886098
+ X12         0.01005525

      Df      AIC      F Pr(>F)
+ X15   1 -5.8564 1.2882  0.01 **

---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

```

Step: R2.adj= 0.03253631  
Call: leafHelcom.hel ~ X1 + X15

	R2.adjusted
<All variables>	0.05227035
+ X6	0.04023366
+ X11	0.03871888
+ X14	0.03743870
+ X8	0.03587940
+ X2	0.03475852
+ X13	0.03462483
+ X3	0.03417832
+ X10	0.03306748
+ X9	0.03259945
<none>	0.03253631
+ X4	0.03237301
+ X5	0.03183275
+ X7	0.02868112
+ X12	0.01954925

```

      Df      AIC      F Pr(>F)
+ X6   1 -5.2314 1.2246  0.042 *

---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

```

Step: R2.adj= 0.04023366  
Call: leafHelcom.hel ~ X1 + X15 + X6

	R2.adjusted
<All variables>	0.05227035
+ X11	0.04695007
+ X14	0.04562066
+ X8	0.04400138
+ X2	0.04283738
+ X13	0.04269856
+ X3	0.04223487
+ X10	0.04108130
+ X9	0.04059528
+ X4	0.04036013

```

<none>          0.04023366
+ X5            0.03979909
+ X7            0.03652624
+ X12           0.02704314

      Df      AIC      F Pr(>F)
+ X11   1 -4.6191 1.1903  0.086 .
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

```

In [32]: step.res

```
Call: rda(formula = leafHelcom.hel ~ X1 + X15 + X6, data = ptd.PCNM)
```

	Inertia	Proportion	Rank
Total	0.7807	1.0000	
Constrained	0.1063	0.1362	3
Unconstrained	0.6743	0.8638	27

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2	RDA3
0.04525	0.03377	0.02731

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.04709	0.04639	0.03775	0.03550	0.03407	0.03251	0.03126	0.03052

(Showed only 8 of all 27 unconstrained eigenvalues)

In [33]: step.res\$anova

	R2.adj	Df	AIC	F	Pr(>F)
+ X1	0.02292186	1	-6.462034	1.703788	0.002
+ X15	0.03253631	1	-5.856416	1.288196	0.010
+ X6	0.04023366	1	-5.231442	1.224561	0.042
<All variables>	0.05227035	NA	NA	NA	NA

In [36]: attributes(step.res\$terms)\$term.labels

1. 'X1' 2. 'X15' 3. 'X6'

In [37]: sigPCNM <- attributes(step.res\$terms)\$term.labels  
sigPCNM <- sigPCNM[c(1,3,2)]

In [40]: leafHelPCNM <- ptd.PCNM[,sigPCNM]  
#save(leafHelPCNM, file='leafHelPCNM.rda')

```
In [42]: head(leafHelPCNM)
```

	X1	X6	X15
3	-281.8947	-4.309998	-0.061690925
6	-281.9868	-4.337738	0.047993390
8	-282.3043	-5.346045	0.007915014
14	-251.4270	163.354497	0.005589706
25	-343.1403	-57.486706	0.082847775
28	-342.8798	-56.991101	-0.082648360

```
In [45]: leafHel.pcnm.rda2 <- rda(leafHelcom.hel ~ ., leafHelPCNM)
axes.test <- anova.cca(leafHel.pcnm.rda2, by='axis') ## takes a minute
```

```
In [46]: leafHel.pcnm.rda2
```

```
Call: rda(formula = leafHelcom.hel ~ X1 + X6 + X15, data = leafHelPCNM)
```

	Inertia	Proportion	Rank
Total	0.7807	1.0000	
Constrained	0.1063	0.1362	3
Unconstrained	0.6743	0.8638	27

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2	RDA3
0.04525	0.03377	0.02731

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.04709	0.04639	0.03775	0.03550	0.03407	0.03251	0.03126	0.03052

(Showed only 8 of all 27 unconstrained eigenvalues)

```
In [50]: leafHel.pcnm.sigtest2 <- anova.cca(leafHel.pcnm.rda2) ## significance test
leafHel.pcnm.sigtest2
```

	Df	Variance	F	Pr(>F)
Model	3	0.1063351	1.419203	0.001
Residual	27	0.6743332	NA	NA

Spatial patterns explain 13% of the variance.

```
In [47]: axes.test
```

	Df	Variance	F	Pr(>F)
RDA1	1	0.04525430	1.811962	0.001
RDA2	1	0.03376712	1.352021	0.017
RDA3	1	0.02731363	1.093625	0.204
Residual	27	0.67433323	NA	NA

Visualize:

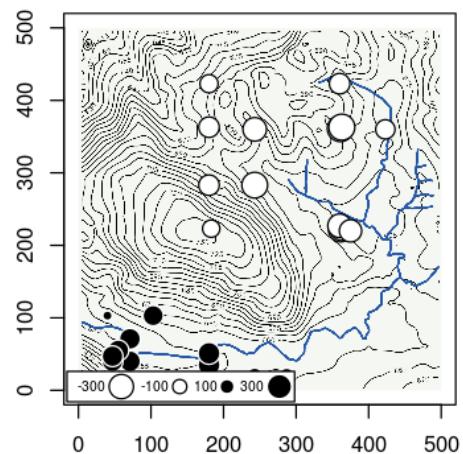
```
In [51]: source("/home/daniel/Documents/taiwan/taiwan_combined_stats/CSrev/NEWR-2ed_code_data/NE
```

Make a function for viewing these.

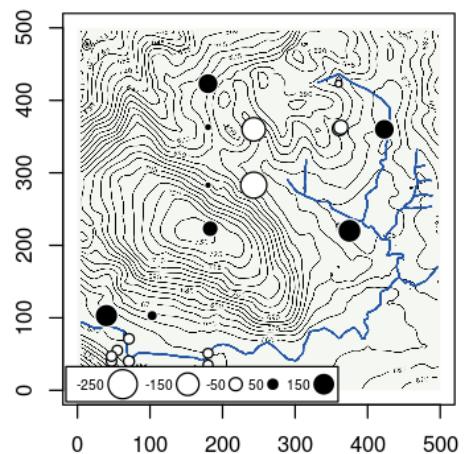
```
In [60]: mapP <- function(PCNM, P, bkg){  
    require('png')  
    topo <- readPNG(bkg) ## load scanned image of Fushan Map  
    plot(1, type='n',  
        xlim=c(0,500),  
        ylim=c(0,500),  
        xlab = ' ',  
        ylab = ' ',  
        main = paste('PCNM', colnames(PCNM)[P], sep = ' ')  
    ) ##blank plot  
    rasterImage(topo,0,0,500,500)  
    sr.value(dfxy=leafHelxy,  
    z=PCNM[,P],  
    # clegend = 0, ## gets rid of legends, they can get in the way  
    add.plot = TRUE,  
    )  
}
```

```
In [61]: options(repr.plot.height = 12)  
par(pty="s")  
par(mfrow=c(2,2))  
for (i in 1:ncol(leafHelPCNM)){mapP(leafHelPCNM,i, '/home/daniel/Documents/taiwan/taiwan  
par(mfrow=c(1,1))
```

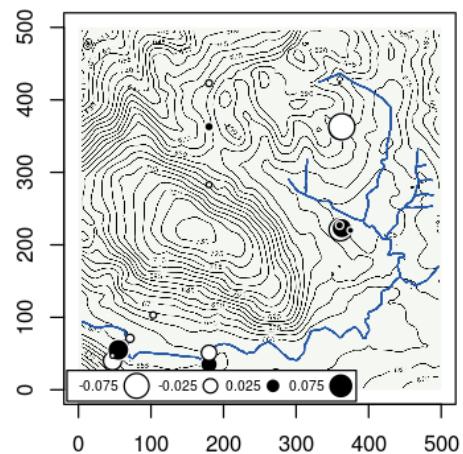
**PCNM X1**



**PCNM X6**



**PCNM X15**



Interesting, X1 looks familiar - the signature of the southwest valley on the microbiome is pretty clear.

Check environmental correlations

```
In [114]: load('deseq95.rda')
           load('leafHelPCNM.rda')
           leafHel95 <- subset_samples(deseq95, Library == 'L' & Host_genus_species == "Helicia_f")

In [115]: leafHel.env <- sample_data(leafHel95)[,c('Forest_Type', 'vegcom')] ## fix weird rownames
           rownames(leafHel.env) <- gsub("leaf.*", "", rownames(leafHel.env))

In [116]: leafHel.env <- data.frame(leafHel.env[order(as.numeric(rownames(leafHel.env))),])
           leafHel.env$Forest_Type <- as.factor(leafHel.env$Forest_Type)
           leafHel.env$vegcom <- as.factor(leafHel.env$vegcom)

           all(rownames(leafHel.env) == rownames(leafHelPCNM)) ## check order of rows match
```

TRUE

```
In [117]: head(leafHelPCNM)
```

	X1	X6	X15
3	-281.8947	-4.309998	-0.061690925
6	-281.9868	-4.337738	0.047993390
8	-282.3043	-5.346045	0.007915014
14	-251.4270	163.354497	0.005589706
25	-343.1403	-57.486706	0.082847775
28	-342.8798	-56.991101	-0.082648360

```
In [118]: leafHelPCNM.X1 <- lm(leafHelPCNM[,1] ~ ., data=leafHel.env)
           leafHelPCNM.X6 <- lm(leafHelPCNM[,2] ~ ., data=leafHel.env)
           leafHelPCNM.X15 <- lm(leafHelPCNM[,3] ~ ., data=leafHel.env)
```

```
In [120]: summary(leafHelPCNM.X1)
```

Call:

```
lm(formula = leafHelPCNM[, 1] ~ ., data = leafHel.env)
```

Residuals:

Min	1Q	Median	3Q	Max
-502.99	-122.26	41.35	75.46	362.01

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-151.21	196.62	-0.769	0.449
Forest_Type4	353.52	313.46	1.128	0.271

Forest_Type5	-74.47	344.92	-0.216	0.831
Forest_Type6	160.47	313.46	0.512	0.613
Forest_Type7	373.72	278.06	1.344	0.192
vegcom2	-341.83	204.09	-1.675	0.107
vegcom3	-54.89	204.09	-0.269	0.790
vegcom4	NA	NA	NA	NA

Residual standard error: 196.6 on 24 degrees of freedom  
Multiple R-squared: 0.467, Adjusted R-squared: 0.3338  
F-statistic: 3.505 on 6 and 24 DF, p-value: 0.01241

In [121]: `summary(leafHelPCNM.X6)`

Call:  
`lm(formula = leafHelPCNM[, 2] ~ ., data = leafHel.env)`

Residuals:

Min	1Q	Median	3Q	Max
-183.74	-20.50	-10.88	3.32	172.85

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	79.56	75.62	1.052	0.3032
Forest_Type4	-118.49	120.55	-0.983	0.3355
Forest_Type5	-256.67	132.65	-1.935	0.0649
Forest_Type6	-39.03	120.55	-0.324	0.7489
Forest_Type7	-111.21	106.94	-1.040	0.3087
vegcom2	47.31	78.49	0.603	0.5523
vegcom3	14.89	78.49	0.190	0.8511
vegcom4	NA	NA	NA	NA

---  
Signif. codes: 0 \*\*\* 0.001 \*\* 0.01 \* 0.05 . 0.1 1

Residual standard error: 75.62 on 24 degrees of freedom  
Multiple R-squared: 0.2725, Adjusted R-squared: 0.0906  
F-statistic: 1.498 on 6 and 24 DF, p-value: 0.2212

In [122]: `summary(leafHelPCNM.X15)`

Call:  
`lm(formula = leafHelPCNM[, 3] ~ ., data = leafHel.env)`

Residuals:

```

      Min       1Q    Median       3Q      Max
-0.084290 -0.007726 -0.001618  0.007095  0.081206

```

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	1.515e-07	3.328e-02	0.000	1.000
Forest_Type4	-3.242e-02	5.306e-02	-0.611	0.547
Forest_Type5	-1.055e-02	5.839e-02	-0.181	0.858
Forest_Type6	-8.143e-03	5.306e-02	-0.153	0.879
Forest_Type7	-8.171e-03	4.707e-02	-0.174	0.864
vegcom2	9.813e-03	3.455e-02	0.284	0.779
vegcom3	1.057e-02	3.455e-02	0.306	0.762
vegcom4	NA	NA	NA	NA

Residual standard error: 0.03328 on 24 degrees of freedom

Multiple R-squared: 0.042, Adjusted R-squared: -0.1975

F-statistic: 0.1753 on 6 and 24 DF, p-value: 0.981

As before, the environmental data I have available does not explain the spatial patterns of the fungal community well (or at all, really).

Helicia wood dbMEMs

```
In [72]: woodHelxy <- sample_data(woodHel95)[,c('X', 'Y')]
rownames(woodHelxy) <- gsub("w","",rownames(woodHelxy))
woodHelxy <- data.frame(woodHelxy[order(as.numeric(rownames(woodHelxy))),])
woodHelxy$X <- as.numeric(woodHelxy$X)
woodHelxy$Y <- as.numeric(woodHelxy$Y)
#save(woodHelxy, file = 'woodHelxy.rda')
ptd <- dist(woodHelxy)
span.ptd <- spantree(ptd)
dmin <- max(span.ptd$dist) ## 144.2 m
ptd[ptd > dmin] <- 4*dmin
ptd.PCoA <- cmdscale(ptd, k=nrow(woodHelxy)-1, eig = TRUE)
nb.ev <- length(which(ptd.PCoA$eig > 0.0000001)) ## keep only the positive eigenvalues
ptd.PCNM <- data.frame(ptd.PCoA$points[1:nrow(woodHelxy), 1:nb.ev])
woodHelcom <- t(otu_table(woodHel95)) ## get community matrix out of phyloseq
```

Warning message in cmdscale(ptd, k = nrow(woodHelxy) - 1, eig = TRUE):  
only 11 of the first 21 eigenvalues are > 0

```
In [73]: head(ptd.PCNM)
```

	X1	X2	X3	X4	X5	X6	X7	X8
3	-196.7016	62.66867	362.657234	95.26081	-32.725328	-53.00397	-23.8080072	-116.18070
14	-152.8447	-11.97322	359.352839	108.55180	-50.059437	-42.78072	0.6207213	165.19603
18	-219.9646	175.69427	8.178574	-100.44715	82.537082	305.27466	-60.9284878	27.83972
25	-332.1799	230.96135	8.365351	-27.13627	34.333876	-13.03012	-19.1974755	-139.60282
28	-306.2121	210.87138	-134.755864	-73.92509	50.974990	28.89790	11.1393885	94.42761
38	-275.6730	152.36410	-137.183210	-24.02992	8.662355	-214.99402	91.8187885	40.06896

```
In [77]: rownames(woodHelcom) <- gsub("w","",rownames(woodHelcom)) ## fix sample names
woodHelcom <- data.frame(woodHelcom[order(as.numeric(rownames(woodHelcom))),]) ## order
woodHelcom[woodHelcom > 0] <- 1 ## P/A
woodHelcom.hel <- decostand(woodHelcom, 'hellinger') ## transform
woodHel.pcnm.rda <- rda(woodHelcom.hel, ptd.PCNM) ## rda of our wood community by the P
woodHel.pcnm.sigtest <- anova.cca(woodHel.pcnm.rda) ## significance test
```

```
In [78]: woodHel.pcnm.rda
woodHel.pcnm.sigtest
```

Call: rda(X = woodHelcom.hel, Y = ptd.PCNM)

	Inertia	Proportion	Rank
Total	0.7998	1.0000	
Constrained	0.3945	0.4933	10
Unconstrained	0.4053	0.5067	11

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2	RDA3	RDA4	RDA5	RDA6	RDA7	RDA8	RDA9	RDA10
0.05923	0.04949	0.04662	0.04014	0.03739	0.03595	0.03432	0.03308	0.03022	0.02806

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8	PC9	PC10
0.05106	0.04644	0.04084	0.03776	0.03733	0.03630	0.03425	0.03227	0.03169	0.02894
PC11									
0.02843									

	Df	Variance	F	Pr(>F)
Model	10	0.3945079	1.070706	0.03
Residual	11	0.4053017	NA	NA

```
In [79]: mod0 <- rda(woodHelcom.hel ~ 1, ptd.PCNM)
mod1 <- rda(woodHelcom.hel ~ ., ptd.PCNM)
step.res <- ordiR2step(mod0, mod1, perm.max = 1000)
```

Step: R2.adj= 0

Call: woodHelcom.hel ~ 1

```

R2.adjusted
<All variables> 0.0325725988
+ X1              0.0197556797
+ X8              0.0086364466
+ X7              0.0027928808
+ X10             0.0017852299
+ X5              0.0005311283
+ X3              0.0004117434
<none>           0.0000000000
+ X9              -0.0024524014
+ X4              -0.0032855687
+ X6              -0.0047362492
+ X2              -0.0055239602

Df      AIC      F Pr(>F)
+ X1   1 -3.4502 1.4232  0.002 **

---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1    1

Step: R2.adj= 0.01975568
Call: woodHelcom.hel ~ X1

R2.adjusted
<All variables> 0.03257260
+ X8              0.02988645
+ X7              0.02373533
+ X10             0.02267464
+ X5              0.02135453
+ X3              0.02122887
<none>           0.01975568
+ X9              0.01821398
+ X4              0.01733696
+ X6              0.01580993
+ X2              0.01498076

Df      AIC      F Pr(>F)
+ X8   1 -2.8072 1.2089  0.014 *

---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1    1

Step: R2.adj= 0.02988645
Call: woodHelcom.hel ~ X1 + X8

R2.adjusted
+ X7              0.03465001
+ X10             0.03353040
<All variables> 0.03257260
+ X5              0.03213695

```

```
+ X3          0.03200430
<none>      0.02988645
+ X9          0.02882192
+ X4          0.02789618
+ X6          0.02628431
+ X2          0.02540907
```

In [80]: step.res

```
Call: rda(formula = woodHelcom.hel ~ X1 + X8, data = ptd.PCNM)
```

	Inertia	Proportion	Rank
Total	0.7998	1.0000	
Constrained	0.0978	0.1223	2
Unconstrained	0.7020	0.8777	19

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2
0.05313	0.04466

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.05474	0.05222	0.04607	0.04454	0.04298	0.04084	0.03983	0.03869

(Showed only 8 of all 19 unconstrained eigenvalues)

In [81]: step.res\$anova

	R2.adj	Df	AIC	F	Pr(>F)
+ X1	0.01975568	1	-3.450195	1.423230	0.002
+ X8	0.02988645	1	-2.807199	1.208857	0.014
<All variables>	0.03257260	NA	NA	NA	NA

In [82]: attributes(step.res\$terms)\$term.labels

1. 'X1' 2. 'X8'

In [85]: sigPCNM <- attributes(step.res\$terms)\$term.labels  
woodHelPCNM <- ptd.PCNM[,sigPCNM]  
#save(woodHelPCNM, file='woodHelPCNM.rda')

In [87]: head(woodHelPCNM)

	X1	X8
3	-196.7016	-116.18070
14	-152.8447	165.19603
18	-219.9646	27.83972
25	-332.1799	-139.60282
28	-306.2121	94.42761
38	-275.6730	40.06896

```
In [88]: woodHel.pcnm.rda2 <- rda(woodHelcom.hel ~ ., woodHelPCNM)
axes.test <- anova.cca(woodHel.pcnm.rda2, by='axis') ## takes a minute
```

```
In [89]: woodHel.pcnm.rda2
```

```
Call: rda(formula = woodHelcom.hel ~ X1 + X8, data = woodHelPCNM)
```

	Inertia	Proportion	Rank
Total	0.7998	1.0000	
Constrained	0.0978	0.1223	2
Unconstrained	0.7020	0.8777	19

Inertia is variance

Eigenvalues for constrained axes:

RDA1	RDA2
0.05313	0.04466

Eigenvalues for unconstrained axes:

PC1	PC2	PC3	PC4	PC5	PC6	PC7	PC8
0.05474	0.05222	0.04607	0.04454	0.04298	0.04084	0.03983	0.03869

(Showed only 8 of all 19 unconstrained eigenvalues)

Spatial patterns explain 12% of the variance.

```
In [90]: woodHel.pcnm.sigtest2 <- anova.cca(woodHel.pcnm.rda2) ## significance test
woodHel.pcnm.sigtest2
```

	Df	Variance	F	Pr(>F)
Model	2	0.09779929	1.323475	0.001
Residual	19	0.70201030	NA	NA

```
In [91]: axes.test
```

	Df	Variance	F	Pr(>F)
RDA1	1	0.05313483	1.438101	0.001
RDA2	1	0.04466446	1.208849	0.019
Residual	19	0.70201030	NA	NA

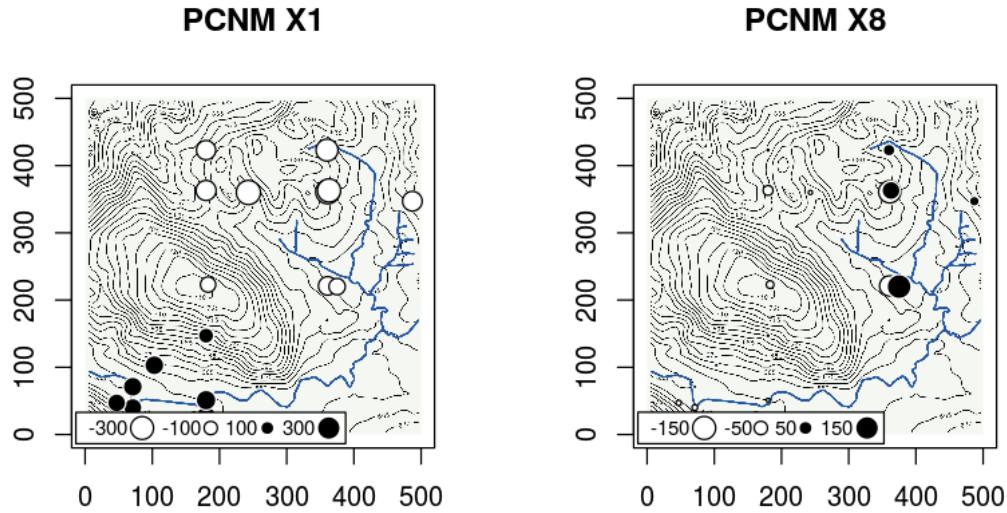
Visualize:

```
In [92]: source("/home/daniel/Documents/taiwan/taiwan_combined_stats/CSrev/NEWR-2ed_code_data/NE
```

Make a function for viewing these.

```
In [98]: mapP <- function(PCNM, P, bkg){
  require('png')
  topo <- readPNG(bkg) ## load scanned image of Fushan Map
  plot(1, type='n',
       xlim=c(0,500),
       ylim=c(0,500),
       xlab = '',
       ylab = '',
       main = paste('PCNM', colnames(PCNM)[P], sep = ' '))
  ) ##blank plot
  rasterImage(topo,0,0,500,500)
  sr.value(dfxy=woodHelxy,
  z=PCNM[,P],
  # legend = 0, ## gets rid of legends, they can get in the way
  add.plot = TRUE,
)
}
```

```
In [101]: #options(repr.plot.height = 8)
par(pty="s")
par(mfrow=c(1,2))
for (i in 1:ncol(woodHelPCNM)){mapP(woodHelPCNM,i, '/home/daniel/Documents/taiwan/taiwa
```



Very similar to leaves.  
Check environmental correlations

```
In [132]: load('deseq95.rda')
load('woodHelPCNM.rda')
woodHel95 <- subset_samples(deseq95, Library == 'W' & Host_genus_species == "Helicia_f"
woodHel.env <- sample_data(woodHel95)[,c('Forest_Type', 'vegcom')] ## fix weird rownames
rownames(woodHel.env) <- gsub("w", "", rownames(woodHel.env))
```

```
In [133]: woodHel.env <- data.frame(woodHel.env[order(as.numeric(rownames(woodHel.env))),])
```

```

woodHel.env$Forest_Type <- as.factor(woodHel.env$Forest_Type)
woodHel.env$vegcom <- as.factor(woodHel.env$vegcom)

all(rownames(woodHel.env) == rownames(woodHelpCNM)) ## check order of rows match

```

TRUE

In [134]: `head(woodHelpCNM)`

	X1	X8
3	-196.7016	-116.18070
14	-152.8447	165.19603
18	-219.9646	27.83972
25	-332.1799	-139.60282
28	-306.2121	94.42761
38	-275.6730	40.06896

In [139]: `woodHelpCNM.X1 <- lm(woodHelpCNM[,1] ~ ., data=woodHel.env)`  
`woodHelpCNM.X8 <- lm(woodHelpCNM[,2] ~ ., data=woodHel.env)`

In [140]: `summary(woodHelpCNM.X1)`

Call:

`lm(formula = woodHelpCNM[, 1] ~ ., data = woodHel.env)`

Residuals:

Min	1Q	Median	3Q	Max
-370.77	-97.97	18.94	75.90	306.75

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-133.92	193.94	-0.691	0.500
Forest_Type3	560.82	342.46	1.638	0.122
Forest_Type5	-141.07	342.46	-0.412	0.686
Forest_Type6	88.32	310.46	0.284	0.780
Forest_Type7	343.77	274.28	1.253	0.229
vegcom2	-290.38	205.07	-1.416	0.177
vegcom3	-59.04	205.07	-0.288	0.777
vegcom4	NA	NA	NA	NA

Residual standard error: 193.9 on 15 degrees of freedom

Multiple R-squared: 0.4936, Adjusted R-squared: 0.291

F-statistic: 2.436 on 6 and 15 DF, p-value: 0.07581

In [141]: `summary(woodHelpCNM.X8)`

```
Call:
lm(formula = woodHelPCNM[, 2] ~ ., data = woodHel.env)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-140.806	-7.093	-5.422	2.439	163.993

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	-16.745	69.489	-0.241	0.813
Forest_Type3	27.475	122.703	0.224	0.826
Forest_Type5	14.403	122.703	0.117	0.908
Forest_Type6	10.123	111.236	0.091	0.929
Forest_Type7	21.899	98.272	0.223	0.827
vegcom2	-3.951	73.476	-0.054	0.958
vegcom3	-2.651	73.476	-0.036	0.972
vegcom4	NA	NA	NA	NA

Residual standard error: 69.49 on 15 degrees of freedom

Multiple R-squared: 0.008722, Adjusted R-squared: -0.3878

F-statistic: 0.022 on 6 and 15 DF, p-value: 0.9999

Nope.

Comparisons of Helicia communities to the SW valley

Helicia leaf community comparison to SW valley

```
In [176]: load('deseq95.rda')
           load("leafHelxy.rda")
           leafHel95 <- subset_samples(deseq95, Library == 'L' & Host_genus_species == "Helicia_f"
           woodHel95 <- subset_samples(deseq95, Library == 'W' & Host_genus_species == "Helicia_f")
```

```
In [166]: ## sample 104, at 180,51.
```

```
## leaves:
bb <- t(otu_table(leafHel95)) ## get otu table for Helicia leaves
bb[bb > 0] <- 1 ## presence/absence
cc <- vegdist(bb, method='bray') ## makes a triangular association matrix
dd <- as.matrix(cc) ## convert to full, symmetric matrix
#leafhelvalBC <- dd[, '131leaf'] ## extract only the comparisons to our Helicia sample
leafhelvalBC <- dd[, '104leaf'] ## extract only the comparisons to our Helicia sample
```

```
In [167]: all(names(leafhelvalBC) == rownames(sample_data(leafHel95))) ## worked
## map a dataframe with the info we want to plot:
mapBC <- cbind(sample_data(leafHel95)[,c('X','Y')], leafhelvalBC)
colnames(mapBC)[3] <- 'BC'
## order by BC dissimilarity
```

```

mapBC <- mapBC[order(mapBC$BC),]
mapBCrev <- mapBC[rev(rownames(mapBC)),]
## make a heat map palette useing colorbrewer
my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)
## need an "adapter", for our BC values to this heat map color scheme:
BCroundup <- rev(round(mapBC$BC*100+1))

```

TRUE

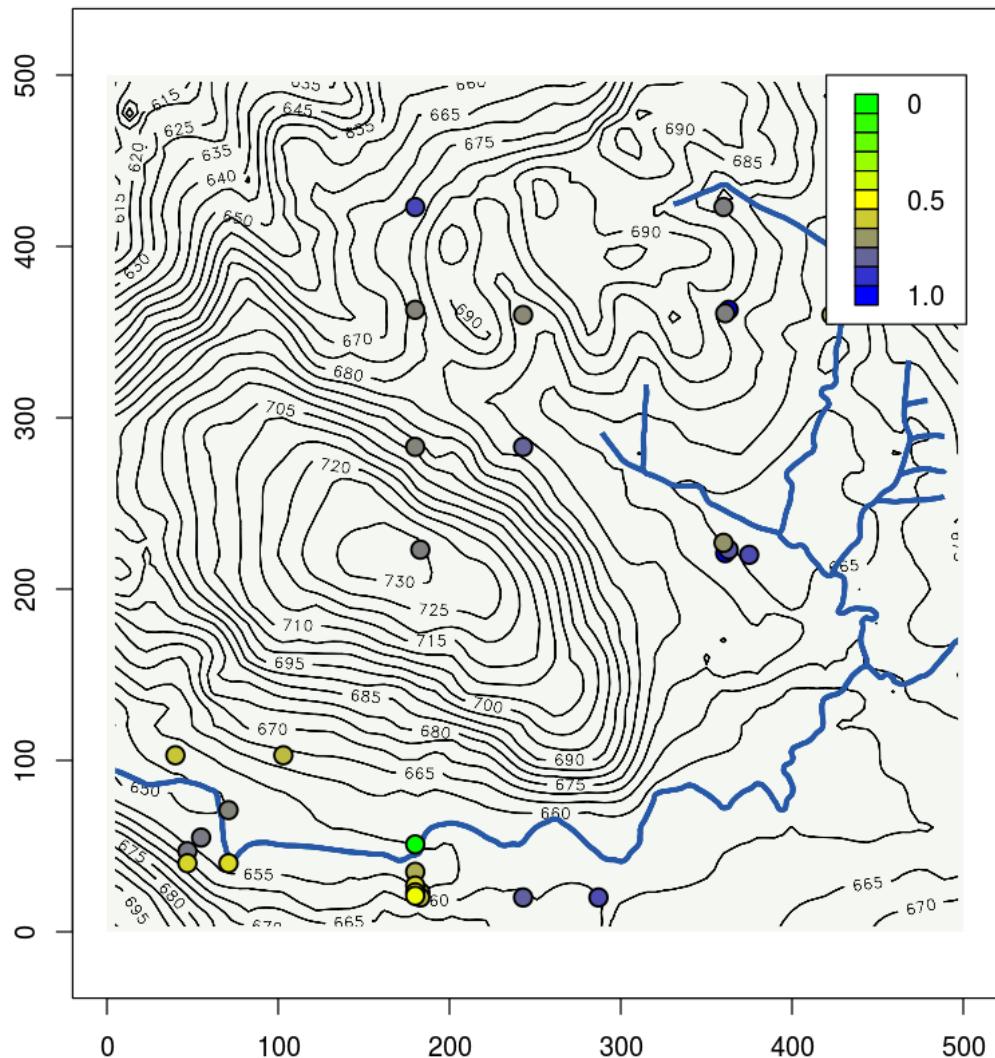
```

In [174]: ## leaves
#svg('helleafBCvalley.svg')
plot(1, type='n',
      xlim=c(0,500),
      ylim=c(0,500),
      xlab = '',
      ylab = '',
      asp=1,
      main='Helicia formosana leaf endophytes, comparison to one valley point'
) ##blank plot
topo <- readPNG('/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png')
rasterImage(topo,0,0,500,500) ## add raster of our plot
## add Helicia points, colored by similarity
points(mapBCrev[,c('X','Y')],
      pch=21,
      cex=1.5,
      bg = my_palette2[BCroundup],
      lwd=1.5,
      )
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'
legend(
#           "topright",
x=420,
y=500,
fill = my_palette2[seq(1,101,10)],
legend = heatlegend,
bg = "white",
y.intersp = 0.5,
)
## if we want a circle around the comparison point:
#points(mapBCrev['104leaf',c('X','Y')],
#      pch=1,
#      cex=3,
#      col='red',
#      lwd = 3,

```

```
#       )
#dev.off()
```

### Helicia formosana leaf endophytes, comparison to one valley point



Green point point is the center of comparison.  
Helicia wood community comparison to SW valley

In [180]: `## sample 104, at 180, 51.`

```
bb <- t(otu_table(woodHel95)) ## get otu table for Helicia wood
bb[bb > 0] <- 1 ## presence/absence
```

```

cc <- vegdist(bb, method='bray') ## makes a triangular association matrix
dd <- as.matrix(cc) ## convert to full, symmetric matrix
#woodhelvalBC <- dd[, '131wood'] ## extract only the comparisons to our Helicia sample
woodhelvalBC <- dd[, '104w'] ## extract only the comparisons to our Helicia sample of c

```

```

In [181]: all(names(woodhelvalBC) == rownames(sample_data(woodHel95))) ## worked
## map a dataframe with the info we want to plot:
mapBC <- cbind(sample_data(woodHel95)[,c('X','Y')], woodhelvalBC)
colnames(mapBC)[3] <- 'BC'
## order by BC dissimilarity
mapBC <- mapBC[order(mapBC$BC),]
mapBCrev <- mapBC[rev(rownames(mapBC)),]
## make a heat map palette useing colorbrewer
my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)
## need an "adapter", for our BC values to this heat map color scheme:
BCroundup <- rev(round(mapBC$BC*100+1))

```

TRUE

```

In [182]: ## wood
#svg('helwoodBCvalley.svg')
plot(1, type='n',
      xlim=c(0,500),
      ylim=c(0,500),
      xlab = '',
      ylab = '',
      asp=1,
      main='Helicia formosana wood endophytes, comparison to one valley point'
) ##blank plot
topo <- readPNG('/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png')
rasterImage(topo,0,0,500,500) ## add raster of our plot
## add Helicia points, colored by similarity
points(mapBCrev[,c('X','Y')],
      pch=21,
      cex=1.5,
      bg = my_palette2[BCroundup],
      lwd=1.5,
      )
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'
legend(
#          "topright",
x=420,
y=500,
fill = my_palette2[seq(1,101,10)],

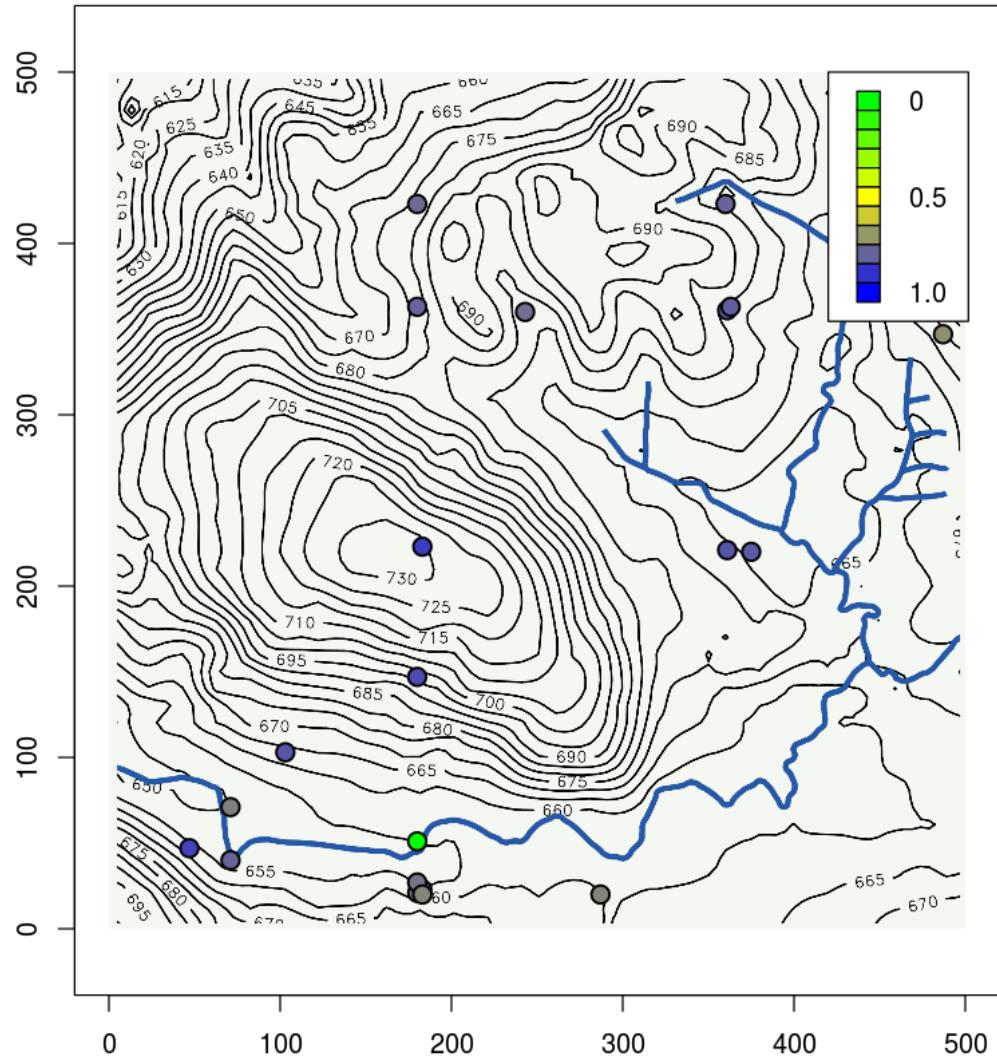
```

```

        legend = heatlegend,
        bg = "white",
        y.intersp = 0.5,
    )
#dev.off()

```

### **Helicia formosana wood endophytes, comparison to one valley point**



As before, the leaf community seems to be similar to itself in the valley.  
Helicia core mycobiome

From our cooccurrence analysis, we find the following OTUs as "core":

```
In [209]: load('deseq95.rda')
    load(file= 'helleaffung.rda') ## OTUs from cooccurrence analysis
    load(file= 'helwoodfung.rda') ## OTUs from cooccurrence analysis
```

```
In [212]: print("leaf OTUs"); helleaffung
    print("wood OTUs"); helwoodfung
```

```
[1] "leaf OTUs"
```

1. 'OTU19:100leaf' 2. 'OTU202:100leaf' 3. 'OTU10:100leaf' 4. 'OTU17:100leaf'  
 5. 'OTU15:100leaf' 6. 'OTU27:100leaf' 7. 'OTU94:114leaf' 8. 'OTU91:100leaf' 9. 'OTU239:100leaf'  
 10. 'OTU16:100leaf' 11. 'OTU199:101leaf' 12. 'OTU678:131leaf'

```
[1] "wood OTUs"
```

1. 'OTU352:1w' 2. 'OTU726:72w' 3. 'OTU269:1w' 4. 'OTU287:3w' 5. 'OTU250:4w'  
 6. 'OTU257:3w' 7. 'OTU84:38w'

We have preliminary IDs on these from the high-throughput taxonomic assignment step in our biom table construction. I don't trust them much, but they are worth looking at:

Leaf endophytes:

```
In [196]: tax_table(deseq95)[helleaffung,]
```

	Kingdom	Phylum	Class	Order
OTU19:100leaf	Fungi	Ascomycota	Eurotiomycetes	Onygenales
OTU202:100leaf	Fungi	Ascomycota	Dothideomycetes	Dothideomycetidae
OTU10:100leaf	Fungi	Ascomycota	Lecanoromycetes	NA
OTU17:100leaf	Fungi	Ascomycota	Dothideomycetes	Pleosporales
OTU15:100leaf	Fungi	Ascomycota	Lecanoromycetes	NA
OTU27:100leaf	Fungi	Ascomycota	Sordariomycetes	Xylariales
OTU94:114leaf	Fungi	Basidiomycota	Microbotryomycetes	Sporidiobolales
OTU91:100leaf	Fungi	Glomeromycota	Glomeromycetes	Glomerales
OTU239:100leaf	Fungi	Ascomycota	Dothideomycetes	Capnodiales
OTU16:100leaf	Fungi	Ascomycota	Sordariomycetes	Sordariales
OTU199:101leaf	Fungi	Chytridiomycota	NA	NA
OTU678:131leaf	Fungi	Ascomycota	Pezizomycotina_cls_Incertae_sedis	Pezizomycotina_order

Yeah, not sure what to think about a glomeromycete in my leaf samples. Manual blasting to follow.

Wood endophytes:

```
In [207]: tax_table(deseq95)[helwoodfung,]
```

	Kingdom	Phylum	Class	Order	Family
OTU352:1w	Fungi	Ascomycota	Eurotiomycetes	Chaetothyriales	NA
OTU726:72w	Fungi	Ascomycota	Dothideomycetes	Capnodiales	NA
OTU269:1w	Fungi	Basidiomycota	Tremellomycetes	Tremellales	Tremellales_fam_Incertae
OTU287:3w	Fungi	NA	NA	NA	NA
OTU250:4w	Fungi	NA	NA	NA	NA
OTU257:3w	Fungi	Ascomycota	Leotiomycetes	Helotiales	Helotiales_fam_Incertae
OTU84:38w	Fungi	Ascomycota	Eurotiomycetes	Chaetothyriales	Herpotrichiellaceae

I have very little confidence in these assignments. Let's reblast them all manually. The pipeline for this is similar to the methods above for identifying intended/index-bleed/contaminants in the controls:

```
In [ ]: ## make a blastable database from our local copy of UNITE:  
cp sh_general_release_dynamic_01.12.2017.fasta unite.copy.fasta  
chmod 777 unite.copy.fasta  
grep "€" unite.copy.fasta  
## the cross/hybrid symbol is causing problems for BLAST:  
sed -i '/_€/ s/€//g' unite.copy.fasta  
  
makeblastdb -in unite.copy.fasta -dbtype nucl  
## -parse_seqids not used, avoids issues with complicated  
## sequence headers.
```

Once we have a searchable UNITE database, use it to show us the closest matches for our core fungi:

Leaf manual taxonomy assignments

```
In [ ]: ## still in R:  
load(file= 'helleaffung.rda')  
sink("leafCore.txt")  
helleaffung  
sink()  
## do some quick editing in vim to make a tuple out of this for our python script.
```

Use a python script to search our OTU sequence file for the sequences that correspond to our leaf core mycobiome:

```
In [ ]: #!/usr/bin/env python3  
  
## leafCoreFindseq.py  
  
coreSeq=("OTU19:", "OTU202:", "OTU10:", "OTU17:",  
"OTU15:", "OTU27:", "OTU94:", "OTU91:",  
"OTU239:", "OTU16:", "OTU199:", "OTU678:")  
  
with open('otus_95_combo_nolb.fasta', 'r') as zoop:  
    refseq = zoop.readlines()  
  
with open('seqs_leafCoreMycobiom.fasta', 'w') as goop:  
    for j, otu in enumerate(coreSeq):  
        for i, line in enumerate(refseq):  
            if otu in line:  
                goop.write(line)  
                goop.write(refseq[i+1])
```

```
In [4]: ## BASH, do the blast
UNITE='/home/daniel/Documents/taiwan/UNITE/unite.copy.fasta'
leafCoreSeqs='/home/daniel/Documents/taiwan/taiwan_combined_stats/CSrev/seqs_leafCoreMyc
blastn -query $leafCoreSeqs -db $UNITE -out leafCoreMycobiom_blast.txt -num_descriptions
```

Look at the results:

```
In [1]: cat leafCoreMycobiom_blast.txt
```

BLASTN 2.2.31+

Reference: Zheng Zhang, Scott Schwartz, Lukas Wagner, and Webb Miller (2000), "A greedy algorithm for aligning DNA sequences", J Comput Biol 2000; 7(1-2):203-14.

Database: unite.copy.fasta  
 30,695 sequences; 16,853,603 total letters

Query= OTU19:leafNotChim\_100;size=88958;

Length=181

Sequences producing significant alignments:	Score (Bits)	E Value
Metarhizium_granulomatis HM195305 SH192590.07FU refs k_Fungi;p...	167	1e-41
Ascomycota_sp EF495097 SH192589.07FU reps k_Fungi;p_Ascomycot...	159	2e-39
Metarhizium_novozealandicum DQ385622 SH192584.07FU reps k_Fung...	154	9e-38
Clavicipitaceae_sp EU553292 SH195497.07FU reps k_Fungi;p_Asc...	147	1e-35
Metarhizium_flavoviride AY624203 SH192592.07FU reps k_Fungi;p...	143	2e-34
Hypomyces_sp KC122874 SH209836.07FU reps k_Fungi;p_Ascomycota...	137	9e-33
Metarhizium_sp AB700550 SH192598.07FU reps k_Fungi;p_Ascomycot...	134	1e-31
Ascomycota_sp JN559403 SH195504.07FU reps k_Fungi;p_Ascomycot...	132	4e-31
Hypocreales_sp EU553283 SH195500.07FU reps k_Fungi;p_Ascomyc...	119	3e-27
Metarhizium_viride AY624197 SH192582.07FU refs k_Fungi;p_Asc...	117	1e-26

> Metarhizium\_granulomatis|HM195305|SH192590.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Sordariomycetes  
 Length=516

Score = 167 bits (90), Expect = 1e-41  
 Identities = 132/150 (88%), Gaps = 11/150 (7%)  
 Strand=Plus/Minus

Query	24	AAAATACAGAAGAGTTAGGTGTCCTCCGGCGGGCGCCTGGTCCGGCGCGG-CGTT-T	81
Sbjct	141	AAAATAC--AAGAGTTA-G-GTCCCCGGCGGCCTGGTCCGGC-CGGCCCTGC	87
Query	82	GGCCCGCGATTCCGGGGCGAAAAACCCGCCGAAGCAACGGTAAAGGTATAAGTCACAGT	141
Sbjct	86	GGGCCTG-TTCCGGGGCG-AAGACCCGCCGAAGCAACGGTAAAGGTATAAGTCACAG-	30
Query	142	GGGTTTGGGAGTTGTAAACTCGGTAAATGA	171
Sbjct	29	GGG-TTGGGAGTTGTAAACTCGGTAAATGA	1

> Ascomycota\_sp|EF495097|SH192589.07FU|reps|k\_Fungi;p\_Ascomycota;c\_unidentified;o\_unidentified  
Length=523

Score = 159 bits (86), Expect = 2e-39  
Identities = 148/176 (84%), Gaps = 12/176 (7%)  
Strand=Plus/Minus

Query	7	CACTCAGACAAG--CATATAAAATACAGAAGAGTTAGGTGTCCTCCGGCGGGCGCCTGG	64
Sbjct	167	CACTCAGACATGTAAATGTAAAAAATACAAGAGTTAGGT--CCCCCGGGCGGCCTGG	110
Query	65	TTCCGGCGCGGGCGTTGGCGCGATTCCGGGGCGAAAAACCCGCCGAAGCAACGGTAAA	124
Sbjct	109	TTCCGGGC--AG-G-AAGAACCTG-TTCCGGGGCG-AGAACCCGCCGAAGCAACGGTAAA	56
Query	125	AGGTATAAGTCACAGTGGTTTGGAGTTGTAAACTCGGTAAATGATCCCTCCGC	180
Sbjct	55	AGGTATAAGTCACAG-GGG-TTGGGAGTTGTAAACTCGGTAAATGATCCTCCGC	2

> Metarhizium\_novozealandicum|DQ385622|SH192584.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Sordariomycetidae  
Length=543

Score = 154 bits (83), Expect = 9e-38  
Identities = 148/177 (84%), Gaps = 14/177 (8%)  
Strand=Plus/Minus

Query	7	CACTCAGACAAGCA-TA-TAAAATACAGAAGAGTTAGGTGTCCTCCGGCGGGCGCCTGG	64
Sbjct	167	CACTCAGACATGCAAAAGTAAAAAATACAAGAGTT-TG-GTCCCCCGGGCGCCTGG	110
Query	65	TTCCGGCGCGGGCGTTG-G-CCCGGATTCCGGGGCGAAAAACCCGCCGAAGCAACGGTA	122
Sbjct	109	TTCCGGGC-CGGCTCTCGCAACCTG-TTCCGGGGCG--GACCCGCCGAAGCAACAGTA	55

Query 123 AAAGGTATAAGTCACAGTGGGTTTGGGAGTTGTAAACTCGGTAAATGATCCCTCCG 179  
|| ||||||| |||| | || || || || || || || || || || || || || || || || || ||  
Sbjct 54 AA-GGTATAAGTCACAG-GGG-TTGAGTTGTAAACTCGGTAAATGATCCCTCCG 1

> Clavicipitaceae\_sp|EU553292|SH195497.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Sordariomycetes;o\_Hy  
Length=533

Score = 147 bits (79), Expect = 1e-35  
Identities = 139/165 (84%), Gaps = 16/165 (10%)  
Strand=Plus/Minus

Query 24 AAAATACAGAAGAGTTAGGTGTCCCTCCGGCGGCCCTGGTCCGGG--CGC--GG-CG 78  
|| ||||||| |||| | || || || || || || || || || || || || || || || || |  
Sbjct 159 AAAATACAGAAGAGTTT--GAGTCCCCGGCGGCCCTGTTCCGGGTGGGCAGGGACC 102

Query 79 T-TTGGCCG-C-GATT--CCGGGGCGAAAAACCCGCCGAAGCAACGGTAAAAGGTATAAG 133  
| |||| | | || || || || || || || || || || || || || || || || || || |  
Sbjct 101 TGACGGCCGCCTGTTCCCCGGGC-AATAACCCGCCGAAGCAACAGTAAA-GGTATAAG 44

Query 134 TTCACAGTGGGTTTGGGAGTTGTAAACTCGGTAAATGATCCCTCC 178  
|| |||| | || || || || || || || || || || || || || || || || || || |  
Sbjct 43 TTCACAG-GGG-TTGAGTTGTAAACTCGGTAAATGATCCCTCC 1

> Metarhizium\_flavoviride|AY624203|SH192592.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Sordariomycetes;  
Length=556

Score = 143 bits (77), Expect = 2e-34  
Identities = 127/149 (85%), Gaps = 12/149 (8%)  
Strand=Plus/Minus

Query 34 AGAGTTA--GGTGTCCCTCCGGCGGCCCTGGTCCGGCGCGCGT-TTG-GCCGCGA 89  
|| |||| | || || || || || || || || || || || || || || || || |  
Sbjct 143 AGAGTTATCGGGGTCCCTCCGGCGGCCCTGGTCCGGTAGGCCAACGAGCC-TG- 86

Query 90 TTCCGGGCGAAAAACCCGCCGAAGCAACGGTAAA--GGTATAAGTCACAGTGGTTT 147  
| |||| | || || || || || || || || || || || || || || || || | |  
Sbjct 85 TCCCAGGGCG--AAACCCGCCGAAGCAACGGTAAAATGGTATAAGTCACAG-GGG-TT 30

Query 148 TGGGAGTTGTAAACTCGGTAAATGATCCCT 176  
|| || || || || || || || || || || || |  
Sbjct 29 TGGGAGTTGTAAACTCGGTAAATGATCCCT 1

> Hypomyces\_sp|KC122874|SH209836.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Sordariomycetes;o\_Hypocre  
Length=526

Score = 137 bits (74), Expect = 9e-33  
Identities = 126/150 (84%), Gaps = 8/150 (5%)  
Strand=Plus/Minus

Query	29	ACAGAAGAGTTAGGTGTCCTCCGGCGGCCCTGGTTCGGGCGCGCGTTGGCCCG 88 
Sbjct	142	ACAGAAGAGTT-GG-GTCCTCCGGCGGGCGCCTGGGTCCGGGGCCGGCGCGGGCCGAG 85
Query	89	ATTCCGGGGCGAAAAACCCGCCGAAGCAACGGTAAAGGTATAAGTTACAGTGGTTTT 148 
Sbjct	84	CGCCCGGGCGTAT--CCCGCCGAGGCAACGG-AGA-GGTA-AGGTTCACAGGGTTT 31
Query	149	GGGAGTTGTAACACTCGGTAAATGATCCCTCC 178 
Sbjct	30	GGGAGTTGTAACACTCGGTAAATGATCCCTCC 1

> Metarhizium\_sp|AB700550|SH192598.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Sordariomycetes;o\_Hypocreales|Length=500

Score = 134 bits (72), Expect = 1e-31  
Identities = 77/79 (97%), Gaps = 1/79 (1%)  
Strand=Plus/Minus

Query	102	AAACCCGCCGAAGCAACGGTAAAAGGTATAAGTTCACAGTGGGTTTGGGAGTTGTAAAC	161
Sbjct	78	AAACCCGCCGAAGCAACGGTAAA-GGTATAAGTTCACAGGGGTTTGGGAGTTGTAAAC	20
Query	162	TCGGTAATGATCCCTCCGC	180
Sbjct	19	TCGGTAATGATCCCTCCGC	1

> Ascomycota\_sp|JN559403|SH195504.07FU|reps|k\_Fungi;p\_Ascomycota;c\_unidentified;o\_unidentified  
Length=538

Score = 132 bits (71), Expect = 4e-31  
Identities = 135/163 (83%), Gaps = 15/163 (9%)  
Strand=Plus/Minus

```

Query 138 CAGTGGGTTTGGGAGTTGTAAACTCGGTAAATGATCCCTCCGC 180
      ||| ||| ||||||||||||||||| ||||||||||||| |
Sbjct  43  CAG-GGG-TTGAGTTGTAAACTCGGTAAATGATCCCTCCGC  3

```

> Hypocreales\_sp|EU553283|SH195500.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Sordariomycetes;o\_Hypocreales  
Length=549

Score = 119 bits (64), Expect = 3e-27  
Identities = 80/87 (92%), Gaps = 3/87 (3%)  
Strand=Plus/Minus

```

Query 92  CGGGGGCGAAAAACCGCCGAAGCAACGGTAAAAGGTATAAGTCACAGTGGGTTTGGG 151
      ||||||| || ||||||||||||||||| |||| | ||||||||||||| ||| |||||
Sbjct  84  CGGGGGC-AATAACCGCCGAAGCAACAGTAA-TGGTATAAGTCACAG-GGGGTTTGGG 28

```

```

Query 152 AGTTGTAAACTCGGTAAATGATCCCTCC 178
      ||||||||||||||||||||| |
Sbjct  27 AGTTGTAAACTCGGTAAATGATCCCTCC  1

```

> Metarhizium\_viride|AY624197|SH192582.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Sordariomycetes;o\_Hypocreales  
Length=551

Score = 117 bits (63), Expect = 1e-26  
Identities = 109/130 (84%), Gaps = 7/130 (5%)  
Strand=Plus/Minus

```

Query 45  GTCCTCCGGCGGGCGCCTGGTCCGGCGCGCGTT-TG-GCCCGATTCCGGGGCGAAA 102
      |||| | ||||||| ||||| | | ||| | ||| | ||||| | |
Sbjct  125 GTCCCCCGGCGGGCGCCTGTTCCGGCAGGCCCTGCGAGCC-TGGCTCCGGGGCCG-A 68

```

```

Query 103 AACCCGCCGAAGCAACGGTAAAAGGTATAAGTCACAGTGGGTTGGAGTTGTAAACT 162
      ||||||||||||||||||||| ||||| | ||||| ||||| |
Sbjct  67 GACCCGCCGAAGCAACGGTAAAAGGTATAAGTCACA-TGG-TTTT-GGAGTTGAAACT 11

```

```

Query 163 CGGTAAATGAT 172
      |||||||||
Sbjct  10 CGGTAAATGAT  1

```

Lambda	K	H
1.33	0.621	1.12

Gapped	K	H
Lambda	0.460	0.850
1.28		

Effective search space used: 2572351767

Query= OTU202:leafNotChim\_100;size=12834;

Length=151

\*\*\*\*\* No hits found \*\*\*\*\*

Lambda	K	H
1.33	0.621	1.12

Gapped

Lambda	K	H
1.28	0.460	0.850

Effective search space used: 2087002377

Query= OTU10:leafNotChim\_100;size=250062;

Length=239

Sequences producing significant alignments:	Score (Bits)	E Value
Phyllosticta_ampelicida HM049170 SH210446.07FU reps k_Fungi;p_...	368	4e-102
Phyllosticta_aloeicola KF154280 SH210440.07FU refs k_Fungi;p_....	333	1e-91
Guignardia_sp EU675682 SH198937.07FU reps k_Fungi;p_Ascomycot...	309	2e-84
Phyllosticta_styracicola JX025040 SH210451.07FU refs k_Fungi;p...	291	9e-79
Phyllosticta_parthenocissi EU683672 SH210443.07FU refs k_Fungi...	283	1e-76
Phyllosticta_ampelicida KC193586 SH210430.07FU refs k_Fungi;p_...	279	2e-75
Phyllosticta_aristolochiicola JX486129 SH107086.07FU refs_singl...	268	4e-72
Phyllosticta_ardisiicola AB454274 SH198936.07FU refs k_Fungi;p...	267	1e-71
Phyllosticta_leucothoicola AB454370 SH091775.07FU refs k_Fungi...	263	2e-70
Phyllosticta_neopyrolae AB454318 SH244990.07FU refs_singleton k...	257	9e-69

> Phyllosticta\_ampelicida|HM049170|SH210446.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;  
Length=585

Score = 368 bits (199), Expect = 4e-102  
Identities = 229/243 (94%), Gaps = 4/243 (2%)  
Strand=Plus/Minus

Query 1	TATATCAGGACTTCACGAAATGATCGCTTGAGTTTGATACTGGCAGGCACCTAGCCGG	60
Sbjct 248	TATATCAGGACTTCACAAAATAATCGCTTGAGTTTGATACTGGCAGGCACCTAGCCGG	189
Query 61	GCGTCCTGGCC-GGTTA-AGGCTGGGGCGCCGCCGCCGGTCGAAACCGGGTCGGCC	118
Sbjct 188	GCGTCCTGGCCCGGTTAGGGCTGGGGCGCCGCCGCCGGTCGAAACCGGGTCGGCC	129
Query 119	CGCCAAGAACATGGTAAGGTACACAAGGGTGGGAAGGG-CTCTTCCGCCGGCTTTT	177
Sbjct 128	CGCCAAGAACATGGTAAGGTACACAAGGGTGAGAAGGGCTCTTCTGCCGGCTTTT	69
Query 178	AGGGCCGGCGTACTCGA-TTACCTTCAAGAGAATTACGTATTCAGTAATGATCCTTCC	236
Sbjct 68	AGGGCCGACGTACTCGAGAGACCTTCAAGAGAATTACGTATTCAGTAATGATCCTTCC	9
Query 237	GCA 239	
Sbjct 8	GCA 6	

> Phyllosticta\_aloeicola|KF154280|SH210440.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;c\_Botryosphaeriales  
Length=529

Score = 333 bits (180), Expect = 1e-91  
 Identities = 210/224 (94%), Gaps = 3/224 (1%)  
 Strand=Plus/Minus

Query 1	TATATCAGGACTTCACGAAATGATCGCTTGAGTTTGATACTGGCAGGCACCTAGCCGG	60
Sbjct 222	TATATCAGGACTTCACAAAATAATCGCTTGAGTTTGATACTGGCAGGCACCTAGCCGG	163
Query 61	GCGTCCTGGCCGGTTAAGGCTGGGGCGCCGCCGCCGGTCGAAACCGGGTCGGCCCG	120
Sbjct 162	GCGTCCTGGCCGAGTAGGGCTGGGGCGCCGCCGCCGGTCGAAACCGGGTCGGCCCG	103
Query 121	CCAAAGAACATGGTAAGGTACACAAGGGTGGGAAGGGCTTTCCGCCGGCTTTAGG	180
Sbjct 102	CCAAAGAACATGGTAAGGTACACAAGGGTGAGAAGGGCTTTCCGCCGGCTTTGAGG	43
Query 181	GCCGGCGTACTCGATTACCTTCAAG-AGAATTACGTATTCAG 223	
Sbjct 42	TCCGACGTACTCGAT--CCTTCAAGAAGAATTACGTATTCAG 1	

> Guignardia\_sp|EU675682|SH198937.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;o\_Botryosphaeriales  
Length=589

Score = 309 bits (167), Expect = 2e-84  
 Identities = 215/238 (90%), Gaps = 3/238 (1%)  
 Strand=Plus/Minus

Query 1	TATATCAGGACTTCACGAAATGATCGCTTGAGTTTGATACTGGCAGGCACCTAGCCGG	60
Sbjct 237	TATATCAGGACTACACGGGATAATCGCTGGAGTTTGATACTGGCGGGCGCTGGCCGG	178
Query 61	GCGTCCTGGCCGGTTAAGGCTGGGGCGCCGCCGGTCGAAACCGGGTCGGCCCG	120
Sbjct 177	GCGTCCTGGCCGGTGAAGGCTGGGTGCGCCGCCGGTCGAAACCGGGTCGGCCCG	118
Query 121	CCAAAGCAACATGGTAAGGTACACAAGGGTGGGAAG-GGCTCTTCCGCCGGCTTTAG	179
Sbjct 117	CCAAAGCAACATGGTAAGGTACACAAGGGTGAGAAGAGGTTCTTCGGCCGGCTTT-G	59
Query 180	GGCCGGCGTACTCGATTACCTTCAAGAGAATTACGT-ATTCAGTAATGATCCTTCC	236
Sbjct 58	GGCCGGCTAACGAAAGACCTTCCAAGAGAATTACTTATTCAGTAATGATCCTTCC	1

> Phyllosticta\_styracicola|JX025040|SH210451.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes  
Length=564

Score = 291 bits (157), Expect = 9e-79  
 Identities = 203/225 (90%), Gaps = 4/225 (2%)  
 Strand=Plus/Minus

Query 1	TATATCAGGACTTCACGAAATGATCGCTTGAGTTTGATACTGGCAGGCACCTAGCCGG	60
Sbjct 223	TATATCAGGACTTCACAAAATAATCGCTTGAGTTTGACGCTGGCGGGCGCTTAGCCGG	164
Query 61	GCGTCCTGGCCGGTTAAGGCTGGGGCGCCGCCGGTCGAAACCGGGTCGGCCCG	120
Sbjct 163	GCGTCCTAGCCCGCGAGGGCTGGGGCGCCGCCGGTCGAAACCGGGTCGGCCCG	104
Query 121	CCAAAGCAACATGGTAAGGTACACAAGGGTGGGAAGGGCTTTCCGCCGGCTTTAGG	180
Sbjct 103	CCAAAGCAACGTGGTAAGGTACACAAGGGTGGGAAGGGCTTTCGGCCAGTTGAGG	44
Query 181	GCCGGCGTACTCGATTACCTTCAAGA-GAATT-ACGTATTCAG	223
Sbjct 43	GCCAGCCTACTCGAT--CCTTCAAGAAGAGTTACGTATTCAG	1

> Phyllosticta\_parthenocissi|EU683672|SH210443.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes  
Length=586

Score = 283 bits (153), Expect = 1e-76  
 Identities = 209/235 (89%), Gaps = 7/235 (3%)  
 Strand=Plus/Minus

Query 5	TCAGGACTTCACGAAATGATCGCTTGAGTTTGATACTGGCAGGCACTTAGCCGGCGT	64
Sbjct 231	TCAGGACTTCACAAGATAATCGCTTGAGTTTGATACTGGCGGGCGCTGGCCCGGGCGT	172
Query 65	CCTGGCC-GGTTAAGGCTGGGGCGCCGCCGCCGGTCGAAACCGGGTCGGCCGCCA	123
Sbjct 171	CCTGGCCCCGCGAGGGCTGGGGCGCCGCCGCCGGTCGAAACCGGGTCGGCCGCCA	112
Query 124	AAGCAACATGGTAAGGTACACAAGGGTGGGAAGGGCTTTCCGCC-GGCTTTTAGGC	182
Sbjct 111	AAGCAACGTGGTAGGGTACACAAGGGTGAGAAGGGCTTTCGGCCGGCTTGACGGC	52
Query 183	CGG-CGTACTCGATTACCTTCAAGAGAATTACGTATTCAGTAATGATCCTTCC	236
Sbjct 51	CGGGCGTACTCGAT--CCTTTC--GAGAATTACCAATTCAAGTAATGATCCTTCC	1

> Phyllosticta\_ampelicida|KC193586|SH210430.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;  
 Length=560

Score = 279 bits (151), Expect = 2e-75  
 Identities = 203/227 (89%), Gaps = 7/227 (3%)  
 Strand=Plus/Minus

Query 1	TATATCAGGACTTCACGAAATGATCGCTTGAGTTTGATACTGGCAGGCACTTAGCCGG	60
Sbjct 224	TATGTCAAGGACTTCACGAAATAATCGCTTGAGTTTGATACTGGCGGGCGCTGGCCGG	165
Query 61	GCGTCCTGGCCGGTTA-AGGCTGGGGCGCCGCCGCCGGTCGAAACCGGGTCGGCCC	119
Sbjct 164	GCGTCCTGGCCCGCGAGGGCTGGGGCGCCGCCGGTCGAAACCGGGTCGGCCC	105
Query 120	GCCAAAGCAACATGGTAAGGTACACAAGGGTGGGAAGGGCTTTCCGGCTTTTAG	179
Sbjct 104	GCCAAAGCAACATGGTAAGGTACACAGGGGTGAGAAGGGCTTTCGGACGGCTTTCA	45
Query 180	GGCCGGCGTACTCGATTACCTTCAAGA-GAA-TTACGTA-TTTCAG	223
Sbjct 44	GGCCGCCCTACTCGAT--CCATTCAAGAAGAAATTAC-TAGTTCA	1

> Phyllosticta\_aristolochiicola|JX486129|SH107086.07FU|refs\_singleton|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;  
 Length=566

Score = 268 bits (145), Expect = 4e-72  
 Identities = 212/242 (88%), Gaps = 14/242 (6%)  
 Strand=Plus/Minus

Query 1	TATATCAGGACTTCACGAAATGATCGCTTGAGTTTGATACTGGCAGGCACCTAGCCGG	60
Sbjct 231	TATATCAGGACTTCACAGAATAATCGCTTGAGTTTGATACTGGCGGGCACCTAGCCGG	172
Query 61	GCGTCCTGGCCGGTTAAGGCTGGGGCGCCGCCGGTCGAAACCGGGTCGGCCCG	120
Sbjct 171	GCGTCCTGGCCGGTTAAGGCTGGGGCGCCGCCGGTCGAAACCGGGTCGGCCCG	112
Query 121	CCAAAGCAACATGGTAAGGTACACAAGGGTGGGAAGGGCTTTCCGCCGGCTTTAGG	180
Sbjct 111	CCAAAGCAACATAGTG-GGTACACAAGGGTGAGAGAGGTTCTTCGGC-G--TTGTT-GC	57
Query 181	GCCGGCGTACTCGATTACCTTC-AGA-GA-ATTACGTATTCAGTAATGATCCTCCG	237
Sbjct 56	GCC---TACTCGAACCTTCAGAGAAGTTATTACG--TTTCAGTAATGATCCTCCG	3
Query 238	CA 239	
Sbjct 2	CA 1	

> Phyllosticta\_ardisiicola|AB454274|SH198936.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes  
Length=563

Score = 267 bits (144), Expect = 1e-71  
 Identities = 210/240 (88%), Gaps = 12/240 (5%)  
 Strand=Plus/Minus

Query 1	TATATCAGGACTTCACGAAATGATCGCTTGAGTTTGATACTGGCAGGCACCTAGCCGG	60
Sbjct 230	TATATCAGGACTTCACAAAATAATCGCTTGAGTTTGATACTGGCGGGCACCTAGCCGA	171
Query 61	GCGTCCTGGCC-GGTTAAGGCTGGGGCGCCGCCGGTCGAAACCGGGTCGGCC	119
Sbjct 170	GCGTCCTGGCCCTGTTGGGCTGGGGCGCCGCCGGCGAAACCGGGTCGGCC	111
Query 120	GCAAAGCAACATGGTAAGGTACACAAGGGTGGGAAGGGCTTTCCGCCGGCTTTAG	179
Sbjct 110	GCAAAGCAACATGGTAAGGTACACAAGGGTGGAGAGGTTCTTCGGCC---TTGC-AG	55
Query 180	GGCCGGCGTACTCGATTACCTTCAGAGAATTACGTATTCAGTAATGATCCTCCGCA	239
Sbjct 54	--CC---TACTCGAACCTTCAAAAGGATTAC-TATTCAGTAATGATCCTCCGCA	2

> Phyllosticta\_leucothoicola|AB454370|SH091775.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetidae  
Length=577

Score = 263 bits (142), Expect = 2e-70  
Identities = 212/244 (87%), Gaps = 11/244 (5%)  
Strand=Plus/Minus

Query	3	TATCAGGACTTCACGAAATGATCGCTTGA GTTTGTATACTGGCAGGCAC TTAGCCGGC 62
Sbjct	241	TATCAGAACTGCACGAAATAATCGCTTGA GTTTGTATACTGGCAGGCAC TTAGCCGGC 182
Query	63	GTCCTGGCCGGTTAAGGCTGGGGCGCCGGCCGCCGGTCGAAACCGGGTCGGCCCCGCC 122
Sbjct	181	GTCCTGGCCGGTGAGGGCTGGGGCGCCGGCCGCCGGTCGAAACCGGGTCGGCCCCGCC 122
Query	123	AAAGCAACATGGTAAGGTACACAAGGGTGGGA-AGGGCTTTCC-GCCGGCTT-TTTA 178
Sbjct	121	AAAGCAACATGGTAAGGTACACAAGGGTGTGAGAGGAC-CTTCCCCAGCCCCCTGTTA 63
Query	179	GGGCCGGC-G--TACTCGATTACCTTCAAGAGAATTACGTATTCAGTAATGATCCTTC 235
Sbjct	62	GGGGGGACTGGGTAC-CGGCGACCTTCA-GAGAATTAC-TAGTTCAAGTAATGATCCTTC 6
Query	236	CGCA 239
Sbjct	5	CGCA 2

> Phyllosticta\_neopyrolae|AB454318|SH244990.07FU|refs\_singleton|k\_Fungi;p\_Ascomycota;c\_Dothideomycetidae  
Length=577

Score = 257 bits (139), Expect = 9e-69  
Identities = 211/244 (86%), Gaps = 11/244 (5%)  
Strand=Plus/Minus

Query	3	TATCAGGACTTCACGAAATGATCGCTTGA GTTTGTATACTGGCAGGCAC TTAGCCGGC 62
Sbjct	241	TATCAGAACTGCACGAAATAATCGCTTGA GTTTGTATACTGGCAGGCAC TTAGCCGGC 182
Query	63	GTCCTGGCCGGTTAAGGCTGGGGCGCCGGCCGCCGGTCGAAACCGGGTCGGCCCCGCC 122
Sbjct	181	GTCCTGGCCGGTGAGGGCTGGGGCGCCGGCCGCCGGTCGAAGCCGGGTGGCCCCGCC 122
Query	123	AAAGCAACATGGTAAGGTACACAAGGGTGGGA-AGGGCTTTCC-GCCGGCTT-TTTA 178
Sbjct	121	AAAGCAACATGGTAAGGTACACAAGGGTGTGAGAGGAC-CTTCCCCAGCCCCCTGTTA 63

Query 179 GGGCCGGC-G--TACTCGATTACCTTCAGAGAATTACGTATTCAGTAATGATCCTTC 235  
||||| |||| | |||| || ||||||||| ||||||||| || ||||||||| |||||  
Sbjct 62 GGGGGGGCCGGGTAC-CGGCGACCTTCAGAATTAC-TAGTTCAAGTAATGATCCTTC 6

Query 236 CGCA 239  
|||||  
Sbjct 5 CGCA 2

Lambda K H  
1.33 0.621 1.12

Gapped  
Lambda K H  
1.28 0.460 0.850

Effective search space used: 3510693921

Query= OTU17:leafNotChim\_100;size=116774;

Length=254

\*\*\*\*\* No hits found \*\*\*\*\*

Lambda K H  
1.33 0.621 1.12

Gapped  
Lambda K H  
1.28 0.460 0.850

Effective search space used: 3753368616

Query= OTU15:leafNotChim\_100;size=153790;

Length=230

	Score (Bits)	E Value
Sequences producing significant alignments:		
Phyllosticta_capitalensis JF261465 SH198965.07FU refs k_Fungi;...	396	2e-110
Phyllosticta_capitalensis FJ769691 SH191739.07FU refs k_Fungi;...	377	6e-105
Phyllosticta_capitalensis JQ317505 SH198971.07FU refs k_Fungi;...	368	4e-102

Phyllosticta_capitalensis FJ769696 SH216176.07FU refs k_Fungi;... 368	4e-102
Phyllosticta_capitalensis FJ769686 SH216177.07FU refs k_Fungi;... 366	1e-101
Phyllostictaceae_sp KC291348 SH216174.07FU refs k_Fungi;p_Asc... 348	5e-96
Phyllosticta_capitalensis JQ317399 SH198966.07FU refs k_Fungi;... 340	8e-94
Phyllosticta_capitalensis JQ317409 SH198967.07FU refs k_Fungi;... 340	8e-94
Phyllosticta_capitalensis JQ317492 SH216175.07FU refs k_Fungi;... 329	2e-90
Phyllosticta_capitalensis JQ317504 SH210450.07FU refs k_Fungi;... 329	2e-90

> Phyllosticta\_capitalensis|JF261465|SH198965.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycete  
Length=540

Score = 396 bits (214), Expect = 2e-110  
Identities = 214/214 (100%), Gaps = 0/214 (0%)  
Strand=Plus/Minus

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGGG 60
Sbjct 214 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGGG 155
Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCGCCTGGTCCGAACCAGGTCGACCCGC 120
Sbjct 154 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCGCCTGGTCCGAACCAGGTCGACCCGC 95
Query 121 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 180
Sbjct 94 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 35
Query 181 CTGGAACCTTCAATAGAACAGTTATTACATTTCAG 214
Sbjct 34 CTGGAACCTTCAATAGAACAGTTATTACATTTCAG 1

> Phyllosticta\_capitalensis|FJ769691|SH191739.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycete  
Length=561

Score = 377 bits (204), Expect = 6e-105  
Identities = 213/217 (98%), Gaps = 1/217 (0%)  
Strand=Plus/Minus

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGGG 60
Sbjct 226 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGGG 167
Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCGCCTGGTCCGAACCAGGTCGACCCGC 120
Sbjct 166 CGTCCTGGCCAGTTAAGGCTGGCGCGCCGGCGCCTGGTCCGAACCAGGTCGACCCGC 107

```

Query 121 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 180
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 106 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 47
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 181 CTGGAACCTTCATAAGTTATTACATTCAGTAA 217
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |
Sbjct 46 CTGGAACCTTCATAAGTTAT-ACACATTCATTAA 11

```

> Phyllosticta\_capitalensis|JQ317505|SH198971.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycete  
Length=560

Score = 368 bits (199), Expect = 4e-102  
Identities = 216/224 (96%), Gaps = 2/224 (1%)  
Strand=Plus/Minus

```

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACGGCGGGCACTTAGCCGGG 60
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 225 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACGGCGGGCACTTAGCCGGG 166
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 120
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 165 CGTCCTGGCCAGTTAAGGCTGCGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 106
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 121 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 180
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 105 CAAAGCCACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 46
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 181 CTGGAACCTTCATAAGTTATTACATTCAGTAAT-GATCC 223
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |
Sbjct 45 CTGGAACCTTCATAAGTTAT-ACACATCAGTAGATCC 3

```

> Phyllosticta\_capitalensis|FJ769696|SH216176.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycete  
Length=564

Score = 368 bits (199), Expect = 4e-102  
Identities = 199/199 (100%), Gaps = 0/199 (0%)  
Strand=Plus/Minus

```

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACGGCGGGCACTTAGCCGGG 60
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 230 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACGGCGGGCACTTAGCCGGG 171
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 120
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 170 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 111
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||

```

```

Query 121 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 180
         ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 110 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 51
         ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Query 181 CTGGAACCTTCATAAGAA 199
         ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 50 CTGGAACCTTCATAAGAA 32

```

> Phyllosticta\_capitalensis|FJ769686|SH216177.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes  
Length=562

Score = 366 bits (198), Expect = 1e-101  
Identities = 202/204 (99%), Gaps = 0/204 (0%)  
Strand=Plus/Minus

```

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACGGCGGGCACTTAGCCGGG 60
         ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 228 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACGGCGGGCACTTAGCCGGG 169

```

```

Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 120
         ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 168 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 109

```

```

Query 121 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 180
         ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 108 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 49

```

```

Query 181 CTGGAACCTTCATAAGAAAGTTAT 204
         ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 48 CTGGAACCTTCGATAGAATTAT 25

```

> Phyllostictaceae\_sp|KC291348|SH216174.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;o\_E  
Length=506

Score = 348 bits (188), Expect = 5e-96  
Identities = 192/194 (99%), Gaps = 0/194 (0%)  
Strand=Plus/Minus

```

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACGGCGGGCACTTAGCCGGG 60
         ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 194 ATATCAGGACTTCACAATATGAATTCTTGAGTTTGATACGGCGGGCACTTAGCCGGG 135

```

```

Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 120
         ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 134 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 75

```

```

Query 121 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 180
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct  74  CAAAGCAACATAGTGAGTACACAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 15
          ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Query 181 CTGGAACCTTC 194
          ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 14  CTGGAACCTTC 1

```

> Phyllosticta\_capitalensis|JQ317399|SH198966.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycete  
Length=564

Score = 340 bits (184), Expect = 8e-94  
Identities = 194/199 (97%), Gaps = 1/199 (1%)  
Strand=Plus/Minus

```

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGG 60
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 230 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGG 171
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 120
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 170 CGTCCTGGCCAGTTAAGGCTGGNGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 111
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Query 121 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 180
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 110 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 51
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Query 181 CTGGAACCTTC-ATAGA 198
          ||| ||||| ||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 50  CTGAAACCTTCGGATAGA 32

```

> Phyllosticta\_capitalensis|JQ317409|SH198967.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycete  
Length=559

Score = 340 bits (184), Expect = 8e-94  
Identities = 196/202 (97%), Gaps = 1/202 (0%)  
Strand=Plus/Minus

```

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGG 60
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 224 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGG 165
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 120
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct 164 CGTCCTGGCCAGTTAAGGCTGGNGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 105
          ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||

```

```

Query 121 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 180
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 104 CAAAGCAACATAGTGAGAACACAAGGGTGAGAAGGTCAATTGGCGTTGT-GCGCCTACT 46
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 181 CTGGAACCTTCATAAGAGTT 202
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 45 CTGCTACCTTCAGAGAGAGTT 24

```

> Phyllosticta\_capitalensis|JQ317492|SH216175.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycete  
Length=561

Score = 329 bits (178), Expect = 2e-90  
Identities = 195/203 (96%), Gaps = 2/203 (1%)  
Strand=Plus/Minus

```

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGG 60
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 227 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGG 168
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 120
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 167 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCACGTCGACCCGC 108
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 121 CAAAGCAACATAGTGAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTACT 180
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 107 CAAAGCAACATAGTCAGTACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGAGCCTACT 48
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 181 CTGGA-ACCTTCATAAGAGTT 202
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 47 CAGGCTACCTT-CAATGGAAGTT 26

```

> Phyllosticta\_capitalensis|JQ317504|SH210450.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycete  
Length=567

Score = 329 bits (178), Expect = 2e-90  
Identities = 207/220 (94%), Gaps = 5/220 (2%)  
Strand=Plus/Minus

```

Query 1 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGG 60
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 230 ATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGG 171
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Query 61 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGGAACCAGGTCGACCCGC 120
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||
Sbjct 170 CGTCCTGGCCAGTTAAGGCTGGGGCGCCGGCCCTGGTCGACCCAGCTCGACCCGC 111
||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| ||| |||

```

```

Query  121  CAAAGCAACATAGTGAGTA-CACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTAC  179
        ||||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct   110  CAAAGCAACATAGTGAGTAACACAAGGGTGAGAAGGTCAATTGGCGTTGTAGCGCCTAC  51

Query  180  TC-TGGAACCTTCATAATAGAAGTTAT-TACATTTCA-GTA  216
        || | ||||| ||| ||||| | ||| ||||| ||| |
Sbjct   50   TCCTTAAACCTTCAGTAGAAGTACTCTAC-TTTCATGTA  12

```

Lambda K H  
1.33 0.621 1.12

Gapped  
Lambda K H  
1.28 0.460 0.850

Effective search space used: 3365089104

Query= OTU27:leafNotChim\_100;size=65318;

Length=159

	Score (Bits)	E Value
Sequences producing significant alignments:		
Diaporthales_sp KF436050 SH180177.07FU reps k_Fungi;p_Ascomyc... 78.7	5e-15	
Rhexodenticula_cylindrospora KM484943 SH529147.07FU reps k_Fun... 76.8	2e-14	

> Diaporthales\_sp|KF436050|SH180177.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Sordariomycetes;o\_Diap

Length=468

Score = 78.7 bits (42), Expect = 5e-15  
 Identities = 53/58 (91%), Gaps = 1/58 (2%)  
 Strand=Plus/Minus

```

Query  40  CCCGGCGAGCACCCCTCGCGGGTCGGGGCCCCCTCCCAGGGGCCGCCGAAGCAAC  97
        ||||||| || ||||| ||| || | ||||| ||||| ||||| ||||| ||||| ||||| |||||
Sbjct   99  CCCGGCGGGCGCCCTCGCGGG-CTGCAGGGCCCCCTCCCAGGGGCCGCCGAAGCAAC  43

```

> Rhexodenticula\_cylindrospora|KM484943|SH529147.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Sordariomyc

Length=485

Score = 76.8 bits (41), Expect = 2e-14  
 Identities = 91/114 (80%), Gaps = 8/114 (7%)  
 Strand=Plus/Minus

```

Query  41  CCGGCGAGCACCCCTCGCGGGTCGGGGCCCCTCCCAGGGGCCGCCGAAGCAACGAG  100
       ||||||| || ||| |||| | ||||||||||||||||||||| ||||| ||||| ||||| |
Sbjct   110  CCGGCGGGCGCCCCGCGAGGGGCTACGGCCCCTCCCAGGGGCCGCCGAAGCAAC-AG  52

Query  101  TTTTCAGGTAACGTTACG-GTGGTT-GGGAGTTTGAACTC--GCTAATGAT  150
       || | |||| | ||||| ||| || ||||||||| ||||| | ||||| ||||| |
Sbjct   51   TTA--CGGGTAG-GTTCACAAAGTGTTAGGGAGTTCAACTCTGGTAATGAT  1

```

Lambda        K            H  
   1.33     0.621        1.12

Gapped  
Lambda        K            H  
   1.28     0.460        0.850

Effective search space used: 2216428881

Query= OTU94:leafNotChim\_114;size=28778;

Length=254

\*\*\*\*\* No hits found \*\*\*\*\*

Lambda        K            H  
   1.33     0.621        1.12

Gapped  
Lambda        K            H  
   1.28     0.460        0.850

Effective search space used: 3753368616

Query= OTU91:leafNotChim\_100;size=29493;

Length=219

\*\*\*\*\* No hits found \*\*\*\*\*

Lambda K H  
1.33 0.621 1.12

Gapped  
Lambda K H  
1.28 0.460 0.850

Effective search space used: 3187127661

Query= OTU239:leafNotChim\_100;size=10568;

Length=211

\*\*\*\*\* No hits found \*\*\*\*\*

Lambda K H  
1.33 0.621 1.12

Gapped  
Lambda K H  
1.28 0.460 0.850

Effective search space used: 3057701157

Query= OTU16:leafNotChim\_100;size=127559;

Length=240

Sequences producing significant alignments:	Score (Bits)	E Value
Phyllosticta_neopyrolae AB454318 SH244990.07FU refs_singleton k...	244	7e-65
Guignardia_sp EU675682 SH198937.07FU reps k_Fungi;p_Ascomycot...	237	1e-62
Phyllosticta_yuccae UDB020917 SH091769.07FU refs k_Fungi;p_As...	235	4e-62
Phyllosticta_leucothoicola AB454370 SH091775.07FU refs k_Fungi...	233	1e-61
Phyllosticta_pachysandricola AB454317 SH186514.07FU refs k_Fun...	233	1e-61
Phyllosticta_paviae AB454261 SH186513.07FU reps k_Fungi;p_Asc...	230	2e-60
Guignardia_gaultheriae JN692543 SH186505.07FU refs k_Fungi;p_...	224	9e-59
Phyllosticta_rubella KF206171 SH244980.07FU refs k_Fungi;p_As...	220	1e-57
Phyllosticta_ampelicida HM049170 SH210446.07FU reps k_Fungi;p_...	217	1e-56
Phyllosticta_ilicis JN692538 SH210438.07FU refs k_Fungi;p_Asc...	217	1e-56

> Phyllosticta\_neopyrolae|AB454318|SH244990.07FU|refs\_singleton|k\_\_Fungi;p\_\_Ascomycota;c\_\_Dothid  
Length=577

Score = 244 bits (132), Expect = 7e-65  
 Identities = 200/230 (87%), Gaps = 16/230 (7%)  
 Strand=Plus/Minus

Query	20	ATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCGTCCTGCCGGTGAGGG	79
Sbjct	224	ATAATCGCTGGAGTTGTATACTGGCGGGCGCTTAGCCTGACGTCTGCCGGTGAGGG	165
Query	80	CTGGGGCGCCGGCCGCCGGTTTCGAGCCCGGGTCGGCCCCCAAAGCAACATGGTAA	139
Sbjct	164	CTGGGGCGCCGGCCGCCGGT--CGAAGCCGGTCGGCCCCCAAAGCAACATGGTAA	107
Query	140	GGTACACAAGGGGTGTGAGAGGCCCTCCAGCCGGCCGC--G---GGGCCGGACGTGG	194
Sbjct	106	GGTACACAAGGG-TGTGAGAGGACCTTCC--CCGGCCCCCTGTTAGGGGGGCCG-GG	51
Query	195	T-CGGG-GACCTTCAG-GA-TTACTGTGTTCACTGATCCTTCCGCA	240
Sbjct	50	TACCGGCGACCTTCAGAGAATTACTA-GTTCACTGATCCTTCCGCA	2

> Guignardia\_sp|EU675682|SH198937.07FU|reps|k\_\_Fungi;p\_\_Ascomycota;c\_\_Dothideomycetes;o\_\_Botryos  
Length=589

Score = 237 bits (128), Expect = 1e-62  
 Identities = 203/238 (85%), Gaps = 10/238 (4%)  
 Strand=Plus/Minus

Query	5	TCAGGACTACACTGGATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCGT	64
Sbjct	233	TCAGGACTACACGGATAATCGCTGGAGTTGTATACTGGCGGGCGCTTGCCGGCGT	174
Query	65	CCTGGCCGGTGAGGGCTGGGGCGCCGGCCGCCGGTTTCGAGCCCGGGTCGGCCGCC	124
Sbjct	173	CCTGGCCGGTGAGGTGGTGCGCCGGCCGCCGGT--CGAACACGGGTGGCCGCC	116
Query	125	AAAGCAACATGGTAAGGTACACAAGGGGTGTG-AGAGGCCCTCCAGCCGGCCGCGGG	183
Sbjct	115	AAAGCAACATGGTAAGGTACACAAGGG-TGAGAAGAGGTTCTTCGGCCGGCTTTGGG	57
Query	184	GCCGGACGTGGTCGGGGACCTTCA-G-GA-TTACTGTG-TTCAGTAATGATCCTCC	237
Sbjct	56	-CCGGCC-TAATCGAAGACCTTCCAAGAGAATTACTTATTCAGTAATGATCCTCC	1
		-	

> Phyllosticta\_yuccae|UDB020917|SH091769.07FU|refs|k\_\_Fungi;p\_\_Ascomycota;c\_\_Dothideomycetes;o\_\_Length=577

Score = 235 bits (127), Expect = 4e-62  
Identities = 208/244 (85%), Gaps = 17/244 (7%)  
Strand=Plus/Minus

Query	5	TCAGGACTACACTGGATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCGT	64
Sbjct	238	TCAGAACTACACGAAATAATCGCTGGAGTTTGATACTGGCGGGCGCTTAGCCTGACGT	179
Query	65	CCTGGCCGGTGAGGGCTGGGGCGCCGCCGGCCGGTTTCGAGCCCGGGTCGGCCCCGC	124
Sbjct	178	CCTGGCCGGCGAGGGCTGGGGCGCCGCCGGCCGGT--CGAAACCGGGTCGGCCCCGC	121
Query	125	AAAGCAACATGGTAAGGTACACAAGGGGTGTGAGAGGCCCTCCAGCCGGCCCGC---G-	180
Sbjct	120	AAAGCAACATGGTAAGGTACACA-GGGGTGTGAGAGGACCTTCC--CCGGCCCCCCTTGT	64
Query	181	--GGGGCCGGACGTGGT-CGGG-GACCTTCAG-GA-TTACTGTGTTAGTAATGATCC	233
Sbjct	63	TTAGGGGGGGGACCGGGTACCGCGACCTTCAGAGAATTACTA-GTTAGTAATGATCC	5
Query	234	TTCC 237	
Sbjct	4	TTCC 1	

> Phyllosticta\_leucothoicola|AB454370|SH091775.07FU|refs|k\_\_Fungi;p\_\_Ascomycota;c\_\_Dothideomycet Length=577

Score = 233 bits (126), Expect = 1e-61  
Identities = 197/229 (86%), Gaps = 14/229 (6%)  
Strand=Plus/Minus

Query	20	ATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCGCTGGCCGGTGAGGG	79
Sbjct	224	ATAATCGCTTGAGTTGTACTGGCGGGCGCTTAGCCTGGAGTCCTGGCCGGTGAGGG	165
Query	80	CTGGGGCGCCGGCCGCCGGTTTCGAGCCGGTCGGCCGCCAAAGCAACATGGTAA	139
Sbjct	164	CTGGGGCGCCGGCCGCCGGT--CGAAACCGGGTCGGCCGCCAAAGCAACATGGTAA	107
Query	140	GGTACACAAGGGGTGTGAGAGGCCCTCCAGCCGGCCCGC--G--GGGGCCGGACGTGGT	195
Sbjct	106	GGTACACAAGGG-TGTGAGAGGACCTTCC--CCAGCCCCCTGTTAGGGGGACTGGT	50
Query	196	-CGGG-GACCTTCAG-GA-TTACTGTGTTAGTAATGATCCTCCGCA 240	

Sbjct 49 ACCGGCGACTTTCAGAGAATTACTA-GTTCAGTAATGATCCTCCGCA 2

> Phyllosticta\_pachysandricola|AB454317|SH186514.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes  
Length=582

Score = 233 bits (126), Expect = 1e-61  
Identities = 209/247 (85%), Gaps = 14/247 (6%)  
Strand=Plus/Minus

Query 4	GTCAGGACTACACTGGATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCG 63
Sbjct 244	185
Query 64	TCCTGCCCGGTGAGGGCTGGGGCGCCGCCGCCGGTTTCGAGCCCGGGTCGGCCCGC 123
Sbjct 184	127
Query 124	CAAAGAACATGGTAAGGTACACAAGGGTGTGAGAGGCCCTCCA-G-CGGCCC-GC- 179
Sbjct 126	68
Query 180	--GGGGGCCGGACGTGGT-CGGG-GACCTTCAG-GA-TTACTGTGTTAGTAATGATCC 233
Sbjct 67	9
Query 234	TTCCGCA 240
Sbjct 8	2

> Phyllosticta\_paviae|AB454261|SH186513.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;o\_Basidiomycotina  
Length=584

Score = 230 bits (124), Expect = 2e-60  
Identities = 210/248 (85%), Gaps = 19/248 (8%)  
Strand=Plus/Minus

Query 5	TCAGGACTACACTGGATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCGT 64
Sbjct 242	183
Query 65	CCTGGCC-GGTGAGGGCTGGGGCGCCGCCGCCGGTTTCGAGCCCGGGTCGGCCCGC 123
Sbjct 182	126
Query 124	CAAAGAACATGGTAAGGTACACAAGGGTGTGAGAGGCCCTCCAGCCGGCCCGC---- 179

Sbjct	125		68
Query	180	--GGGGGCCGGACGTGGT-CGGG-GACCTTCAG-GATTACTGTGTTCAGTAATGATC	232
Sbjct	67	TTGGGGGG-GGACCGGGTACCGCGACCTTCAAAGAGATTACTA-GTTCAGTAATGATC	10
Query	233	CTTCCGCA 240	
Sbjct	9	CTTCCGCA 2	

> Guignardia\_gaultheriae|JN692543|SH186505.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;o\_ Length=566

Score = 224 bits (121), Expect = 9e-59  
 Identities = 188/218 (86%), Gaps = 14/218 (6%)  
 Strand=Plus/Minus

Query	20	ATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCGTCCTGGCCGGTGAGGG	79
Sbjct	212	ATAATCGCTGGAGTTTGTTACTGGCGGGCGCTTAGCCTGACGTCTGGCCGGTGAGGG	153
Query	80	CTGGGGCGCCGGCCGCCGGTTTCGAGCCC GGTCGGCCCCCAAAGCAACATGGTAA	139
Sbjct	152	CTGGGGCGCCGGCCGCCGGT--CGAAACGGGT CGGGCCCCCAAAGCAACATGGTAA	95
Query	140	GGTACACAAGGGGTGTGAGAGGCCCTCCAGCCGGCCGC--G--GGGGCCGGACGTGGT	195
Sbjct	94	GGTACACAAGGG-TGTGAGAGGACCTTCC--CCGGCCCCCTGTTAGGGGGACCGGGT	38
Query	196	-CGGG-GACCTTCAG-GA-TTACTGTGTTCAGTAATG 229	
Sbjct	37	ACCGGCGACCTTCAGAGAATTACTA-GTTCAGTAATG 1	

> Phyllosticta\_rubella|KF206171|SH244980.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;o\_ Length=535

Score = 220 bits (119), Expect = 1e-57  
 Identities = 156/173 (90%), Gaps = 5/173 (3%)  
 Strand=Plus/Minus

Query	5	TCAGGACTACACTGGATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCGT	64
Sbjct	224	TCAGAACTACACGAAATAATCGCTTGAGTTTGTTACTGGCGGGCGCTAGCCTGACGT	165
Query	65	CCTGGCCGGTGAGGGCTGGGGCGCCGGCCGGTTTCGAGCCC GGTCGGCCCCGCC	124

Sbjct	164	 CCTGGCCGGTGAGGGCTGGGGCGCCGGCCGCCGGT--CGAAACCGGGTCGGCCCCGC	107
Query	125	AAAGCAACATGGTAAGGTACACAAGGGGTGTGAGAGGCCCTCCAGCCGGCCC 	177
Sbjct	106	AAAGCAACATGGTAAGGTACACAAGGG-TGTGAGAGGACCTTCC--CCGGCCC	57

> Phyllosticta\_ampelicida|HM049170|SH210446.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;  
Length=585

Score = 217 bits (117), Expect = 1e-56  
 Identities = 204/244 (84%), Gaps = 13/244 (5%)  
 Strand=Plus/Minus

Query	5	TCAGGACTACACTGGATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCGT 	64
Sbjct	244	TCAGGACTTCACAAAATAATCGCTTGAGTTTGATACTGGCGAGCACTTAGCCGGCGT	185
Query	65	CCTGGCC-GGTGA-GGGCTGGGGCGCCGGCCGCCGGTTTCGAGCCGGTCGGCCCG 	122
Sbjct	184	CCTGGCCCGGTTAGGGCTGGGGCGCCGGCCGCCGGT--CGAAACCGGGTCGGCCCG	127
Query	123	CCAAAGCAACATGGTAAGGTACACAAGGGGTGTG-AGAGGCCCTCCAGCCGGCCCG 	181
Sbjct	126	CCAAAGCAACATGGTAAGGTACACAAGGG-TGAGAAGGGCTCTTCTGCCGGCTTTTA	68
Query	182	GGGCCGGACGTGGTCGGG-GACCTTCA-G-GA-TTACTGTG-TTCAGTAATGATCCTTC 	236
Sbjct	67	GGGCCG-ACGTACTCGAGAGACCTTCAAGAGAATTAC-GTATTCAGTAATGATCCTTC	10
Query	237	CGCA 240 	
Sbjct	9	CGCA 6	

> Phyllosticta\_ilicis|JN692538|SH210438.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;o\_B  
Length=568

Score = 217 bits (117), Expect = 1e-56  
 Identities = 146/159 (92%), Gaps = 5/159 (3%)  
 Strand=Plus/Minus

Query	5	TCAGGACTACACTGGATAATCGCTGGAATTTGTTACTGGCGGGCGCTTAGCCGGCGT 	64
Sbjct	232	TCAGGACTACACGTAATAATCGCTGGAGTTTGATACTGGCGGGCGCTTAGCCGGGAGT	173
Query	65	CCTGGCC-GGTGA-GGGCTGGGGCGCCGGCCGCCGGTTTCGAGCCGGTCGGCCCG 	122

Sbjct 172 CCTGGCCCGGTGAGGGCTGGGGCGCCGCCGGT--CGAAACCGGGTCGGCCCG 115

Query 123 CCAAAGCAACATGGTAAGGTACACAAGGGTGTGA-GAG 160

Sbjct 114 CCAAAGCAACATGGTAAGGTACACAAGGGGTGTGAAGAG 76

Lambda	K	H
1.33	0.621	1.12

Gapped		
Lambda	K	H
1.28	0.460	0.850

Effective search space used: 3526872234

Query= OTU199:leafNotChim\_101;size=13015;

Length=254

\*\*\*\*\* No hits found \*\*\*\*\*

Lambda	K	H
1.33	0.621	1.12

Gapped		
Lambda	K	H
1.28	0.460	0.850

Effective search space used: 3753368616

Query= OTU678:leafNotChim\_131;size=2272;

Length=191

### Sequences producing significant alignments:

Dermatocarpon dolomiticum | EFO14211 | SH205817\_Q7FU | refs | k Fungi : ... 97.1 2e-20

> Dermatocarpon\_dolomiticum|EFO14211|SH205817\_07FU|refs|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes

Length=488

Score = 97.1 bits (52), Expect = 2e-20  
Identities = 65/71 (92%), Gaps = 2/71 (3%)  
Strand=Plus/Minus

Query	104	ACGGGGCCCGCCGAAGCAACATGGTAAGGTAAACACAGGGTTGGAGAGGGGGCCCCGAGG	163
Sbjct	70	ACGGGGCCGCCAAGCAACGTGGTA-GGTAGACA-AGGGTTGGAGAGGGGGCCCCGAGG	13

Query	164	ACCCAAACTCG	174
Sbjct	12	ACCGGAACCTCG	2

Lambda	K	H
1.33	0.621	1.12

Gapped		
Lambda	K	H
1.28	0.460	0.850

Effective search space used: 2734134897

Database: unite.copy.fasta  
Posted date: Jul 15, 2018 6:08 PM  
Number of letters in database: 16,853,603  
Number of sequences in database: 30,695

Matrix: blastn matrix 1 -2  
Gap Penalties: Existence: 0, Extension: 2.5

Strikingly different from the high-throughput results. I trust these more, especially if there are multiple hits that agree.

So let's make a new chart of IDs for our core mycobiomes, with the following rules:

1. If several high quality matches in the manual blast are available that agree on a taxonomic assignment, use this consensus at the finest resolution possible, however broad.
  2. If a single very strong BLAST match exists (ex - 97%+ agreement on 100+ BP), use this, even if there isn't other evidence for it.
  3. If the two searches, BLAST and USEARCH agree in some way, this also strongly suggests a taxonomic assignment, even if there are not numerous BLAST matches to corroborate

4. Otherwise, no assignment is given. Weak assignments that seem plausible can be shown in parentheses.

#### Leaf taxonomy assignment tables

OTU	ID
OTU10	<i>Phyllosticta</i>
OTU15	<i>Phyllosticta capitalensis</i>
OTU16	<i>Phyllosticta</i>
OTU17	NoHit
OTU19	Clavicipitaceae ( <i>Metarhizium</i> ?)
OTU27	Ascomycota (Sordariomycetes?)(Dothideomycetes?)
OTU91	NoHit
OTU94	NoHit (Basidiomycota, Pucciniomycotina)
OTU199	NoHit (Chytridiomycota?)
OTU202	NoHit
OTU239	NoHit
OTU678	NoHit ( <i>Dermatocarpon</i> ?, <i>Phaeoisaria</i> ?)

We can expand this to include the higher ranks:

Taxa	Kingdom	Phylum	Subphylum	Class	Order	Family	Genus	Species
OTU10Fungi	Ascomycota	Pezizomycotina	Dothideomycetes	Bolzyces	Bolzyces	Bolzyces	Phyllosticta	
OTU15Fungi	Ascomycota	Pezizomycotina	Dothideomycetes	Bolzyces	Bolzyces	Bolzyces	Phyllosticta	<i>capitalensis</i>
OTU16Fungi	Ascomycota	Pezizomycotina	Dothideomycetes	Bolzyces	Bolzyces	Bolzyces	Phyllosticta	
OTU17Fungi								
OTU19Fungi	Ascomycota	Pezizomycotina	Sordariomycetes	Hypocreales	Clavicipitaceae			
OTU27Fungi	Ascomycota							
OTU91Fungi								
OTU94Fungi								
OTU199Fungi								
OTU202Fungi								
OTU239Fungi								
OTU678Fungi								

---

Taxa	Kingdom	Phylum	Subphylum	Class	Order	Family	Genus	Species
------	---------	--------	-----------	-------	-------	--------	-------	---------

---

Wood manual taxonomy assignments

Repeat the leaf BLAST pipeline here, for wood reads:

```
In [1]: ## still in R:  
load('helwoodfung.rda')  
sink("woodCore.txt")  
helwoodfung  
sink()  
## do some quick editing in vim to make a tuple out of this for our python script.
```

bash: syntax error near unexpected token `<'

Repeat: command not found

```
In [ ]: #!/usr/bin/env python3
```

```
## woodCorefindseq.py  
  
coreSeq=("OTU352:","OTU726:","OTU269:",  
"OTU287:","OTU250:","OTU257:","OTU84:")  
  
with open('otus_95_combo_nolb.fasta', 'r') as zoop:  
    refseq = zoop.readlines()  
  
with open('seqs_woodCoreMycobiom.fasta', 'w') as goop:  
    for j, otu in enumerate(coreSeq):  
        for i, line in enumerate(refseq):  
            if otu in line:  
                goop.write(line)  
                goop.write(refseq[i+1])
```

```
In [1]: UNITE='/home/daniel/Documents/taiwan/UNITE/unite.copy.fasta'  
woodCoreSeqs='/home/daniel/Documents/taiwan/taiwan_combined_stats/CSrev/seqs_woodCoreMycobiom.fasta'  
blastn -query $woodCoreSeqs -db $UNITE -out woodCoreMycobiom_blast.txt -num_descriptions=1
```

```
In [2]: cat woodCoreMycobiom_blast.txt
```

BLASTN 2.2.31+

Reference: Zheng Zhang, Scott Schwartz, Lukas Wagner, and Webb Miller (2000), "A greedy algorithm for aligning DNA sequences", J Comput Biol 2000; 7(1-2):203-14.

Database: unite.copy.fasta  
30,695 sequences; 16,853,603 total letters

Query= OTU352:1w;size=6443;

Length=182

Sequences producing significant alignments:	Score (Bits)	E Value
Chaetothyriales_sp KC965461 SH212404.07FU reps k_Fungi;p_Asco...	257	6e-69
Fungi_sp KU580527 SH530645.07FU reps k_Fungi;p_unidentified;c...	244	5e-65

> Chaetothyriales\_sp|KC965461|SH212404.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;o\_Ch

Length=509

Score = 257 bits (139), Expect = 6e-69  
Identities = 166/179 (93%), Gaps = 1/179 (1%)  
Strand=Plus/Plus

Query 1	AAGAGTTCGGGTCCTCACGGGCCTAACCTCCAACCCTTGTCTACCAAACCTCACGTTGC	60
Sbjct 1	AAGAGTTCGGGTCCTCACGGGCCTAACCTCCAACCCTTGTCTACCAAACCATACGTTGC	60
Query 61	TTCGGCGGGTCTGTCTTCAGACTGCCGGGAGTTCACACCCCTGGCTCGTACCCGCCGG	120
Sbjct 61	TTCGGCGGGTCTGTCTTCAGACTGCCGGGGGTACACCCCTGGCTCGTACCCGCCGG	120
Query 121	TGGCCAACCTAACCAAAACTCTTAATTGAACGTGTCTGAATACTTATTAAATAATAAT	179
Sbjct 121	TGGCCAAC-CAACCAAAACTCTTAACCTAACCGTGTCTGAATACTTAAAAGTAATAAT	178

> Fungi\_sp|KU580527|SH530645.07FU|reps|k\_Fungi;p\_unidentified;c\_unidentified;o\_unidentified;  
Length=500

Score = 244 bits (132), Expect = 5e-65  
Identities = 165/180 (92%), Gaps = 5/180 (3%)  
Strand=Plus/Plus

Query 2	AGAGTTCGGGTCCTCACGGGCCTAACCTCCAACCCTTGTCTACCAAACCTCACGTTGCT	61
Sbjct 2	AGAGTTCGGGTCCTCACGGGCCTAACCTCCAACCCTTGTCTACCAAACCTCACGTTGCT	61

Query	62	TCGGCGGGTCTGTCTTCAGACTGCCGGGAGTTCACACCCCTGGCTCGTACCCGCCGGT	121
Sbjct	62	TCGGCAGGTCTGTCTTCAGACTGCCGGG-TTTC-CACCCCTGGCTCGTACCCGCCGGT	119
Query	122	GGCCAACCAACCAAAACTCTT--AATTGAACGTGTCTGAATACTTATTAAA-TAATAA	178
Sbjct	120	GGCCAACCAACCAAAACTCTTTAAATAATCGCGTCTGAATACTTAAAAAGTAATAA	179

Lambda	K	H
1.33	0.621	1.12

Gapped		
Lambda	K	H
1.28	0.460	0.850

Effective search space used: 2588530080

Query= OTU726:72w;size=2046;

Length=205

Sequences producing significant alignments:	Score (Bits)	E Value
Dothideomycetes_sp EF373557 SH202790.07FU reps k_Fungi;p_Asco...	213	2e-55
Capnodiales_sp JF691278 SH202792.07FU reps k_Fungi;p_Ascomyco...	196	2e-50
Sarocladium_im implicatum EF556211 SH177343.07FU reps k_Fungi;p_...	56.5	3e-08

> Dothideomycetes\_sp|EF373557|SH202790.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;o\_un  
Length=498

Score = 213 bits (115), Expect = 2e-55  
 Identities = 155/173 (90%), Gaps = 7/173 (4%)  
 Strand=Plus/Plus

Query	38	TAGGTGAACCTGCGGAGGGATCATTACCAAGAGATGGCGGGCCTCCGTGCTCCGTCCCCA	97
Sbjct	6	TAGGTG-ACCTGCGGAGGGATCATTACCAAAGATGGCGGGCCTCGGGTCCCCGTCCCCA	64
Query	98	ACCCATTGTTGA-CCAAAACCCCTTGCCTCGGGGCAGAGCGT---CCGCCGCCCGG	153
Sbjct	65	ACCCATTGTTGACCCAAAACCTTTGCCTCGGGGCAGAGCGTCCGCCGCCGG	124
Query	154	AGGACCACCGAACGCTGTCTCTCGCGTCAGAGTAATTGATT-AAATGAATGA	205

Sbjct 125 ||||||| ||||| | | ||| ||||| ||||| ||||| |||||  
AGGACCACCGAACGCTGTCTTACGTCGGAGTA-TTGATTTAAATAATGA 176

> Capnodiales\_sp|JF691278|SH202792.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Dothideomycetes;o\_Capnod

Length=480

Score = 196 bits (106), Expect = 2e-50  
Identities = 132/144 (92%), Gaps = 4/144 (3%)  
Strand=Plus/Plus

Query 47	CTGC GGAGGGATCATT ACCAGAGATGGCGGGCCTCCGTGCTCCCGTCCCCAACCCATTGT	106
Sbjct 1	CTGC GGAGGGATCATT ACCAGAGATGGCGGGCCCCCGCGCCCCCGTCCCCAACCCATTGT	60
Query 107	TGACCAAAACCCCTTTGCCTCGGGGGCAGAGCGT---CCGCCGCCCCCGGAGGACCACCG	163
Sbjct 61	TGACCAAAA-CCTTTGCCTCGGGGGCGGAGCGTCCGCCGCCGGAGGACCACCG	119
Query 164	AACGCTGTCTCTGCGTCAGAGT	187
Sbjct 120	AACGCTGTCTCTGTCAGTCGGAGT	143

> Sarocladium\_im implicatum|EF556211|SH177343.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Sordariomycetes;o\_C

Length=560

Score = 56.5 bits (30), Expect = 3e-08  
Identities = 33/34 (97%), Gaps = 1/34 (3%)  
Strand=Plus/Minus

Query 38	TAG-GTGAACCTGCGGAGGGATCATTACCAGAGA	70
Sbjct 553	TAGTGTGAACCTGCGGAGGGATCATTACCAGAGA	520

Lambda	K	H
1.33	0.621	1.12

Gapped		
Lambda	K	H
1.28	0.460	0.850

Effective search space used: 2960631279

Query= OTU269:1w;size=9056;

Length=126

\*\*\*\*\* No hits found \*\*\*\*\*

Lambda K H  
1.33 0.621 1.12

Gapped

Lambda K H  
1.28 0.460 0.850

Effective search space used: 1701945840

Query= OTU287:3w;size=8152;

Length=254

\*\*\*\*\* No hits found \*\*\*\*\*

Lambda K H  
1.33 0.621 1.12

Gapped

Lambda K H  
1.28 0.460 0.850

Effective search space used: 3753368616

Query= OTU250:4w;size=9658;

Length=191

Sequences producing significant alignments:	Score (Bits)	E Value
Dermatocarpon_dolomiticum EF014211 SH205817.07FU refs k_Fungi;...	97.1	2e-20

> Dermatocarpon\_dolomiticum|EF014211|SH205817.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes  
Length=488

Score = 97.1 bits (52), Expect = 2e-20  
 Identities = 65/71 (92%), Gaps = 2/71 (3%)  
 Strand=Plus/Plus

Query	2	CGAGTTGGGTCTTCGGGCCCCCTCTCCAACCCGTGTTACCTTACCATGTTGCTTC	61
Sbjct	2	CGAGTTGGGTCTTCGGGCCCCCTCTCCAACCC-TGTCTACC-TACCACGTTGCTTT	59
Query	62	GGCGGGCCCGT	72
Sbjct	60	GGCGGGCCCGT	70

Lambda	K	H
1.33	0.621	1.12

Gapped		
Lambda	K	H
1.28	0.460	0.850

Effective search space used: 2734134897

Query= OTU257:3w;size=9392;

Length=249

Sequences producing significant alignments:	(Bits)	Score	E
Trichomerium_foliicola JX313651 SH177466.07FU reps k_Fungi;p_-...	344	7e-95	
Chaetothyriales_sp GU054287 SH177468.07FU reps k_Fungi;p_Asco...	287	1e-77	
Trichomerium_dioscoreae KP004468 SH186766.07FU refs k_Fungi;p_...	272	3e-73	
Trichomeriaceae_sp KP174853 SH530157.07FU reps k_Fungi;p_Asco...	207	9e-54	
Trichomeriaceae_sp KP174850 SH528223.07FU reps k_Fungi;p_Asco...	198	6e-51	
Chaetothyriales_sp GQ999538 SH186764.07FU reps k_Fungi;p_Asco...	132	6e-31	
Chaetothyriales_sp GU055944 SH177465.07FU reps k_Fungi;p_Asco...	121	1e-27	
Capronia_sp EU520629 SH180479.07FU reps k_Fungi;p_Ascomycota;...	63.9	2e-10	
Cladosporium_adianticola DQ008125 SH101078.07FU refs_singleton ...	62.1	8e-10	
Arthrocladium_sp HQ634657 SH208969.07FU reps k_Fungi;p_Ascomy...	60.2	3e-09	

> Trichomerium\_foliicola|JX313651|SH177466.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;o\_

Length=546

Score = 344 bits (186), Expect = 7e-95  
 Identities = 238/261 (91%), Gaps = 12/261 (5%)

## Strand=Plus/Plus

Query	1	CCGAGTTAGGGTCTCCTCTGTGAGGCCGACCTCCAACCGCGTCTAATAAACCTGTT	60
Sbjct	1		
Query	61	TGTGTTGCTTCGGCGAACGGCAGTCGT--C---CTCCCCTTCACCgggg--g--ggATA	111
Sbjct	61		
Query	112	ACCGTCGCCGGGGACT-CCCGGTCTCACGACCG-CCCCTGGAGAGCGTCCGCCATGGC	169
Sbjct	121		
Query	170	CCCAACCAAAACAACATACCAAACCAAACATATGGAGTTAAATTCTGAATCAGA-CTT	228
Sbjct	181		
Query	229	TTGATATACCAATAAAAACA	249
Sbjct	241		
Sbjct	241	TTGATATACCAATAAAAACA	261

> Chaetothyriales\_sp|GU054287|SH177468.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;o\_Cha  
Length=588

Score = 287 bits (155), Expect = 1e-77  
Identities = 229/262 (87%), Gaps = 15/262 (6%)  
Strand=Plus/Plus

Sbjct 239 TTTGATATAACCAATCAAAAAACA 260

> Trichomerium\_dioscoreae|KP004468|SH186766.07FU|refs|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;c\_Length=571

Score = 272 bits (147), Expect = 3e-73  
Identities = 219/251 (87%), Gaps = 15/251 (6%)  
Strand=Plus/Plus

Query	1	CCGAGTTAGGGTCTCCTCTGTGAGGCCCGACCTCCAACCCTGTCTAATAAACCTGTT	60
Sbjct	1	CCGAGTTAGGGTCTCCTCTGTGAGGCCCGACCTCCAACCCTGTCTAATAAACCTGTA	60
Query	61	TGTGTTGCTTCGGCGGAACGGCAGTCGTCCCTCCCCTCACCGgggggggATAACCGTCGCC	120
Sbjct	61	TGTGTTGCTTCGGCGGAACGGCAGTC---AACCCCTT---C-GGGGAGA-GACCGTCGCC	111
Query	121	GGGGGA-CTC-CCGGTCTCACGACC-GCCCCCTGGAGAGCGTCCGCCGATGGCCCAACCA	177
Sbjct	113	GGGGGATCTCACCGGTCTCACGACCGGCCCTGGAGAGCGTCCGCCGATGGCCC-AACCA	171
Query	178	AAACAAC TACCAAACCCAAACTATGG-AGTTAAAATTCTGAATCAGACTTTGATATA	230
Sbjct	172	AACAA ACTCCAA ACC- AAATAATAGTACCTAAC- TTCTGAATCAGACTTTGATATA	220
Query	237	CCAATCAAAAA 247	
Sbjct	230	CCAATCAAAAA 240	

> Trichomericiaeae\_sp|KP174853|SH530157.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;o\_Chrysosporidales;f\_Coprinaceae;g\_Coprinus;sp\_Coprinus\_trichomericiaeae;v\_1 Length=579

Score = 207 bits (112), Expect = 9e-54  
Identities = 209/251 (83%), Gaps = 25/251 (10%)  
Strand=Plus/Plus

```

Sbjct 116 GGGGG-----C-G--T--C--CCGCCCTGGAGAGCGTCCGCCGATGGCCC-AACCAAAA 162
Query 181 CAA-CTACCAAACCCAAACTATGG-AGTTAAAATTTCTGAATCAGACTTTGATATAACC 238
||||||| ||||||||| ||||| ||| | ||||| ||||||||||||||||||||| ||||| ||||| |
Sbjct 163 CAAACTCCCAAACC-AAATAATAGTACCTAAAAC-TTCTGAATCAGACTTTGATATAACC 220
Query 239 AATCAAAAACA 249
||||||| |||||
Sbjct 221 AATCAAAAACA 231

```

> Trichomericaceae\_sp|KP174850|SH528223.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;o\_Cha  
Length=598

Score = 198 bits (107), Expect = 6e-51  
 Identities = 214/261 (82%), Gaps = 26/261 (10%)  
 Strand=Plus/Plus

```

Query 1 CCGAGTTAGGGTCTCC-TCTGTGAGGCCCGACCTCCAACCCCGTGTCTAATAAACCTGT 59
||||||| ||||| ||||| ||| | | ||||| ||||||||||||||||||||| ||||| ||||| |
Sbjct 4 CCGAGTTAGGGTCTCCCTC-GCGGGGCCCGACCTCCAACCCCGTGTCTAATAAACCTGT 62
Query 60 TTGTGTTGCTTCGGCGAACGGCAGTCGTCC-TCCCCT-T-CACCgggggggATAACCGT 116
||||||| ||||| ||||| ||||| | ||| | | | | | | | | | | | | | | | | | | | |
Sbjct 63 ACGTGGTGCCTCGGCGGACCGGC-GTCGTGCGTCTCGTATGCAC-G-----AT--CTGC 112
Query 117 CGCCGGGGGACTC-CCGG-T-CTCACGA--CC-GCCCCTGGAGAGCGTCCGCCGATGGCC 170
||||||| ||| | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
Sbjct 113 CGCCGGGGGAGTCACCGGCTGCTCACGCCGCCCTGGAGAGCGTCCGTCGATGGCC 172
Query 171 CCAACCAAAACAACCAACCAACCAACT-ATGGAGTTAAAATTTCTGAATCAGA-CTT 228
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
Sbjct 173 C-AACCAAAACAACCAACCAAAATTATGGA-TGAAACCTT-CTGAATCAGACCTT 229
Query 229 TTGATATACCAATCAAAAACA 249
||||||| ||||| ||||| |||||
Sbjct 230 TTGATATACCAATCAAAAACA 250

```

> Chaetothyriales\_sp|GQ999538|SH186764.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;o\_Cha  
Length=595

Score = 132 bits (71), Expect = 6e-31  
 Identities = 198/257 (77%), Gaps = 18/257 (7%)  
 Strand=Plus/Plus

```

Query 1 CCGAGTTAGGGTCTCCT-CTGTGAGGCCCGACCTCCAACCCCGTGTCTAATAAACCTGT 59
||||||| ||||| | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

```

> Chaetothyriales\_sp|GU055944|SH177465.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;o\_Cha  
Length=561

Score = 121 bits (65), Expect = 1e-27  
Identities = 197/254 (78%), Gaps = 36/254 (14%)  
Strand=Plus/Plus

Query	1	CCGAGTTAGGG-TCTCCTCTGTGAGGCCGACCTCCAAACCCGTGTCTAATAAACCTGT 	59
Sbjct	4	CCGAGTTAGGGTCT-CTC-CCGAGGCCGACCTCCAAACCTGTGTCTACCACCTGT 	61
Query	60	-TTGTGTTGCTTCGGCGAACGGCAGTCGTCTCCCCTCACCGgggggggATAACCGTCG 	118
Sbjct	62	CCCGTGTGCTTCGGCGACCGGCAGTC-----CCC-TCA-C--GGGGG-T--CCGTCG 	108
Query	119	CCGGGGGACTCCCGGTCTCACGACCGCCCTGGAGAGCGTCCGCCGATGGCCCCAACCAA 	178
Sbjct	109	CC-GGGG-----GG--T--C-A---CCCCTGGAGAGCGTCCGCCGATGGCCCCAACCAA 	153
Query	179	AACAAC TACCAAAACCCAAACTATGG-AGTTAAAATTTCTGAATCAGAC--TTTGATAT 	235
Sbjct	154	AACAACGCCAACCCAAATTACGTACCTAAACTT-CTGAATCAAACCTTTGATGT 	212
Query	236	ACCAATCAAAAACA 249 	
Sbjct	213	ACCAATCAAAAACA 226	

> Capronia\_sp|EU520629|SH180479.07FU|reps|k\_\_Fungi;p\_\_Ascomycota;c\_\_Eurotiomycetes;o\_\_Chaetothyriales

Length=522

Score = 63.9 bits (34), Expect = 2e-10  
Identities = 41/44 (93%), Gaps = 2/44 (5%)  
Strand=Plus/Plus

Query	142	CCGCCCTGGAGAGCGTCCGCCGATGGCCCCAACCAAAACACT	185
Sbjct	111	CCGCCCTGGAGAGCGTCCGCCGATGGCCC-AACCACAA-AACT	152

> Cladosporium\_adianticola|DQ008125|SH101078.07FU|refs\_singleton|k\_Fungi;p\_Ascomycota;c\_Dothi  
Length=555

Score = 62.1 bits (33), Expect = 8e-10  
Identities = 40/43 (93%), Gaps = 2/43 (5%)  
Strand=Plus/Plus

Query	144	GCCCCTGGAGAGCGTCCGCCGATGGCCCCAACCA-AAACAACT	185
Sbjct	111	GCCCCTGGAGAGCGTCCGCCGATGGCCC-AACCACAAAAAAACT	152

> Arthrocladium\_sp|HQ634657|SH208969.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;o\_Chaet  
Length=522

Score = 60.2 bits (32), Expect = 3e-09  
Identities = 39/42 (93%), Gaps = 2/42 (5%)  
Strand=Plus/Plus

Query	144	GCCCCTGGAGAGCGTCCGCCGATGGCCCCAACCAAAACACT	185
Sbjct	105	GCCCCTGGAGAGCGTCCGCCGATGGCCC-AACCACA-CAACT	144

Lambda	K	H
1.33	0.621	1.12

Gapped  

Lambda	K	H
1.28	0.460	0.850

Effective search space used: 3672477051

Query= OTU84:38w;size=31914;

Length=220

Sequences producing significant alignments:	Score (Bits)	E Value
Chaetothyriales_sp KF614884 SH532208.07FU reps k_Fungi;p_Asco...	76.8	2e-14
Candelariella_sp KR052103 SH532250.07FU reps k_Fungi;p_Ascomy...	65.8	5e-11

> Chaetothyriales\_sp|KF614884|SH532208.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Eurotiomycetes;o\_Cha

Length=626

Score = 76.8 bits (41), Expect = 2e-14  
Identities = 92/115 (80%), Gaps = 10/115 (9%)  
Strand=Plus/Plus

Query 13	CTAACCCtttttttAGGTGAACCTGCGGAAGGATCATTAAAGAGTGGGGTTGCTCGATT	72
Sbjct 42	CTAACCC--TTTTTAGGTGAACCTGCGGAAGGATCATTAAAGAGTTAGGGT-CTCTAT-	97
Query 73	GAGCGCCCAAACTCCCAACCCTTGATGACTTGACAAAAATTGTTGCTTCGGCGG	127
Sbjct 98	G-GC-CCGAC-CTCCCAACCCTATG-TGTATTGA-ACCTG-TGTTGCTTCGGCGG	146

> Candelariella\_sp|KR052103|SH532250.07FU|reps|k\_Fungi;p\_Ascomycota;c\_Lecanoromycetes;o\_Cand

Length=565

Score = 65.8 bits (35), Expect = 5e-11  
Identities = 35/35 (100%), Gaps = 0/35 (0%)  
Strand=Plus/Plus

Query 24	ttttAGGTGAACCTGCGGAAGGATCATTAAAGAGT	58
Sbjct 61	TTTTAGGTGAACCTGCGGAAGGATCATTAAAGAGT	95

Lambda K H  
1.33 0.621 1.12

Gapped  
Lambda K H  
1.28 0.460 0.850

Effective search space used: 3203305974

Database: unite.copy.fasta

Posted date: Jul 15, 2018 6:08 PM  
Number of letters in database: 16,853,603  
Number of sequences in database: 30,695

Matrix: blastn matrix 1 -2  
Gap Penalties: Existence: 0, Extension: 2.5

Wood taxonomy assignment tables

Same rules as the leaves...

OTU	ID
OTU84	Chaetothyriales
OTU250	NoHit ( <i>Dermatocarpon</i> )
OTU257	Trichocomeriaceae
OTU269	NoHit (Tremellales)
OTU287	NoHit
OTU352	Chaetothyriales
OTU726	Capnodiales

We can expand this to include the higher ranks:

Taxa	Kingdom	Phylum	Subphylum	Class	Order	Family	Genus	Species
OTU84	Fungi	Ascomycota	Pezizomycotina	Barotiomycetes	Chaetothyriales			
OTU25	Fungi							
OTU25	Fungi	Ascomycota	Pezizomycotina	Barotiomycetes	Chaetothyriales	Trichocomeriaceae		
OTU26	Fungi							
OTU28	Fungi							
OTU35	Fungi	Ascomycota	Pezizomycotina	Barotiomycetes	Chaetothyriales			
OTU72	Fungi	Ascomycota	Pezizomycotina	Dothideomycetes	Capnodiales			

Spatial patterns of "core" *Helicia* leaf mycobiome

Our network analysis brought up candidates for fungi that are closely associated with *Helicia formosana*. But how do these fungi behave spatially?

```
In [ ]: load('deseq95.rda')
load('helleaffung.rda') ## our list of strongly associated Helicia leaf fungi
```

```

leafHel95 <- subset_samples(deseq95, Library == 'L' & Host_genus_species == "Helicia_for"

In [218]: ## get the BC distance of all helicia points from the core:
aa <- t(otu_table(leafHel95))
aa <- aa[,helleaffung] ## subset OTU to just these rows
aa[aa > 0] <- 1 ## PA
leafXY <- sample_data(leafHel95)[,c('X','Y')] ## spatial coords
aaNo0 <- aa[rowSums(aa) > 0,] ## get rid of zero rows, for nms
core <- vector(length=ncol(aa)); core[] <- 1 ## our "core" mycobiome
leafHelcoreOTU <- rbind(core, aa) ## stack them, including zeroes (for map)
leafHelcoreOTUno0 <- rbind(core, aaNo0) ## stack them, not including zeroes (for NMS)
bray <- as.matrix(vegdist(leafHelcoreOTU)) ## comparisons of mutually zero rows go to NA
leafHelcoreBC <- bray['core',-1] ## use core comparison, but remove core entry to match
coreBCroundup <- round(leafHelcoreBC*100+1) ## turn our BC values into heatmap values
core_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101) ## colors for p

```

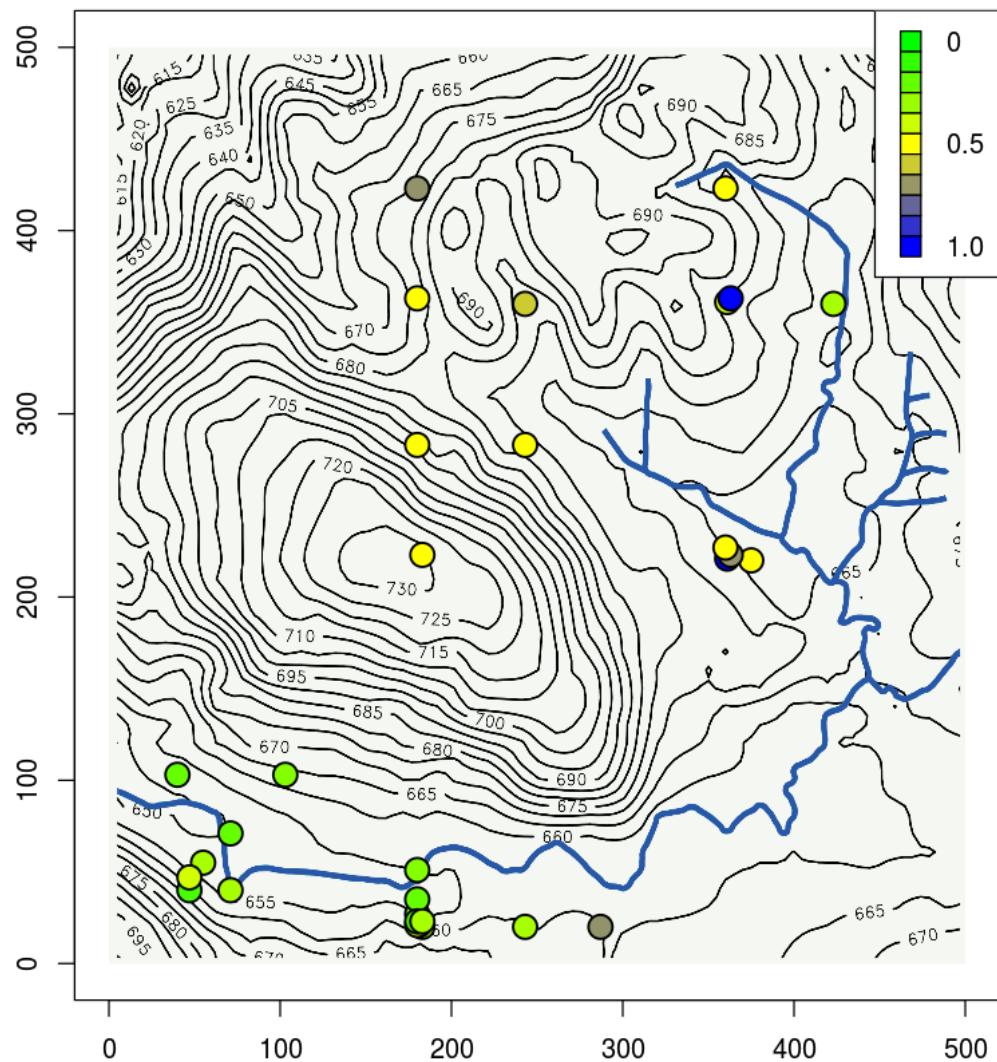
Warning message in vegdist(leafHelcoreOTU):

you have empty rows: their dissimilarities may be meaningless in method  
brayWarning message in vegdist(leafHelcoreOTU):  
missing values in results

```

In [221]: topo <- readPNG('/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png') ## load
plot(1, type='n',
      xlim=c(0,500),
      ylim=c(0,500),
      xlab = '',
      ylab = '',
) ##blank plot
rasterImage(topo,0,0,500,500)
points(leafXY,
      pch=21,
      cex=2,
      bg = core_palette2[coreBCroundup],
      lwd=1.5,
)
#draw.circle(x=183,y=223, radius=200, border="red", lwd=2, lty = 2)
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'
legend("topright",
      fill = core_palette2[seq(1,101,10)],
      legend = heatlegend,
      bg = "white",
      y.intersp = 0.5,
)

```



BC dissimilarity scale used here. So dark blue spots indicate a site with no fungi from the list of strongly cooccurring fungi. Green spots indicate sites with all of these fungi. As before, something is going on in the southwest valley...

Spatial patterns of "core" *Helicia* wood mycobiome

```
In [223]: load('helwoodfung.rda') ## our list of strongly associated Helicia wood fungi, 'helwoodfung'
woodHel95 <- subset_samples(deseq95, Library == 'W' & Host_genus_species == "Helicia_fungus")
```

```
In [224]: ## get the BC distance of all helicia points from the core:
aa <- t(otu_table(woodHel95))
```

```

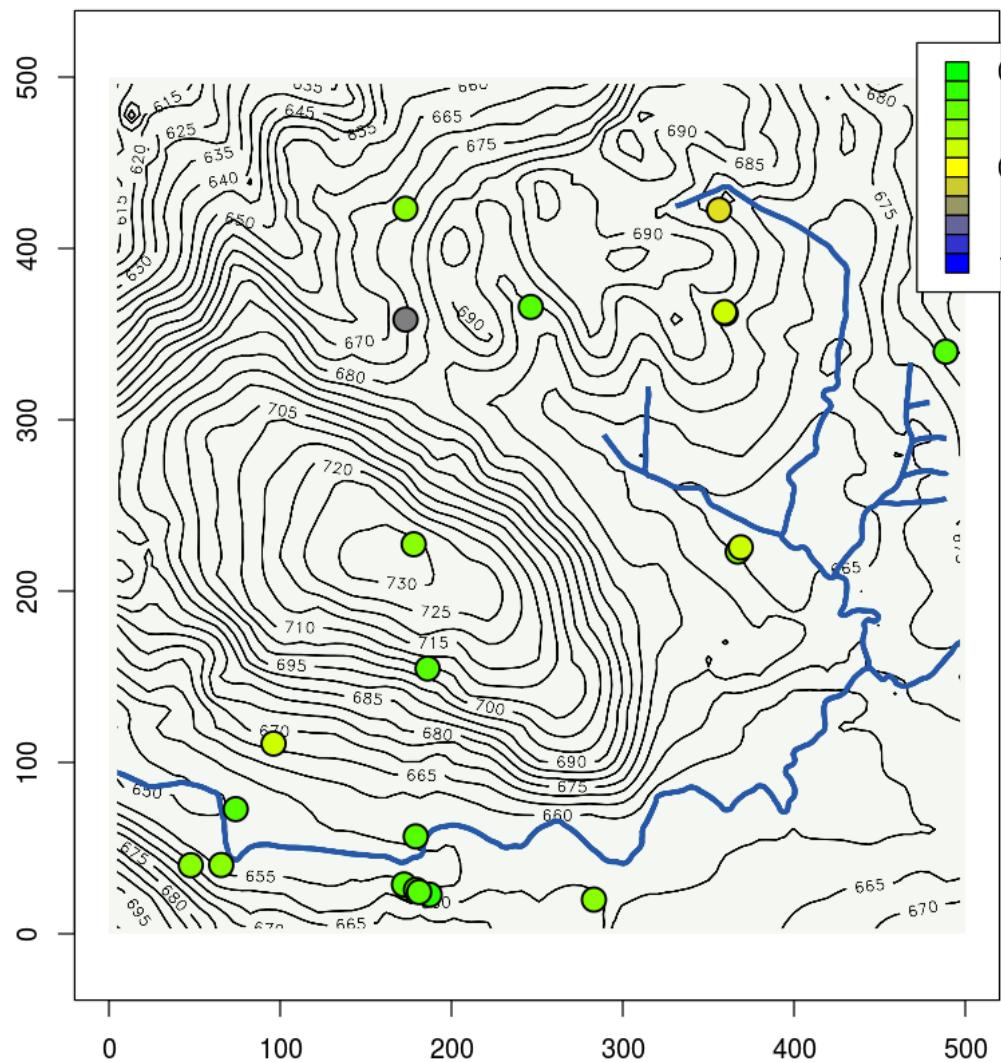
aa <- aa[,helwoodfung] ## subset OTU to just these rows
aa[aa > 0] <- 1 ## PA
woodXY <- sample_data(woodHel95)[,c('X','Y')] ## get positions
## for jittering, might be useful to fully convert the positions:
woodHelXY <- data.frame(cbind(as.numeric(woodXY$X), as.numeric(woodXY$Y)))
colnames(woodHelXY) <- woodXY@names
core <- vector(length=ncol(aa)); core[] <- 1 ## our "core" mycobiome
woodHelcoreOTU <- rbind(core, aa) ## stack them, including zeroes (for map)
bray <- as.matrix(vegdist(woodHelcoreOTU)) ## comparisons of mutually zero rows go to M
woodHelcoreBC <- bray['core',-1] ## use core comparison, but remove core entry to match
coreBCroundup <- round(woodHelcoreBC*100+1) ## turn our BC values into heatmap values
core_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101) ## colors for p

```

```

In [225]: topo <- readPNG('/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png') ## load
plot(1, type='n',
      xlim=c(0,500),
      ylim=c(0,500),
      xlab = '',
      ylab = '',
      asp=1
) ##blank plot
rasterImage(topo,0,0,500,500)
#points(woodXY, ## no jitter
points(cbind(jitter(woodHelXY$X,40), jitter(woodHelXY$Y,40)), ## jitter
      pch=21,
      cex=2,
      bg = core_palette2[coreBCroundup],
      lwd=1.5,
      )
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'
legend("topright",
      fill = core_palette2[seq(1,101,10)],
      legend = heatlegend,
      bg = "white",
      y.intersp = 0.5,
)

```



Comparisons of all samples to hilltop  
 Leaf endophytes compared to hilltop  
 Leaf comparison to hilltop in euclidean space (NMS)

In [273]: `load('deseq95.rda')`

```
## make our dataframe of spatial coordinates and BC index:  

leaf95 <- subset_samples(deseq95, Library == 'L') ## general leaf phyloseq obj  

leafHel95 <- subset_samples(leaf95, Host_genus_species == 'Helicia_formosana') ## heli
```

```

bb <- t(otu_table(leafHel95)) ## get otu table for Helicia leaves
bb[bb > 0] <- 1 ## presence/absence
cc <- vegdist(bb, method='bray') ## makes a triangular association matrix
dd <- as.matrix(cc) ## convert to full, symmetric matrix
leafheltopBC <- dd[, '72leaf'] ## extract only the comparisons to our hilltop Helicia s

all(names(leafheltopBC) == rownames(sample_data(leafHel95))) ## check, compatible with

```

TRUE

In [229]: ## map a dataframe with the info we want to plot:

```

mapBC <- cbind(sample_data(leafHel95)[,c('X','Y')], leafheltopBC)
colnames(mapBC)[3] <- 'BC'
## order by BC dissimilarity
mapBC <- mapBC[order(mapBC$BC),]
head(mapBC)

```

	X	Y	BC
72leaf	183	223	0.0000000
99leaf	183	23	0.6097561
127leaf	40	103	0.6226415
97leaf	180	21	0.6470588
8leaf	360	227	0.6875000
33leaf	423	360	0.6923077

In [230]: mapBCrev <- mapBC[rev(rownames(mapBC)),]

In [235]: ## make a heat map palette useing colorbrewer

```

my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)
## need an "adapter", for our BC values to this heat map color scheme:
BCroundup <- rev(round(mapBC$BC*100+1))

options(repr.plot.width = 10, repr.plot.height = 7)

#png('hilltopmap.png')

```

```

topo <- readPNG('/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png') ## load topo map
plot(1, type='n',
      xlim=c(0,500),
      ylim=c(0,500),
      xlab = '',
      ylab = '',
      asp=1,
      main='Helicia formosana leaves, comparison to Hilltop')
## blank plot
rasterImage(topo,0,0,500,500) ## add raster of our plot

## add Helicia points, colored by similarity
points(mapBCrev[,c('X','Y')],
```

```

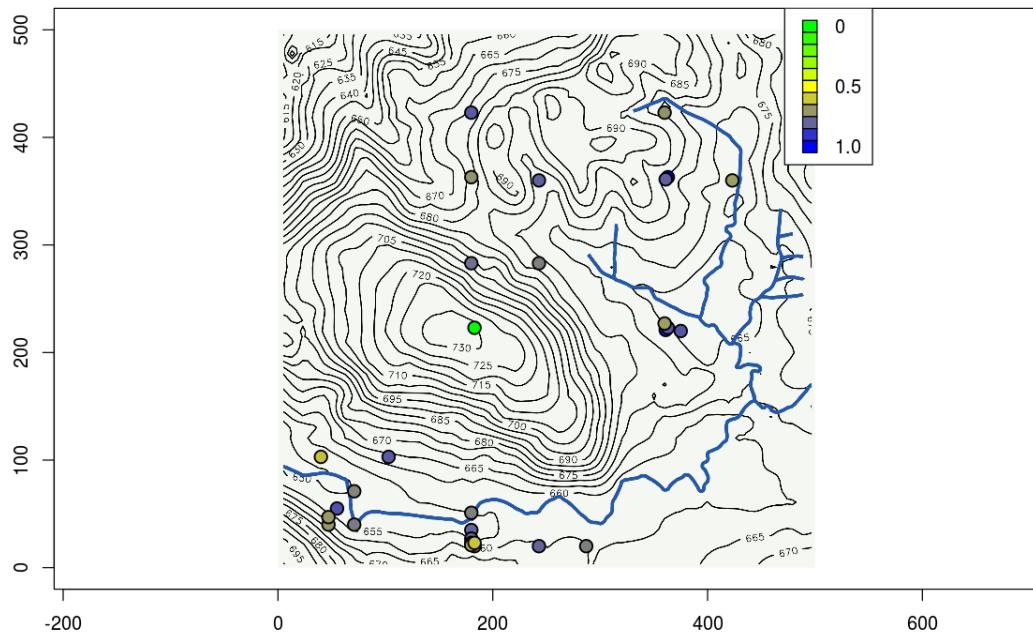
      pch=21,
      cex=1.5,
      bg = my_palette2[BCroundup],
      lwd=1.5,
    )

#draw.circle(x=183, y=223, radius=200, border="red", lwd=2, lty=2) ## use the plotrix pac

## make a legend:
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'
legend("topright",
      fill = my_palette2[seq(1,101,10)],
      legend = heatlegend,
      bg = "white",
      y.intersp = 0.5,
)
#dev.off()

```

**Helicia formosana leaves, comparison to Hilltop**



Leaf comparison to hilltop in BC dissimilarity space (NMS)

Where does this hilltop tend to be when we do an NMS ordination of the BC distances?

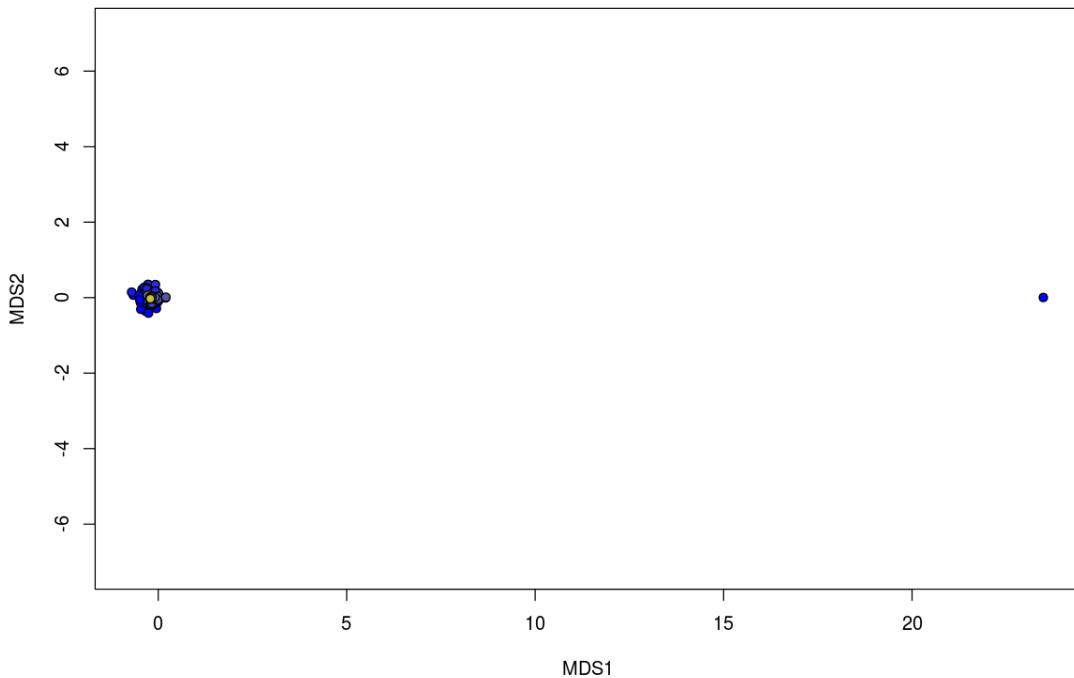
```
In [274]: leafOTU <- t(otu_table(leaf95))
          leafOTU[leafOTU > 0] <- 1
          leafMDS <- metaMDS(leafOTU)

Run 0 stress 0.1260773
Run 1 stress 0.126222
... Procrustes: rmse 0.008143772 max resid 0.03413517
Run 2 stress 0.1260959
... Procrustes: rmse 0.006410951 max resid 0.03108768
Run 3 stress 0.1259856
... New best solution
... Procrustes: rmse 0.006827669 max resid 0.0181532
Run 4 stress 0.1265765
Run 5 stress 0.1262299
... Procrustes: rmse 0.008884247 max resid 0.02344429
Run 6 stress 0.1259681
... New best solution
... Procrustes: rmse 0.002355039 max resid 0.01238553
Run 7 stress 0.1260302
... Procrustes: rmse 0.006796167 max resid 0.0193296
Run 8 stress 0.126221
... Procrustes: rmse 0.008384131 max resid 0.01884438
Run 9 stress 0.1261614
... Procrustes: rmse 0.008310365 max resid 0.02453497
Run 10 stress 0.1262036
... Procrustes: rmse 0.008942904 max resid 0.02465052
Run 11 stress 0.1262609
... Procrustes: rmse 0.006660805 max resid 0.01839853
Run 12 stress 0.1261559
... Procrustes: rmse 0.006421822 max resid 0.03125817
Run 13 stress 0.126263
... Procrustes: rmse 0.007207525 max resid 0.02365601
Run 14 stress 0.126007
... Procrustes: rmse 0.004006621 max resid 0.0165842
Run 15 stress 0.12611
... Procrustes: rmse 0.005868089 max resid 0.02346685
Run 16 stress 0.1260058
... Procrustes: rmse 0.002647116 max resid 0.01497188
Run 17 stress 0.126162
... Procrustes: rmse 0.007435341 max resid 0.02237596
Run 18 stress 0.1262586
... Procrustes: rmse 0.008244743 max resid 0.03657024
Run 19 stress 0.1260916
... Procrustes: rmse 0.006237917 max resid 0.03297871
Run 20 stress 0.1260675
... Procrustes: rmse 0.004404558 max resid 0.03062487
```

```
*** No convergence -- monoMDS stopping criteria:  
20: stress ratio > sratmax
```

```
In [276]: allBC <- as.matrix(vegdist(leafOTU))[, "72leaf", drop=FALSE]  
allBCroundup <- round(allBC*100+1) ## turn our allBC values into heatmap values  
my_palette2 <- colorRampPalette(c("green", "yellow", "blue"))(n = 101)
```

```
In [278]: plot(leafMDS$points,  
           pch=21,  
           bg = my_palette2[allBCroundup[,1]],  
           asp=1  
)
```



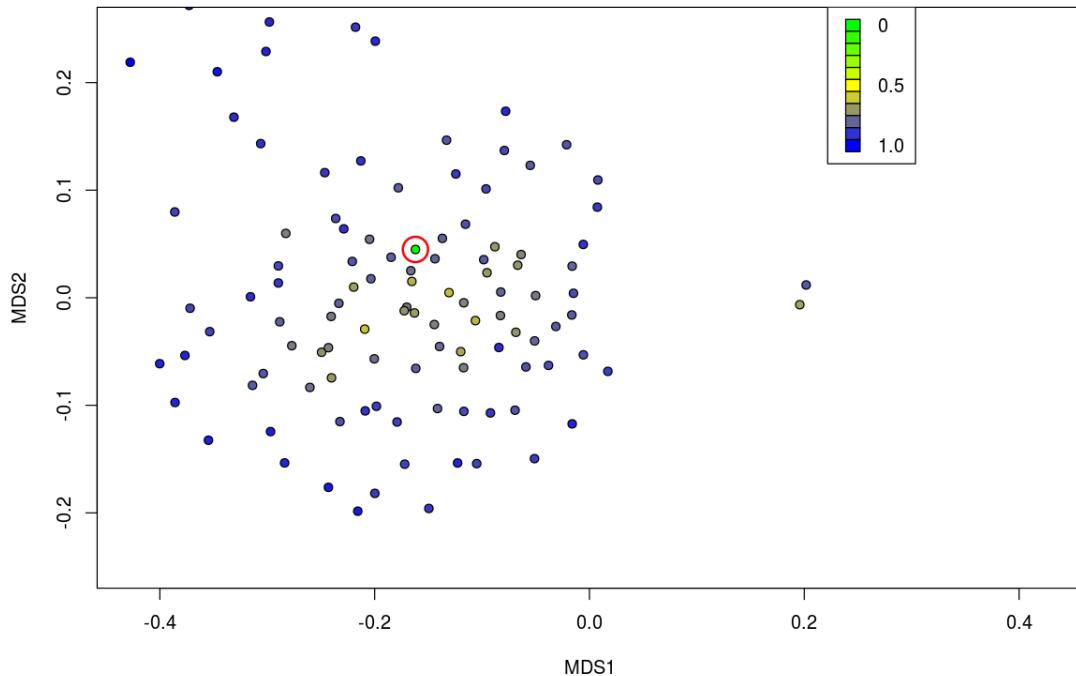
Zoom in, without the outlier:

```
In [281]: #png('hilltopNMS.png')  
plot(leafMDS$points,  
      pch=21,  
      bg = my_palette2[allBCroundup[,1]],  
      xlim = c(-0.25,0.25),  
      ylim = c(-0.25,0.25),  
      asp=1  
)
```

```

        legend("topright",
               fill = my_palette2[seq(1,101,10)],
               legend = heatlegend,
               bg = "white",
               y.intersp = 0.5,
)
points(leafMDS$points['72leaf',1],
       leafMDS$points['72leaf',2],
       pch = 1,
       cex = 3,
       col = 'red',
       lwd=2,
)
#dev.off()

```



```

In [285]: #identify(leafMDS$points) ## ID the outlier point
          leafMDS$points[90,drop=FALSE] ## get its dataframe position/rowname

leafNoOut <- leafOTU[row.names(leafOTU) != "67leaf",] ## get rid of ourlier row

```

```
In [286]: leafMDSnoOut <- metaMDS(leafNoOut)
```

Run 0 stress 0.228814

Run 1 stress 0.2231124

```

... New best solution
... Procrustes: rmse 0.07646505  max resid 0.3547096
Run 2 stress 0.2248534
Run 3 stress 0.2254021
Run 4 stress 0.224686
Run 5 stress 0.2286426
Run 6 stress 0.2233411
... Procrustes: rmse 0.03047058  max resid 0.1988328
Run 7 stress 0.225404
Run 8 stress 0.2244533
Run 9 stress 0.2254235
Run 10 stress 0.2243584
Run 11 stress 0.2265294
Run 12 stress 0.2241343
Run 13 stress 0.2255818
Run 14 stress 0.2233027
... Procrustes: rmse 0.03012185  max resid 0.2012423
Run 15 stress 0.2246742
Run 16 stress 0.2256866
Run 17 stress 0.2248967
Run 18 stress 0.2258309
Run 19 stress 0.2241649
Run 20 stress 0.2249329
*** No convergence -- monoMDS stopping criteria:
  4: no. of iterations >= maxit
  16: stress ratio > sratmax

```

```

In [276]: allBC <- as.matrix(vegdist(leafNoOut))[, "72leaf", drop=FALSE]
           allBCroundup <- round(allBC*100+1) ## turn our allBC values into heatmap values
           my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)

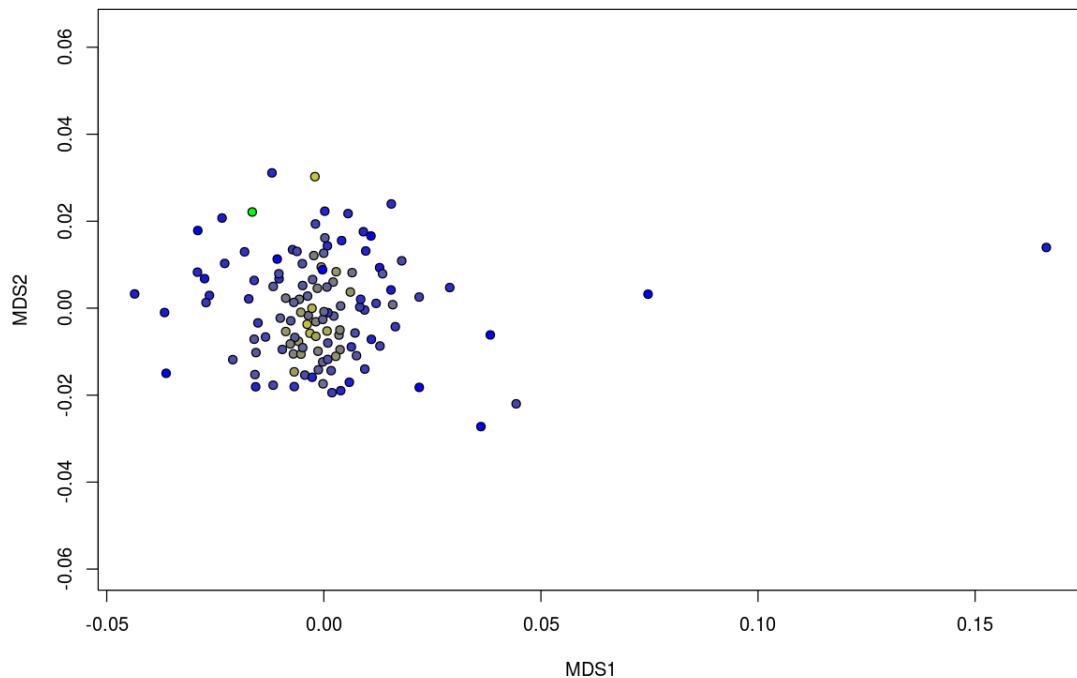
```

```

In [289]: plot(leafMDSnoOut$points,
            pch=21,
            bg = my_palette2[allBCroundup[,1]],
            asp=1,
            main='Leaf comparison to hilltop, without outliers'
        )

```

**Leaf comparison to hilltop, without outliers**



Wood endophytes compared to hilltop  
Wood comparison to hilltop in euclidean space (NMS)

```
In [264]: woodOTU <- t(otu_table(wood95))
woodOTU[woodOTU > 0] <- 1

woodMDS <- metaMDS(woodOTU)

Run 0 stress 0.2752839
Run 1 stress 0.2772215
Run 2 stress 0.2731556
... New best solution
... Procrustes: rmse 0.08576075 max resid 0.3949623
Run 3 stress 0.2717249
... New best solution
... Procrustes: rmse 0.03333857 max resid 0.1799176
Run 4 stress 0.2841524
Run 5 stress 0.275739
Run 6 stress 0.2719552
... Procrustes: rmse 0.008702442 max resid 0.06319884
Run 7 stress 0.2831439
Run 8 stress 0.2774238
Run 9 stress 0.2758084
```

```

Run 10 stress 0.279804
Run 11 stress 0.2743058
Run 12 stress 0.2816734
Run 13 stress 0.2783536
Run 14 stress 0.286105
Run 15 stress 0.2771506
Run 16 stress 0.2769533
Run 17 stress 0.2726776
Run 18 stress 0.2741197
Run 19 stress 0.2743129
Run 20 stress 0.2755739
*** No convergence -- monoMDS stopping criteria:
  20: stress ratio > sratmax

```

In [259]: *## map a dataframe with the info we want to plot:*

```

mapBC <- cbind(sample_data(woodHel95)[,c('X','Y')], woodheltopBC)
colnames(mapBC)[3] <- 'BC'
## order by BC dissimilarity
mapBC <- mapBC[order(mapBC$BC),]
head(mapBC)

```

	X	Y	BC
72w	183	223	0.0000000
28w	363	363	0.7076923
25w	361	361	0.7404580
96w	181	21	0.7678571
14w	375	220	0.7964072
99w	183	23	0.7966102

In [260]: `mapBCrev <- mapBC[rev(rownames(mapBC)),]`

In [262]: *## make a heat map palette useing colorbrewer*

```

my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)
## need an "adapter", for our BC values to this heat map color scheme:
BCroundup <- rev(round(mapBC$BC*100+1))

options(repr.plot.width = 10, repr.plot.height = 7)

#png('hilltopmap.png')

```

```

topo <- readPNG('/home/daniel/Documents/taiwan/taiwan_combined_stats/topo.png') ## load
plot(1, type='n',
      xlim=c(0,500),
      ylim=c(0,500),
      xlab = '',
      ylab = '',
      asp=1,
      main='Helicia formosana wood, comparison to Hilltop'

```

```

) ##blank plot
rasterImage(topo,0,0,500,500) ## add raster of our plot

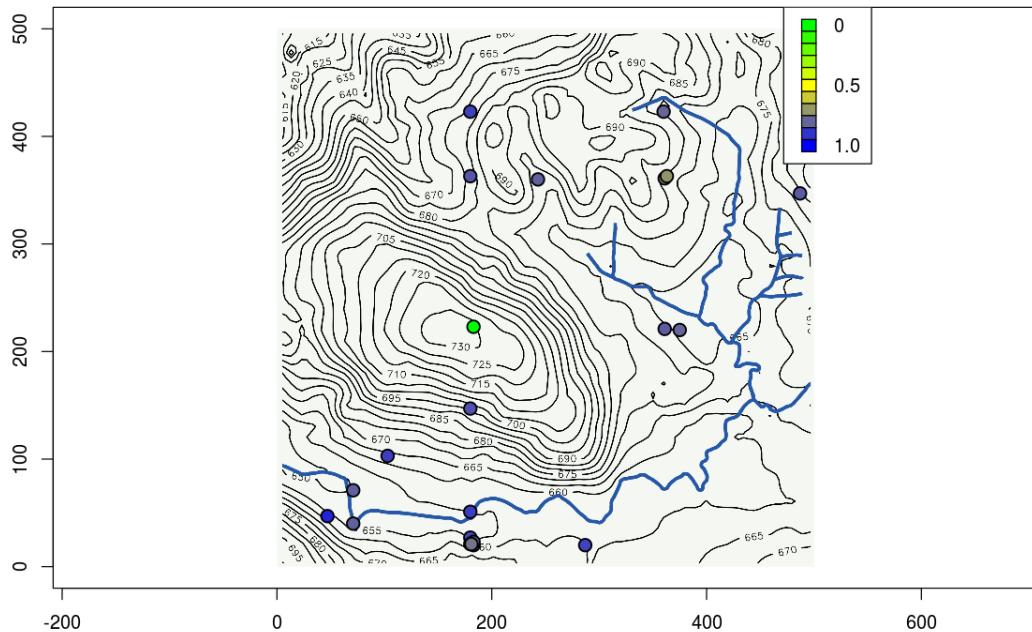
## add Helicia points, colored by similarity
points(mapBCrev[,c('X','Y')], 
       pch=21,
       cex=1.5,
       bg = my_palette2[BCroundup],
       lwd=1.5,
       )

#draw.circle(x=183, y=223, radius=200, border="red", lwd=2, lty=2) ## use the plotrix pac

## make a legend:
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'
legend("topright",
       fill = my_palette2[seq(1,101,10)],
       legend = heatlegend,
       bg = "white",
       y.intersp = 0.5,
       )
#dev.off()

```

### Helicia formosana wood, comparison to Hilltop



Map wood comparison to hilltop in BC dissimilarity space (NMS)

```
In [269]: woodOTU <- t(otu_table(wood95))
woodOTU[woodOTU > 0] <- 1

woodMDS <- metaMDS(woodOTU)

allBC <- as.matrix(vegdist(woodOTU))[, "72w", drop=FALSE]
allBCroundup <- round(allBC*100+1) ## turn our allBC values into heatmap values
my_palette2 <- colorRampPalette(c("green","yellow","blue"))(n = 101)

Run 0 stress 0.2752839
Run 1 stress 0.2749867
... New best solution
... Procrustes: rmse 0.05399962 max resid 0.3481395
Run 2 stress 0.2791709
Run 3 stress 0.2768793
Run 4 stress 0.2742886
... New best solution
... Procrustes: rmse 0.08608061 max resid 0.3459764
Run 5 stress 0.2734657
... New best solution
... Procrustes: rmse 0.01634504 max resid 0.08382424
```

```

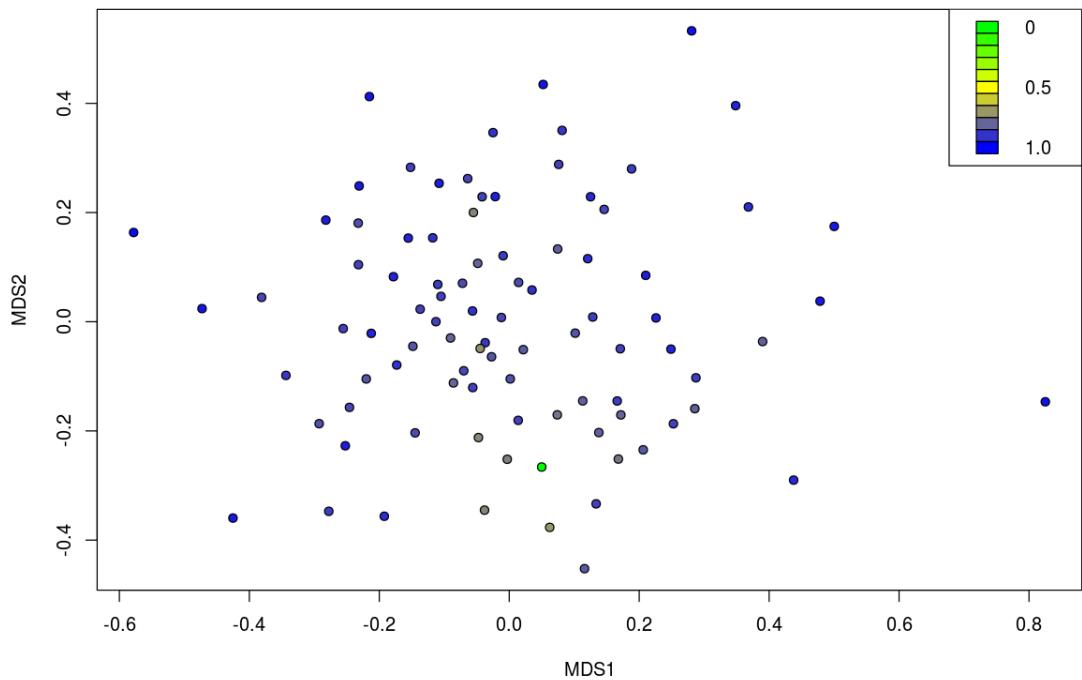
Run 6 stress 0.2764642
Run 7 stress 0.2736337
... Procrustes: rmse 0.04964461 max resid 0.334079
Run 8 stress 0.2779225
Run 9 stress 0.2891091
Run 10 stress 0.2746194
Run 11 stress 0.2731145
... New best solution
... Procrustes: rmse 0.02257999 max resid 0.1305221
Run 12 stress 0.2806462
Run 13 stress 0.2805967
Run 14 stress 0.2769614
Run 15 stress 0.2937179
Run 16 stress 0.2744863
Run 17 stress 0.2733944
... Procrustes: rmse 0.02474187 max resid 0.1829851
Run 18 stress 0.2972708
Run 19 stress 0.2782096
Run 20 stress 0.2958267
*** No convergence -- monoMDS stopping criteria:
  1: no. of iterations >= maxit
  19: stress ratio > sratmax

```

```

In [270]: plot(woodMDS$points,
            pch=21,
            bg = my_palette2[allBCroundup[,1]],
            #      asp=1,
            #ylim=c(-0.2,0.2),
            #xlim=c(-0.5,0.5),
            )
heatlegend <- vector(length=11)
heatlegend[] <- ''
heatlegend[1] <- '0'
heatlegend[6] <- '0.5'
heatlegend[11] <- '1.0'
legend("topright",
       fill = my_palette2[seq(1,101,10)],
       legend = heatlegend,
       bg = "white",
       y.intersp = 0.5,
     )

```



Comparison of dissimilarity at all levels: all hosts, *Helicia*, and *Helicia* core mycobiome

```
In [318]: load('deseq95.rda')

## leaves, at large:
leaf95 <- subset_samples(deseq95, Library == 'L')
leafOTU <- t(otu_table(leaf95))
leafOTU[leafOTU > 0] <- 1
aa <- vegdist(leafOTU)
bb <- c(aa)
print('all hosts, all fungi, leaves:')
mean(bb)
sd(bb)

## wood, at large:
wood95 <- subset_samples(deseq95, Library == 'W')
woodOTU <- t(otu_table(wood95))
woodOTU[woodOTU > 0] <- 1
cc <- vegdist(woodOTU)
dd <- c(cc)
print('all hosts, all fungi, wood:')
mean(dd)
```

```

sd(dd)

## Helicia only, leaves:
leafHel95 <- subset_samples(deseq95, Library == 'L' & Host_genus_species == "Helicia_f"
leafOTU <- t(otu_table(leafHel95))
leafOTU[leafOTU > 0] <- 1
aa <- vegdist(leafOTU)
bb <- c(aa)
print('Helicia, all fungi, leaves:')
mean(bb)
sd(bb)

## Helicia only, wood:
woodHel95 <- subset_samples(deseq95, Library == 'W' & Host_genus_species == "Helicia_f"
woodOTU <- t(otu_table(woodHel95))
woodOTU[woodOTU > 0] <- 1
cc <- vegdist(woodOTU)
dd <- c(cc)
print('Helicia, all fungi, wood:')
mean(dd)
sd(dd)

## helicia leaves, core:
leafHel95 <- subset_samples(deseq95, Library == 'L' & Host_genus_species == "Helicia_f
load('helleaffung.rda')
## get the average BC of all helicia leaf points from this core:
## get the BC distance of all helicia points from the core:
aa <- t(otu_table(leafHel95))
aa <- aa[,helleaffung] ## subset OTU to just these rows,
aa <- aa[rowSums(aa)>0,] ## remove zero rows
aa[aa > 0] <- 1 ## PA
bray <- c(vegdist(aa)) ##
## new results:
print('Helicia, core fungi, leaves:')
mean(bray, na.rm=TRUE)
sd(bray, na.rm=TRUE)
lcb <- bray

## Helicia core, wood:
woodHel95 <- subset_samples(deseq95, Library == 'W' & Host_genus_species == "Helicia_f
load('helwoodfung.rda')
## get the average BC of all helicia wood points from this core:
## get the BC distance of all helicia points from the core:
aa <- t(otu_table(woodHel95))
aa <- aa[,helwoodfung] ## subset OTU to just these rows
aa[aa > 0] <- 1 ## PA
bray <- c(vegdist(aa)) ##
print('Helicia, core fungi, wood:')

```

```

mean(bray)
sd(bray) ## .17
wcb <- bray

## Helicia leaf, non-core,
aa <- t(otu_table(leafHel95))
aa <- aa[,!(colnames(aa) %in% helleaffung)] ## get rid of cores
aa[aa > 0] <- 1 ## PA
bray <- c(vegdist(aa)) ##
print('Helicia, non-core fungi, leaves:')
mean(bray) ## .86
sd(bray) ## .11

## Helicia wood, non-core:
aa <- t(otu_table(woodHel95))
aa <- aa[,!(colnames(aa) %in% helwoodfung)] ## get rid of cores
aa[aa > 0] <- 1
bray <- c(vegdist(aa))
print('Helicia, non-core fungi, wood:')
mean(bray)
sd(bray)

## are the two core BC means different?
t.test(wcb,lcb)

[1] "all hosts, all fungi, leaves:"
```

0.897387332069972  
0.0883387424827165

```
[1] "all hosts, all fungi, wood:"
```

0.873903915174344  
0.0694090626138818

```
[1] "Helicia, all fungi, leaves:"
```

0.799032377735972  
0.107349929451309

```
[1] "Helicia, all fungi, wood:"
```

0.80353154017327  
0.0607413094547431

```

[1] "Helicia, core fungi, leaves:"
```

0.383548761116857  
0.167949494026408

```

[1] "Helicia, core fungi, wood:"
```

0.377571947052467  
0.171537476539615

```

[1] "Helicia, non-core fungi, leaves:"
```

0.88515913522036  
0.0777586905082737

```

[1] "Helicia, non-core fungi, wood:"
```

0.833499242587381  
0.0628368905465071

#### Welch Two Sample t-test

```

data: wcb and lcb
t = -0.42599, df = 469.92, p-value = 0.6703
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-0.03354723  0.02159360
sample estimates:
mean of x mean of y
0.3775719 0.3835488
```

```

In [335]: load('deseq95.rda')
## leaves, at large:
leaf95 <- subset_samples(deseq95, Library == 'L')
leafOTU <- t(otu_table(leaf95))
leafOTU[leafOTU > 0] <- 1
aa <- c(vegdist(leafOTU))
BClist <- aa
namez <- rep('leaf95', length(aa))

## wood, at large:
wood95 <- subset_samples(deseq95, Library == 'W')
```

```

woodOTU <- t(otu_table(wood95))
woodOTU[woodOTU > 0] <- 1
aa <- c(vegdist(woodOTU))
BClist <- c(BClist, aa)
namez <- c(namez, rep('wood95', length(aa)))

## leaves, for Helicia:
leafHel95 <- subset_samples(deseq95, Library == 'L' & Host_genus_species == "Helic"
leafOTU <- t(otu_table(leafHel95))
leafOTU[leafOTU > 0] <- 1
aa <- c(vegdist(leafOTU))
BClist <- c(BClist, aa)
namez <- c(namez, rep('leafHel95', length(aa)))

## wood, for Helicia:
woodHel95 <- subset_samples(deseq95, Library == 'W' & Host_genus_species == "Helic
woodOTU <- t(otu_table(woodHel95))
woodOTU[woodOTU > 0] <- 1
aa <- c(vegdist(woodOTU))
BClist <- c(BClist, aa)
namez <- c(namez, rep('woodHel95', length(aa)))

## Helicia-only, core, wood, avg BC
woodHel95 <- subset_samples(deseq95, Library == 'W' & Host_genus_species == "Helic
load('helwoodfung.rda')
## get the average BC of all helicia wood points from this core:
aa <- t(otu_table(woodHel95))
aa <- aa[,helwoodfung] ## subset OTU to just these rows
aa[aa > 0] <- 1 ## PA
aa <- c(vegdist(aa)) ##
BClist <- c(BClist, aa)
namez <- c(namez, rep('woodHel95core', length(aa)))

## Helicia-only, non-core, wood, avg BC
aa <- t(otu_table(woodHel95))
aa <- aa[,!(colnames(aa) %in% helwoodfung)] ## get rid of cores
aa[aa > 0] <- 1 ## PA
aa <- c(vegdist(aa)) ##
BClist <- c(BClist, aa)
namez <- c(namez, rep('woodHel95nocore', length(aa)))

## helicia-only, leaves, core
leafHel95 <- subset_samples(deseq95, Library == 'L' & Host_genus_species == "Helic
load('helleaffung.rda')
## get the average BC of all helicia leaf points from this core:
aa <- t(otu_table(leafHel95))
aa <- aa[,helleaffung] ## subset OTU to just these rows
aa <- aa[rowSums(aa)>0,] ## remove zero sum rows

```

```

aa[aa > 0] <- 1 ## PA
aa <- c(vegdist(aa)) ##
BClist <- c(BClist, aa)
namez <- c(namez, rep('leafHel95core', length(aa)))

## Helicia-only, non-core, leaf, avg BC
aa <- t(otu_table(leafHel95))
aa <- aa[, !(colnames(aa) %in% helleaffung)] ## get rid of cores
aa[aa > 0] <- 1 ## PA
aa <- c(vegdist(aa)) ##
BClist <- c(BClist, aa)
namez <- c(namez, rep('leafHel95nocore', length(aa)))

```

```

In [336]: bcp <- data.frame(BClist, namez)
          ## control the plotting order:
bcp$namez <- factor(bcp$namez, levels=c('leaf95',
                                             'wood95',
                                             'leafHel95',
                                             'leafHel95core',
                                             'leafHel95nocore',
                                             'woodHel95',
                                             'woodHel95core',
                                             'woodHel95nocore'
                                             ))

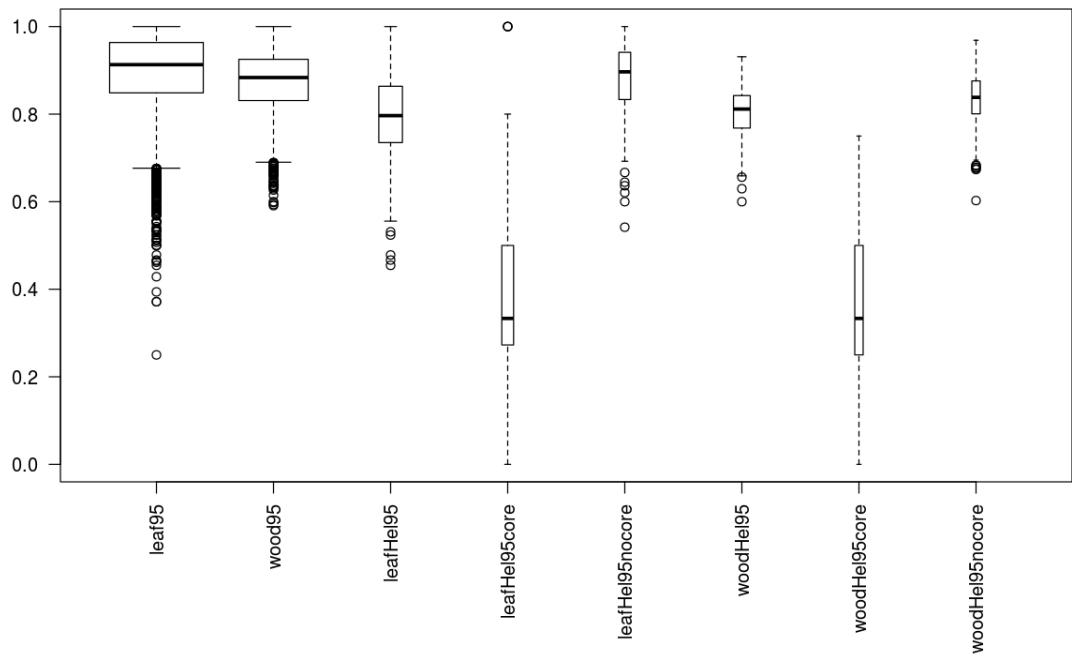
```

```

In [337]: ## control the box size, make proportions a little more
          ## realistic:
levelProportions <- c(123, 91, 31, 15.5, 15.5, 22, 11, 11)/100
names(levelProportions) <- c('leaf95', 'wood95', 'leafHel95', 'leafHel95core', 'leafHe

par(mar=c(10,3,4,2)) # increase y-axis margin.
boxBC <- boxplot(BClist~namez, data=bcp,
                  las = 2,
                  width = levelProportions
)

```



In table form:

organ	all hosts, all endophytes	<i>Helicia</i> , all-endophytes	<i>Helicia</i> , core-fungi	<i>Helicia</i> , non-core
leaf	0.897(+/-0.088)	0.799(+/-0.107)	0.383(+/-0.168)	0.885(+/-0.078)
wood	0.874(+/-0.069)	0.804(+/-0.061)	0.378(+/-0.172)	0.833(+/-0.063)

The numbers have changed a bit, but the same basic story remains here, so quote myself again:

The story here: endophytes on the whole are highly variable - when all host trees are looked at, the average dissimilarity between any two trees is high, bc=.9. Samples become more similar on average, when we constrain to a single host, to around bc=.8, for wood and leaves. Non-core fungi behave similarly, bc=.86 for both leaves and wood. But we do see a core of fungi, 8 species in leaves and 10 in wood, that behave differently. In leaves these "core fungi" seem to have a strong spatial pattern, namely that they seem to be most present in the southern valley, and are often completely missing in other areas of the study. In wood, they are more "loyal", and coexist more reliably with *Helicia* throughout the plot.

# One big biome table

Since I am hoping to make comparisons of wood and leaf endophyte environmental patterns, I need to combine these datasets early in the biomformatics pipeline, to make them as comparable as possible. We'll try to stick to the [usearch \(uparse\) \(http://drive5.com/usearch/\)](http://drive5.com/usearch/) pipeline for the process, as much as possible.

---

That was a year ago.

Since then my car has been stolen, with my laptop in it. While most of the important graphics, etc, were backed up, I lost some of the computationally expensive intermediate files necessary to repopulate.

So to make this notebook useable to reviewers and readers, I'll be picking through this process again, and the downstream analysis.

## Table of contents

[Work environment](#)

[Rearranging barcodes](#)

[Trimming reads](#)

[Merging paired-end reads](#)

- [Leaf reads](#)

- [Wood reads](#)

[Visualizing merged read qualities](#)

- [Make quality score charts](#)

- [Leaf read qualities](#)

- [Wood read qualities](#)

[Quality filtering reads](#)

[Convert fastq files to fasta format](#)

[Demultiplex leaf reads](#)

[Clip primers](#)

[Checking for chimeras](#)

[Finding ITS1 region](#)

[OTU clustering](#)

- [Dereplication and Sorting of reads](#)

- [Cluster reads](#)

- [Assign unique names to OTU clusters](#)

- [Assign taxonomy](#)

- [Make biom table](#)

[Formatting Biom table and adding metadata](#)

- [Change biom taxonomy metadata format](#)

- [Add sample metadata](#)

## Work environment

Working directory, on my machine:

```
In [2]: cd /home/daniel/Documents/Taiwan_data/combined/combo biome
```

We'll be using the [usearch \(uparse\) \(http://drive5.com/usearch/\)](http://drive5.com/usearch/) pipeline, version v8.0.1623\_i86linux32, on the University of Oregon's Talapas computing cluster.

## Rearranging barcodes

We need to merged paired end sequences of the leaves and wood. But before we can do this, there are several steps. First, the leaf study reads include a split 6+6 bp barcode scheme for identifying reads, so these need to be clipped from one read and combined on the other. I wrote a python script for this:

```
In [4]: cat scripts/BCunsplit.py
```

```
#!/usr/bin/env python3

## lets try to take two unpaired read files, cut out the bp from the reverse,
## and tack it onto the forward.
## have to preserve the fastq format so that pandaseq can do unsplit3.py forward_reads reverse_reads

#The first six bps and quality ratings of the reverse reads should be chopped off and placed after
#the first six bps of the forward reads and quality ratings. For use with fastq files. It will spit
#out two files, with the names: "rearranged_[your original forward and reverse read file names].fastq".

import itertools ##to let us jump around
from sys import argv

script, forward_file, reverse_file = argv
```

I have details on how I used this [here \(https://github.com/danchurch/taiwan\\_dada2/blob/master/dada2pipeline.ipynb\)](https://github.com/danchurch/taiwan_dada2/blob/master/dada2pipeline.ipynb).

This outputs two files, "rearranged\_Roo\_R2.fastq" and "rearranged\_Roo\_R2.fastq". I did this in another directory, so we'll add some sym links here for convenience:

```
In [8]: ## leaves
ln -s /home/daniel/Documents/taiwan/taiwan_dada2/rearranged_leafR1.fastq reLea
ln -s /home/daniel/Documents/taiwan/taiwan_dada2/rearranged_leafR2.fastq reLea
```

## Trimming reads

Next we trim a little to make sure we're doing our alignments with high quality base calls. The sites for trimming are decided by looking at the raw reads ([see below](#)), and finding where quality begins to drop off. To trim, we'll use the [FASTX-toolkit](#) ([http://hannonlab.cshl.edu/fastx\\_toolkit/](http://hannonlab.cshl.edu/fastx_toolkit/)).

Our wood reads are already demultiplexed, so we don't have a single forward and reverse read file for all of our wood samples, like we do above with the leaves. So let's make a script for this:

```
In [ ]: ## trims.sh
#####
## wood reads live here:
woaddir=/home/daniel/Documents/taiwan/woodreads/
## working directory is here:
cd /home/daniel/Documents/taiwan/taiwan_combined_biom
##### R1 reads:
## home for trimmed R1 wood reads here:
R1trimdir='/home/daniel/Documents/taiwan/taiwan_combined_biom/trimmed_wood/R1/'

## trim just the R1s, output to their new home with new filename:
for i in $woaddir*_R1_*; do
    echo $i
    out=${R1trimdir}$(basename ${i/_001\.fastq/_trimmed\.fastq})
    fastx_trimmer -l 255 -i $i -o $out && echo $out
done

##### R2 reads:
## home for trimmed R2 wood reads here:
R2trimdir='/home/daniel/Documents/taiwan/taiwan_combined_biom/trimmed_wood/R2/'

## trim just the R2s, output to their new home with new filename:
for j in $woaddir*_R2_*; do
    echo $j
    out=${R2trimdir}$(basename ${j/_001\.fastq/_trimmed\.fastq})
    fastx_trimmer -l 210 -i $j -o $out && echo $out
done

In [2]: ## leaves. These lengths were decided by Roo. They are all still in one pile:
fastx_trimmer -l 263 -i reLeafR1.fastq -o Roo_R1_trimmed.fastq
fastx_trimmer -l 170 -i reLeafR2.fasta -o Roo_R2_trimmed.fasta
```

## Merging paired-end reads

Unfortunately, I no longer have access to the 64-bit version of usearch 8.1. So let's use the 32-bit version:

```
In [7]: usearch
```

```
usearch v8.1.1861_i86linux32, 4.0Gb RAM (8.0Gb total), 4 cores  
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.  
http://drive5.com/usearch (http://drive5.com/usearch)
```

```
License: danchurchthomas@gmail.com
```

### Leaf reads

The leaf files are too large to be handled by the 32-version. We don't want to demultiplex yet, this will be messy with the split barcodes. So let's break up the leaf files into smaller ones and merge these.

```
In [ ]: cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_lea  
for i in Roo_R*_*; do  
    ls $i  
    split -d -l 500000 $i ${i/fastq/split}.fastq  
done
```

Makes a lot of files:

```
In [10]: cat leafSplitLs.txt
```

```
aln_3.txt  
leafSplitLs.txt  
Roo_R1_trimmed.split.fastq00  
Roo_R1_trimmed.split.fastq01  
Roo_R1_trimmed.split.fastq02  
Roo_R1_trimmed.split.fastq03  
Roo_R1_trimmed.split.fastq04  
Roo_R1_trimmed.split.fastq05  
Roo_R1_trimmed.split.fastq06  
Roo_R1_trimmed.split.fastq07  
Roo_R1_trimmed.split.fastq08  
Roo_R1_trimmed.split.fastq09  
Roo_R1_trimmed.split.fastq10  
Roo_R1_trimmed.split.fastq11  
Roo_R1_trimmed.split.fastq12  
Roo_R1_trimmed.split.fastq13  
Roo_R1_trimmed.split.fastq14  
Roo_R1_trimmed.split.fastq15  
Roo_R1_trimmed.split.fastq16  
Roo_R1_trimmed.split.fastq17
```

Merge these:

```
In [ ]: for forward in *_R1_*; do
    #ls $forward
    reverse=${forward/_R1_/_R2_}
    usearch -fastq_mergepairs $forward \
        -reverse $reverse \
        -fastq_maxdiffpct 40 \
        -alnout aln_3.txt \ ## oops, fix this if reused
        -report ../reports/$forward.report.txt \
        -fastqout ../merged/$forward.merged.fastq
    #ls $reverse
done
```

And put them back together.

```
In [ ]: cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed
cat * > leaf_trimmed_merged.fasta
```

### Wood reads

We don't have the same issue on the wood, because they are already demultiplexed, and the 32-bit version of usearch can handle these smaller files fine:

```
In [ ]: for forward in *fastq; do
    ls -l $forward
    reverse="..../$forward/R1/R2"
    usearch -fastq_mergepairs $forward \
        -reverse $reverse \
        -fastq_maxdiffpct 40 \
        -alnout aln_3.txt \ ## oops, fix this if reused
        -report ../reports/$forward.report.txt \
        -fastqout ./${forward}.merged.fastq
    echo $reverse
done
```

### Visualizing merged read qualities

#### Make quality score charts

Let's make some charts of our read quality, using fastx tools. First, compile the stats on each basepair:

```
In [ ]: #!/usr/bin/env bash

## leaf reads

cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/qCharts/
rawLeafReadsR1="/home/daniel/Documents/taiwan_supp/roo_reads/TaiwanFA_R1.fastq"
rawLeafReadsR2="/home/daniel/Documents/taiwan_supp/roo_reads/TaiwanFA_R2.fastq"
trimmedLeafReadsR1="/home/daniel/Documents/submissions/taibioinfo/taiwan_combi"
trimmedLeafReadsR2="/home/daniel/Documents/submissions/taibioinfo/taiwan_combi"
leafmerg="/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/t

## leaf quality stats:
fastx_quality_stats -i $rawLeafReadsR1 -o rawLeafReadsR1_fastxstats.txt
fastx_quality_stats -i $rawLeafReadsR2 -o rawLeafReadsR2_fastxstats.txt
fastx_quality_stats -i $trimmedLeafReadsR1 -o trimmedLeafReadsR1_fastxstats.txt
fastx_quality_stats -i $trimmedLeafReadsR2 -o trimmedLeafReadsR2_fastxstats.txt
fastx_quality_stats -i $leafmerg -o leafmerged_fastxstats.txt
```

For the wood, to visualize them as a whole, we'll combine reads of the steps we've done so far:

```
In [ ]: cat /home/daniel/Documents/taiwan_supp/wood_reads/*R1* > /home/daniel/Documents/trimmedWoodReadsR1.fastq
cat /home/daniel/Documents/taiwan_supp/wood_reads/*R2* > /home/daniel/Documents/trimmedWoodReadsR2.fastq
cat /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed
cat /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed
cat /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed
```

Compile the stats for these combined wood files:

```
In [ ]: #!/usr/bin/env bash

cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom

fastx_quality_stats -i rawWoodReadsR1.fastq -o qCharts/wood/rawWoodReadsR1_fastxstats.txt
fastx_quality_stats -i rawWoodReadsR2.fastq -o qCharts/wood/rawWoodReadsR2_fastxstats.txt
fastx_quality_stats -i trimmedWoodR1.fastq -o qCharts/wood/trimmedWoodR1_fastxstats.txt
fastx_quality_stats -i trimmedWoodR2.fastq -o qCharts/wood/trimmedWoodR2_fastxstats.txt
fastx_quality_stats -i woodMerged.fasta -o qCharts/wood/woodMerged_fastxstats.txt
```

Then make the actual graphics.

```
In [ ]: cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/qCharts
cd leaf

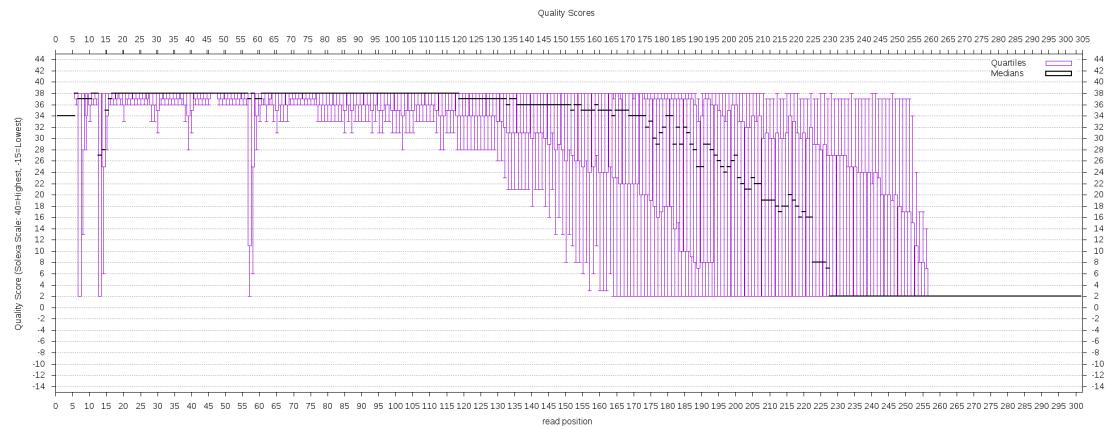
for i in *; do
    ./dan_fastx_plot.sh -i $i -o ${i/\.\txt/\.\png}
done

cd ../wood

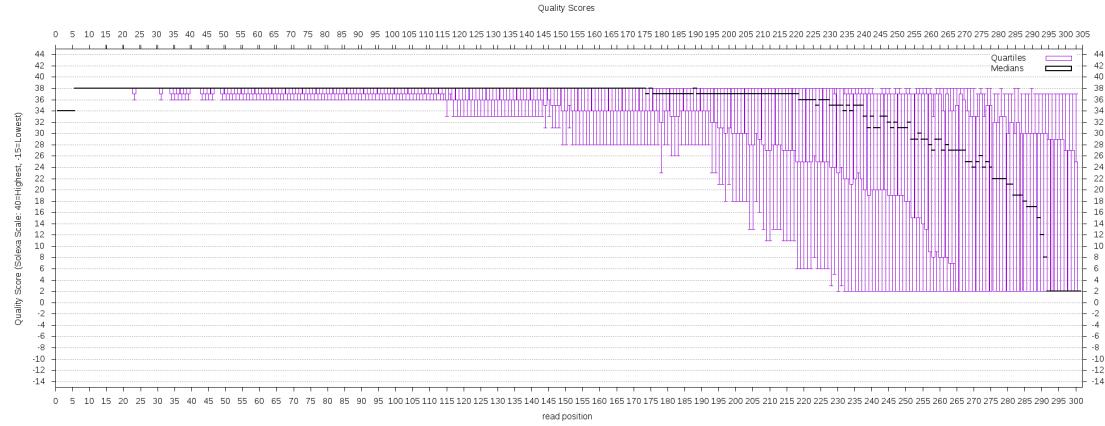
for i in *; do
    ./dan_fastx_plot.sh -i $i -o ${i/\.\txt/\.\png}
done
```

### Leaf read qualities

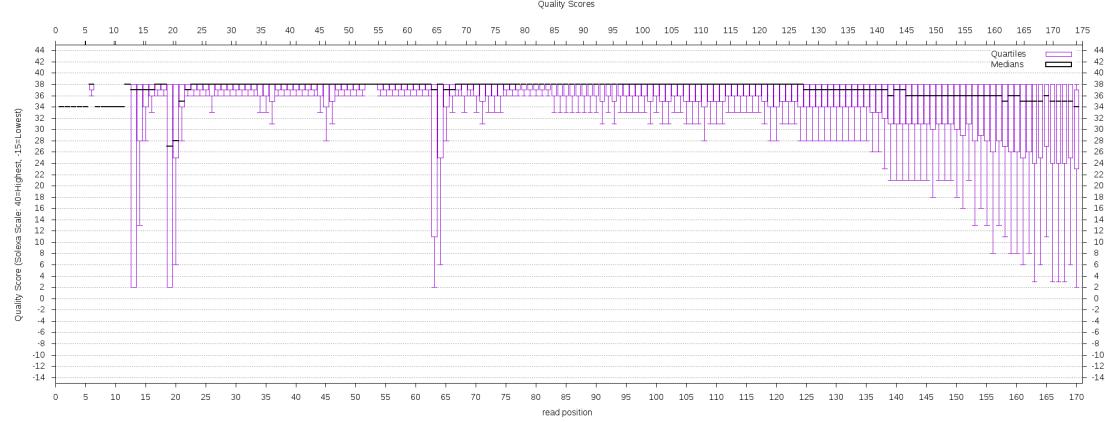
#### Forward raw reads:



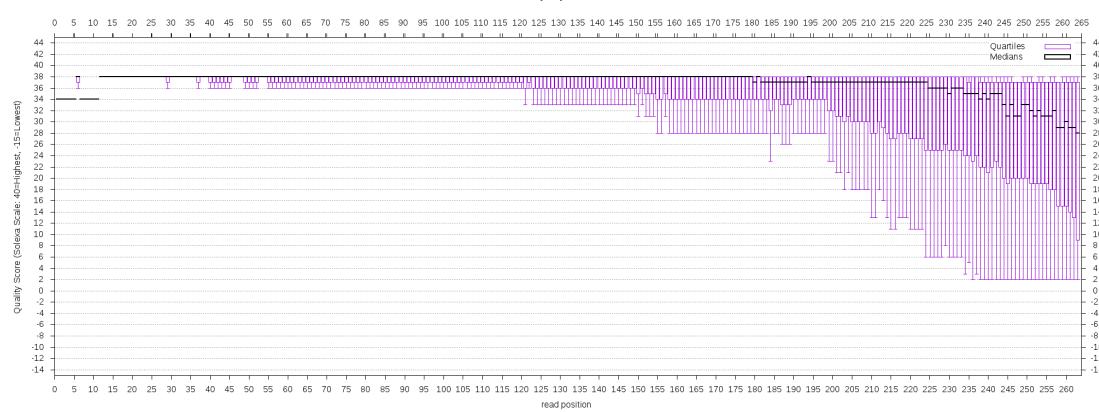
#### Reverse raw Leaf reads.



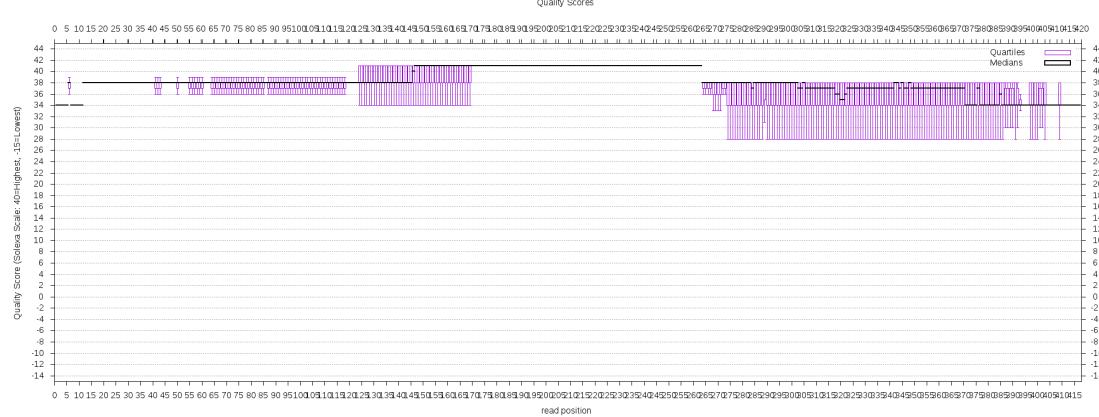
#### Forward trimmed Leaf reads.



### Reverse trimmed Leaf reads.

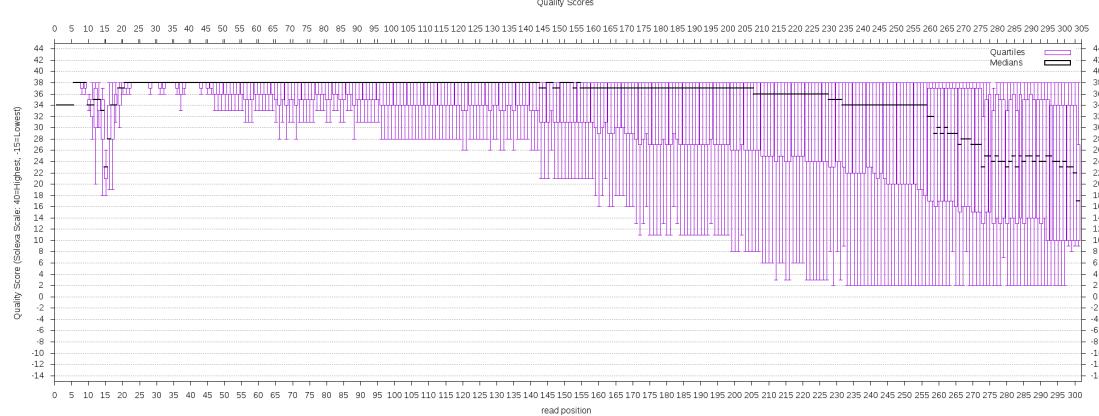


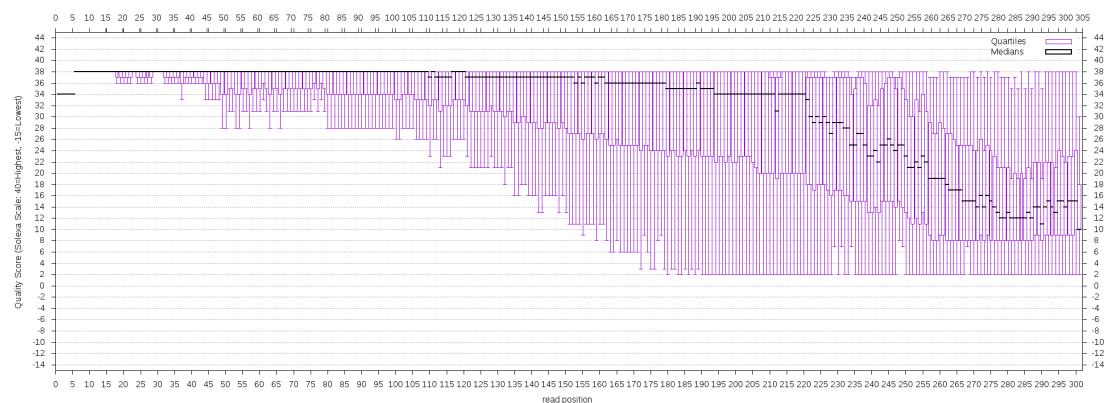
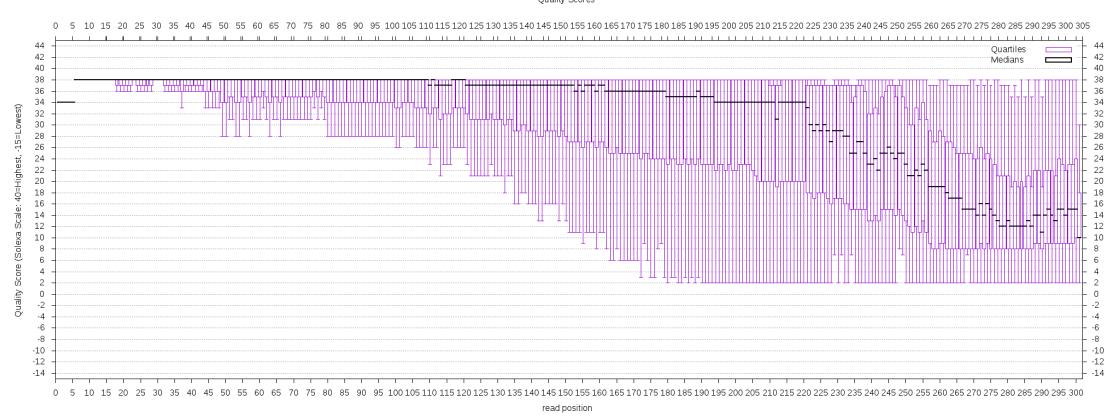
### And the merged leaf reads.



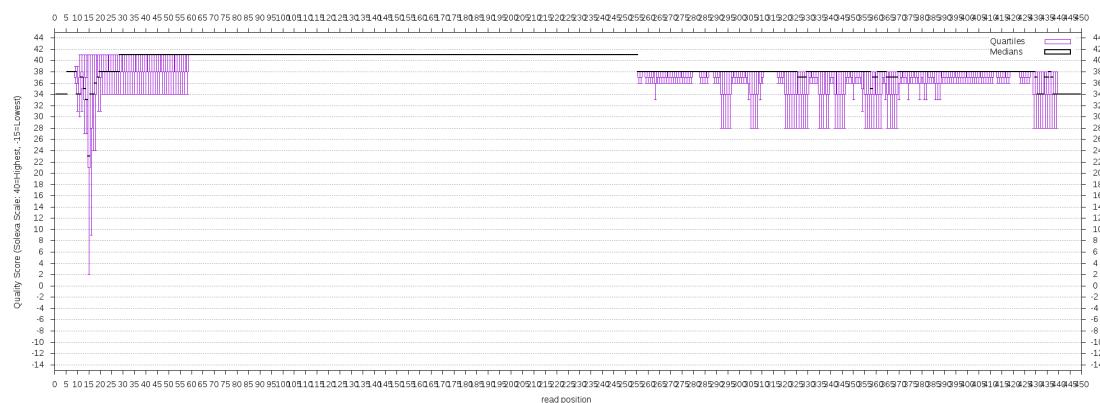
### Wood read qualities

#### Wood raw forward reads:



**Wood raw reverse reads:****Wood trimmed forward reads:****Wood trimmed reverse reads:**

### Wood merged reads:



## Quality filtering reads

USEARCH does quite a bit of filtering in the merging process, I think, based on how many reads from our wood samples were removed. But let's also use the USEARCH filtering program on both wood and leaf reads.

### Filter leaves

```
In [ ]: ## leaves
cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves
for i in *; do
    out=${i/.merged.fastq/.mergedfilt.fastq}
    #echo $i $out
    usearch -fastq_filter $i -fastq_maxee_rate .01 -fastqout "../filtered/$out"
done
```

```
In [4]: cat leaf filterStdout.txt
```

```
usearch v8.1.1861_i86linux32, 4.0Gb RAM (12.1Gb total), 4 cores
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.
http://drive5.com/usearch (http://drive5.com/usearch)
```

License: danchurchthomas@gmail.com

```
00:01 32Mb 100.0% Filtering, 97.5% passed
102021 FASTQ recs (102.0k)
99492 Converted (99.5k, 97.5%)
usearch v8.1.1861_i86linux32, 4.0Gb RAM (12.1Gb total), 4 cores
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.
http://drive5.com/usearch (http://drive5.com/usearch)
```

License: danchurchthomas@gmail.com

```
00:01 32Mb 100.0% Filtering, 98.2% passed
104711 FASTQ recs (104.7k)
102793 Converted (102.8k, 98.2%)
usearch v8.1.1861_i86linux32, 4.0Gb RAM (12.1Gb total), 4 cores
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.
```

### Filter wood

```
In [ ]: ## wood
cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed_
for i in *; do
    out=${i}.fastq.merged.fastq\|.merge\|.filt\.fastq
    usearch -fastq_filter $i -fastq_maxee_rate .01 -fastqout $out -notrunclabe
done
```

```
In [2]: cat wood filterStdout.txt

usearch v8.1.1861_i86linux32, 4.0Gb RAM (12.1Gb total), 4 cores
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.
http://drive5.com/usearch (http://drive5.com/usearch)

License: danchurchthomas@gmail.com

00:00 32Mb 100.0% Filtering, 100.0% passed
    61717 FASTQ recs (61.7k)
        61717 Converted (61.7k, 100.0%)
usearch v8.1.1861_i86linux32, 4.0Gb RAM (12.1Gb total), 4 cores
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.
http://drive5.com/usearch (http://drive5.com/usearch)

License: danchurchthomas@gmail.com

00:00 32Mb 100.0% Filtering, 100.0% passed
    44977 FASTQ recs (45.0k)
        44975 Converted (45.0k, 100.0%)
usearch v8.1.1861_i86linux32, 4.0Gb RAM (12.1Gb total), 4 cores
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.
```

### Convert fastq files to fasta format

Let's use [BBMap tools](https://sourceforge.net/projects/bbmap/) (<https://sourceforge.net/projects/bbmap/>) to do the conversion. FASTX toolbox has something for this also, but FASTX is a little brittle when dealing with fastq files in modern illumina quality scores, etc., and sometimes generates funny errors.

### Leaf reads

```
In [ ]: ## drop bbtools into a nearby directory. Java, so can't really put in bin fold
bb=/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/bbmap

## just one large leaf file
$bb/reformat.sh in=leaf_merged_filt.fastq \
    out=leaf_merged_filt.fasta \
    fastawrap=0 \
    &> makeLeafFasta.txt
```

Outputs from bbmap for this:

In [10]: cat makeLeafFasta.txt

```
java -ea -Xmx200m -cp /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/bbmap/current/ jgi.ReformatReads in=leaf_merged_filt.fastq out=leaf_af_merged_filt.fasta fastawrap=0
Executing jgi.ReformatReads [in=leaf_merged_filt.fastq, out=leaf_merged_filt.fasta, fastawrap=0]

Input is being processed as unpaired
Input: 14372164 reads 4600246850 bases
Output: 14372164 reads (100.00%) 4600246850 bases (100.00%)

Time: 151.584 seconds.
Reads Processed: 14372k 94.81k reads/sec
Bases Processed: 4600m 30.35m bases/sec
```

### Wood reads

In [5]: ## wood to fasta, lots of smaller files:

```
wfd=/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed
cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed_
for i in *; do
$bb/reformat.sh in=$i \
    out=$wfd${i/_R1_trimmed.mergefilt.fastq/.fasta} \
    fastawrap=0 \
    &>> ../../woodFastaStdout.txt
done
```

In [11]: cat woodFastaStdout.txt

```
java -ea -Xmx200m -cp /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/bbmap/current/ jgi.ReformatReads in=lanel-s160-index-AAGCACTG-GTGATCCANNNN-Dc-X_S160_L001_R1_trimmed.mergefilt.fastq out=/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed_wood/woodFasta/lanel-s160-index-AAGCACTG-GTGATCCANNNN-Dc-X_S160_L001.fasta fastawrap=0
Executing jgi.ReformatReads [in=lanel-s160-index-AAGCACTG-GTGATCCANNNN-Dc-X_S160_L001_R1_trimmed.mergefilt.fastq, out=/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed_wood/woodFasta/lanel-s160-index-AAGCACTG-GTGATCCANNNN-Dc-X_S160_L001.fasta, fastawrap=0]

Input is being processed as unpaired
Input: 61717 reads 15788419 bases
Output: 61717 reads (100.00%) 15788419 bases (100.00%)

Time: 0.638 seconds.
Reads Processed: 61717 96.76k reads/sec
Bases Processed: 15788k 24.75m bases/sec
java -ea -Xmx200m -cp /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/bbmap/current/ jgi.ReformatReads in=lanel-s161-index-AAGCACTG-TTC
```

### Demultiplex leaf reads

The leaf reads are in one massive file that needs to be taken apart and organized by the sample barcodes, which are the first 12 bps of each read, after we did [some rearranging above](#).

```
In [ ]: ld="/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_demult_loa.txt"
(cat leaf_merged_filt.fasta | fastx_barcode_splitter.pl \
--bcfile leafread_fastx_map.txt \
--prefix $ld"leaf_" \
--suffix ".fa" \
--bol --mismatches 1 --partial 1 \
&>> "leaf_demult_loa.txt" &) &
```

```
In [12]: cat leaf_demult_loa.txt
```

Barcode	Count	Location
1	237434	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_1.fa
100	138715	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_100.fa
101	117532	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_101.fa
102	122279	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_102.fa
103	58121	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_103.fa
104	85418	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_104.fa
105	35436	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_105.fa
106	8	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_106.fa
107	91405	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_107.fa
108	24447	/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves/demult/leaf_108.fa

2,645,675 reads unmatched. That's a lot. Hmm.

## Clip primers

The leaves still have primers and barcodes on them:

```
In [14]: cd demult
```

In [16]: *## ITS2 was our linker primer:*

```
grep GCTGCGTTCTCATCGATGC leaf_94.fa | wc -l
arep GCTGCGTTCTCATCGATGC <(head -n 20 leaf_94.fa)
```

87084

```
CGTGATAAGACGGCTGCGTTCTCATCGATGCCAGAACCAAGAGATCCGTTAAAAGTTTAATTATTGCTTGT
GCCACTCAGAAGAGACGTCGTAAATAGAGTTGGTTCCCTCCGGGGCGCCCCGTCCCCGTGGTGGGGGCCGGC
GCCGGGAGGGGAGGCCGAGAGGGCTCCCTGCCGCCGAAGCAACGGTAGGTACGTTACAAAGGGTTATAGA
GCGGTAACTCAGTAATGATCCCTCCGCTGGTTACCAACGGAGACCTGTTACGACTTTACTCCTAAATNACC
AACCGTCTTACAGC
CGTGATCAGACGGCTGCGTTCTCATCGATGCCAGAACCAAGAGATCCGTTAAAAGTTTAATTATTGCTTGT
GCCACTCAGAAGAGACGTCGTAAATAGAGTTGGTTCCCTCCGGGGCGCCCCGTCCCCGTGGTGGGGGCCGGC
GCCGGGAGGGGAGGCCGAGAGGGCTCCCTGCCGCCGAAGCAACGGTAGGTACGTTACAAAGGGTTATAGA
GCGGTAACTCAGTAATGATCCCTCCGCTGGTTACCAACGGAGACCTGTTACGACTTTACTCCTAAATNACC
AACCGTCTGATCACG
CGTGATAAGACAGCTGCGTTCTCATCGATGCCAGAACCAAGAGATCCGTTAAAAGTTTGATTATTGCTTGT
ACCACTCAGAAGAAACGTCGTTAAATCAGAGTTGGTTATCCTCCGGGGCGCCGACCCGCCGGGGGGAGG
CGGGAGGGTACGGAGACCCCTACCGCCGAAGCAACAGTTAGGTATGGTTACAAAGGGTTAGAGCGTAAACTC
AGTAATGATCCCTCCGCTGGTTACCAACGGAGACCTGTTACGACTTTACTCCTAAATNACCAAGTGTCTTA
TCACG
CGTGATTAGACGGCTGCGTTCTCATCGATGCCAGAACCAAGAGATCCGTTAAAAGTTTGATTATTGCTTGT
GCCACTCAGAAGAGACGTCGTAAATAGAGTTGGTTCCCTCCGGGGCGCCCCGTCCCCGTGGTGGGGGCCGGC
GCCGGGAGGGGAGGCCGAGAGGGCTCCCTGCCGCCGAAGCAACGGTAGGTACGTTACAAAGGGTTATAGA
GCGGTAACTCAGTAATGATCCCTCCGCTGGTTACCAACGGAGACCTGTTACGACTTTACTCCTAAATNACC
AACCGTCTAATCAGC
CGTGATAAGACGGCTGCGTTCTCATCGATGCCAGAACCAAGAGATCCGTTAAAAGTTTGATTATTGCTTGT
GCCACTCAGAAGAAACGTCGTTAAATCAGAGTTGGTTATCCTCCGGGGACGCCCGCGTGAAGCGGGGGGGGG
CCGGGGGGCCGACCCGAAGCAACATGTAGGTATGGTTACAAAGGGTTAGAGTGGTAACTCGATAATGATCCCT
CCGCTGGTTACCAACGGAGACCTGTTACGACTTTACTCCTAAATNACCAAGCGTCTTACAGC
CGTGATAAGACGGCTGCGTTCTCATCGATGCCAGAACCAAGAGATCCGTTAAAAGTTTAATCAACTAAATGA
TATATCAGGACTTCACAAATGAATTCTTGAGTTGTATACTGGGGGCACTTAGCCGGGCTCTGGCCAGTTAA
GGCTGGGGGGCCGGGGCGCTGGTCCGAAACCTTCAATAGAAGTTATTACATTCAGTAATGATCCCTCCGCA
GGTTCACCTACGAAACCTTGTACGACTTTACTCCTAAATNACCAAGCGTCTTACAGC
CGTGATAAGACGGCTGCGTTCTCATCGATGCCGGAGCCAAGAGATCCGTTAAAAGTTTATAATTAAATT
AAGAATTTCGTTCTCCGGGCTTTCCGACAGCGGGAAAAAGTACTCTAAATGGAAAACAAGCATTCAAATT
AGGGCCGACGGTGAGCACTCATTACCAATGAGCTGACCCGGCTCCGCCCTGGAGCCGCCACCGAGGCAACGAGTC
AAAAAAACCATACATTACAAGGGGTTTGGCGATAGAGAGACGAAAAACGCTACTAATGATCCCTCCGCTGGTT
CACCAACGGAGACCTTGTACGACTTTACTCCTCTAAAGNACCAAGCGTCTTACAGC
CGTGATAAGACGGCTGCGTTCTCATCGATGCCAGAACCAAGAGATCCGTTAAAAGTTTGTGTTGAGTGGT
ACAATCAATGGTTGATTCAAGAATTATGTTGTGTTACCGTTAGGTATGAGTTGGTTATGGGGTTGGGCCATC
GACGGACGCTCTCCGGGGCGGTACGGCCCCGGCGCGCTCCGAAGGACGCCGGTCCGCCGAAGCAACATAATG
GTTTATTAGACACAGGGTGGGAGTTGGGCCCCAACGGGAACCCAGACTCGGTAATGATCCTCCGCAAGGTTAC
TACGGAAACCTTGTACGACTTTACTCCTACATNACCAAGCGTCTTACAGC
```

This forward barcode plus primer is 32 bp long:

In [17]: aa=CGTGATAAGACGGCTCGTTCTCATCGATGC

```
echo ${#aa}
```

32

The reverse primer (ITS1f) was much more degraded, not sure why. So unless we put a bunch of wildcards in our search, we don't turn it up as often. But it is still definitely present, and we can look for its reverse compliment in these merged files to confirm how much we need to clip.

```
In [18]: grep TTACTTCCTCTAAATGACCAAG leaf_94.fa | wc -l
grep TTACTTCCTCTAAATGACCAAG <(head -n 1000 leaf_94.fa)
34561
CGTATAAGACGGCTCGTTCTTCATCGATGCCAGAACCAAGAGATCCGTTGAAAGTTGATTCAATTCTTCAT
CAAACCGACGCATAAAACCGCGTGGAAAGGTCCACCGGGGCGCGGGTCTCGCTCCCCGAGGAAACAAGGGTA
TTCATACAAAAGGGTGGGAGGTGGGGCCTGGGGCCCTCACTCGGTATGATCCCTCCGCAGGTTCACCTACGGAGAC
CTTGTGACTTTTACTTCCTCTAAATGACCAAGCGTCTTATCAGC
CGTATAAGACGGCTCGTTCTTCATCGATGCCAGAACCAAGAGATCCGTTGAAAGTTAATCAATTAAATGA
TATATCAGGACTTCACAAAATGAATTCTTGAGTTTGATACTGGCGGGCACTTAGCCGGCTGGCCAGTTAA
GGCTGGGGCGCCGGCGCCTGGTCGGAACCAGGTCGACCCGCCAAGCAACATAGTGAGTAGACTTTTACTTCCT
CTAAATGACCAAGCGTCTTATCAGC
CGTATAAGTCGGCTCGTTCTTCATCGATGCTGGAGCCAAGAGATCCGTTAAAAGTTGACAGTTGCTAAG
AACACTCAGAAGTATCGTCGGGTTGAAAACAGAGATTCTGATGAGACCGGGCACCCCTCGCGGGCGCCGAA
GCAACAGGTATAATAGTTACAAAGGGTAGAGAGTATACTCATTAATGATCCCTCCGCTGGTTACCAACGGAG
ACCTTGTACGACTTTTACTTCCTCTAAATGACCAAGCGTCTTATCAGC
CGTATAAGACGGCTCGTTCTTCATCGATGCTGGAGCCAAGAGATCCGTTAAAAGTTGACAGTTGCTAAG
AACACTCAGAAGTATCGTCGGGTTGAAAACAGAGATTCTGATGAGACCGGGCACCCCTCGCGGGCGCCGAA
GCAACAGGTATAATAGTTACAAAGGGTAGAGAGTATACTCAGTAATGATCCCTCCGCTGGTTACCAACGGAG
ACCTTGTACGACTTTTACTTCCTCTAAATGACCAAGCGTCTTATCAGC
CGTATAAGACGGCTCGTTCTTCATCGATGCTGGAGCCAAGAGATCCGTTAAAAGTTGACAGTTGCTAAG
AACACTCAGAAGTATCGTCGGGTTGAAAACAGAGATTCTGATGAGACCGGGCACCCCTCGCGGGCGCCGAA
GCAACAGGTATAATAGTTACAAAGGGTAGAGAGTATACTCAGTAATGATCCCTCCGCTGGTTACCAACGGAG
ACCTTGTACGACTTTTACTTCCTCTAAATGACCAAGCGTCTTATCAGC
CGTATAAGACGGCTCGTTCTTCATCGATGCTGGAGCCAAGAGATCCGTTAAAAGTTGACAGTTGCTAAG
AACACTCAGAAGTATCGTCGGGTTGAAAACAGAGATTCTGATGAGACCGGGCACCCCTCGCGGGCGCCGAA
GCAACAGGTATAATAGTTACAAAGGGTAGAGAGTATACTCAGTAATGATCCCTCCGCTGGTTACCAACGGAG
ACCTTGTACGACTTTTACTTCCTCTAAATGACCAAGCGTCTTATCAGC
```

```
In [19]: bb=TTACTTCCTCTAAATGACCAAGCGACTTATCAGC
echo ${#bb}
34
```

So we need to clip 32 bps off of the 5' end of our reads, and 34 bps off of our 3' end. Makes sense, Barcodes (12 bp) + ITS2 (20 bp) = 32 bp, and (12 bp) + ITS1f (22 bp) = 34 bp ITS1f.

We'll use fastx again:

```
In [22]: ## leaves:
cd /home/daniel/Documents/taiwan/taiwan_combined_biom/demult

for i in *; do
    fastx_trimmer -i $i -f 33 | fastx_trimmer -t 34 -o ../leafNoPrim/${i}/leaf/
done
```

## Checking for chimeras

Let's look for and remove chimeric sequences. For the USEARCH pipeline, we'll use the [ITS1 reference files from UNITE](#) (<https://unite.ut.ee/repository.php>).

```
In [ ]: ## leaf reads:
ITS1_ref='/home/daniel/Documents/submissions/taibioinfo/UNITE/uchime_reference

cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_lea

(for i in *; do
echo $i
j=../leafNotChim/"${i/NoPrim/NotChim}"
k=${j/.fa/.log}
echo $j
echo $k
usearch -uchime_ref $i \
-db $ITS1_ref \
-nonchimeras $j \
-uchimeout $k \
-strand plus \
-notrunclabels \
&>> ../leafNotChim/leafUchime_stdout.txt
done & ) &
```

With leaf reads, 14372161 out of 14372164 reads were non-chimeric. So a loss of three reads.

```
In [ ]: ## wood reads

ITS1_ref='/home/daniel/Documents/submissions/taibioinfo/UNITE/uchime_reference

cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed

(for i in *; do
echo $i
j=../woodNotChim/"${i/\.fasta/.notChim.fasta}"
k=${j/.fasta/.log}
echo $j
echo $k
usearch -uchime_ref $i \
-db $ITS1_ref \
-nonchimeras $j \
-uchimeout $k \
-strand plus \
-notrunclabels \
&>> woodUchime_stdout.txt
done & ) &
```

With wood reads, 3,732,153 out of 3,743,135 reads were non-chimeric, so 10,982 (0.3%) reads were chimeric.

## Finding ITS1 region

Even though we trimmed the primers, it seems like to maximize the accuracy of the OTU clustering process, we should get rid of regions that are highly conserved among all fungi, i.e. the small subunit and the 5.8s subunit. Bits of both are in our reads, since our forward primer is seated in the ssu and our reverse in the 5.8s. We can estimate their locations with the ITSx tool. This is computationally a very expensive process, so we'll just look at some reads and estimate

I guess we could have done this earlier, after demultiplexing, and skipped our primer clipping step...

**Leaf ITS1 region**

```
In [ ]: ## look at leaf reads:
cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_lea
## we gotta get rid of the linebreaks, made a script
#wget https://raw.githubusercontent.com/danchurch/taiwan_combined_biom/master/
#fastq_to_fa.py

for i in *; do
echo $i
j=checkITS/${i/\_.fa/_noLB\.fa}
#echo checkITS/${i/\_.fa/_noLB\.fa}
fastq_remove_linebreaks.py $i $j
head -n 2 $j >> checkITS/allFirstReads.fa
done

## what's next? check ITS for all of these. ITxS Binaries are in the working d
.../.../ITxS_1.0.11/ITxS \
-i checkITS/allFirstReads.fa \
--preserve_T \
--allow_single_domain \
-t F \
-o checkITS/allFirstLeafReads
```

```
In [3]: cat allFirstLeafReads.positions.txt
```

HWI-M01380:62:000000000-A65GR:1:1101:18554:1494 257 bp. SSU: 1-46 ITS1: 47-227 5.8S: No end ITS2: Not found LSU: Not found Broken or partial sequence, only partial 5.8S!  
HWI-M01380:62:000000000-A65GR:1:1101:20607:1531 215 bp. SSU: 1-46 ITS1: 47-185 5.8S: No end ITS2: Not found LSU: Not found Broken or partial sequence, only partial 5.8S!  
HWI-M01380:62:000000000-A65GR:1:1101:17026:1493 238 bp. SSU: 1-46 ITS1: 47-208 5.8S: No end ITS2: Not found LSU: Not found Broken or partial sequence, only partial 5.8S!  
HWI-M01380:62:000000000-A65GR:1:1101:13354:1572 274 bp. SSU: 1-46 ITS1: 47-244 5.8S: No end ITS2: Not found LSU: Not found Broken or partial sequence, only partial 5.8S!  
HWI-M01380:62:000000000-A65GR:1:1101:11460:1564 218 bp. SSU: 1-46 ITS1: 47-188 5.8S: No end ITS2: Not found LSU: Not found Broken or partial sequence, only partial 5.8S!  
HWI-M01380:62:000000000-A65GR:1:1101:10872:1538 306 bp. SSU: 1-46 ITS1: 47-276 5.8S: No end ITS2: Not found LSU: Not found Broken or partial sequence, only partial 5.8S!  
HWI-M01380:62:000000000-A65GR:1:1101:11388:1528 220 bp. SSU: 1-46 ITS1: 47-100 5.8S: No end ITS2: Not found LSU: Not found Broken or partial sequence, only partial 5.8S!

**Wood ITS1 region**

```
In [ ]: cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed  
  
for i in *; do  
echo $i  
j=checkITS/${i/\_.fasta/_noLB\.fa}  
#echo $j  
fasta_remove_linebreaks.py $i $j  
head -n 2 $j >> checkITS/allFirstWoodReads.fa  
done  
  
../../ITSx_1.0.11/ITSx \  
-i checkITS/allFirstWoodReads.fa \  
--preserve T \  
--allow_single_domain \  
-t F \  
-o checkITS/allFirstWoodReads  
  
In [5]: cat allFirstWoodReads.positions.txt  
  
M01498:244:000000000-ANT97:1:1101:16239:1160    256 bp. SSU: 1-46      ITS1:  
47-226 5.8S: No end    ITS2: Not found LSU: Not found Broken or partial seq  
uence, only partial 5.8S!  
M01498:244:000000000-ANT97:1:1101:13588:1168    220 bp. SSU: 1-46      ITS1:  
47-190 5.8S: No end    ITS2: Not found LSU: Not found Broken or partial seq  
uence, only partial 5.8S!  
M01498:244:000000000-ANT97:1:1101:9388:1167    256 bp. SSU: 1-46      ITS1:  
47-226 5.8S: No end    ITS2: Not found LSU: Not found Broken or partial seq  
uence, only partial 5.8S!  
M01498:244:000000000-ANT97:1:1101:11226:1106    275 bp. SSU: 1-46      ITS1:  
47-245 5.8S: No end    ITS2: Not found LSU: Not found Broken or partial seq  
uence, only partial 5.8S!  
M01498:244:000000000-ANT97:1:1101:20154:1278    224 bp. SSU: 1-46      ITS1:  
47-194 5.8S: No end    ITS2: Not found LSU: Not found Broken or partial seq  
uence, only partial 5.8S!  
M01498:244:000000000-ANT97:1:1101:19378:1196    257 bp. SSU: 1-46      ITS1:  
47-227 5.8S: No end    ITS2: Not found LSU: Not found Broken or partial seq  
uence, only partial 5.8S!  
M01498:244:000000000-ANT97:1:1101:17394:1124    201 bp. SSU: 1-46      ITS1:  
47-171 5.8S: No end    ITS2: Not found LSU: Not found Broken or partial seq
```

So in both the leaves and the wood, we see that the large subunit usually ends at bp 46 of the read, and the small subunit begins at 30 bp before the end of the read. There are some exceptions, but the ITSx algorithms are computationally expensive, if we run them on our entire data set it can take days to weeks. So we'll do our best here to reduce the role of more highly conserved regions of the read (the 18s and 28s) in our OTU clustering, which is intended to capture species-ish diversity.

We do this by clipping 46 bp off of the 5' end of our reads, and 30 bp off of the 3':

```
In [ ]: ## woods
cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trimmed

for i in *fasta; do
# echo $i
echo ${i/notChim/ITSonly}
fastx_trimmer -i <(fasta_formatter -i $i) -f 47 | fastx_trimmer -t 30 -o ../w
done

## and leaves:
cd /home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/trim_leaves

for i in *fa; do
echo $i
#echo ..../leafITSonly/${i/notChim/ITSonly}
fastx_trimmer -i <(fasta_formatter -i $i) -f 47 | fastx_trimmer -t 30 -o ../l
done

## combine these into a single, big file of both leaves and reads:

leafITSonly=/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom
woodITSonly=/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom

cat $leafITSonly/* $woodITSonly/* > allReads.fasta
```

## OTU clustering

With labels simplified and reads reduced to ITS1 region. This is three steps, actually: dereplication, sorting, and clustering.

### Dereplication and Sorting of reads

We've been doing this pipeline with USEARCH, 32-bit. Not sure how this works, but this free version of USEARCH restricts RAM usage to 4 gig. If loading databases of reads takes more than this, we're punted, with a message that basically says "buy the 64 bit version, you bum."

So I went to Oslo and got a copy of [VSEARCH](#). This is an open-source, freely available parallel to USEARCH. Since I am trying to rebuild a pipeline previously with USEARCH 64-bit (not available to me now), we'll use VSEARCH sparingly, when memory limits are a problem.

```
In [ ]: ## get rid of singletons for clustering while we're at it
vsearch --derep_fulllength allReads.fasta \
--output allReads_derep.fasta \
--sizeout \
--minseqlength 1 \
--minuniquesize 2 \
&> derep_stdout.loa
```

```
In [7]: cat derep stdout.loa
```

```
vsearch v2.8.0_linux_x86_64, 11.5GB RAM, 4 cores
https://github.com/torognes/vsearch (https://github.com/torognes/vsearch)

Reading file allReads.fasta 100%
3203788852 nt in 18101955 seqs, min 2, max 373, avg 177
Dereplicating 100%
Sorting 100%
2961231 unique sequences, avg cluster 6.1, median 1, max 472437
Writing output file 100%
748827 uniques written, 2212404 clusters discarded (74.7%)
```

To sort, we go back to USEARCH...

```
In [ ]: usearch -sortbysize allReads derep.fasta -fastaout allReads sorted.fasta &> us
```

```
In [9]: cat usearch sort stdout.loa
```

```
usearch v8.1.1861_i86linux32, 4.0Gb RAM (12.1Gb total), 4 cores
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.
http://drive5.com/usearch (http://drive5.com/usearch)

License: danchurchthomas@gmail.com

00:02 224Mb 100.0% Reading allReads_derep.fasta
00:02 191Mb Getting sizes
00:03 197Mb Sorting 748827 sequences
00:06 200Mb 100.0% Writing output
```

### Cluster reads

```
In [ ]: usearch -cluster_smallmem allReads_sorted.fasta \
-id 0.95 \
-centroids otus_95_combo.fasta \
-sizein \
-sizeout \
-sortedby size \
|& tee clust stdout.loa
```

```
In [11]: cat clust stdout.log
```

```
usearch v8.1.1861_i86linux32, 4.0Gb RAM (12.1Gb total), 4 cores
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.
http://drive5.com/usearch (http://drive5.com/usearch)

License: danchurchthomas@gmail.com

00:48 59Mb 100.0% 12601 clusters, max size 681167, avg 1261.0
00:48 59Mb 100.0% Writing centroids to otus_95_combo.fasta

Seqs 748827 (748827 ())
Clusters 12601 (12601 (12601 (126)
Max size 681167 (681167 (681167 (681167 (6)
Avg size 1261.0
Min size 2
Singletons 0, 0.0% of seqs, 0.0% of clusters
Max mem 59Mb
Time 48.0s
Throughput 15.6k seqs/sec.
```

12,601 clusters (otus). ~2000 more than our last pipeline. Not sure why, though I did several things differently, including NOT seeding this pipeline with Roo's hand curated *Xylaria* stromata. We also had many more unique sequences going into this step this time around, ~750,000 seqs, compared to ~450,000 last time I did this, with the same data. Hmm....

### Assign unique names to OTU clusters

In this older version of USEARCH, with the `-cluster_smallmem` command, there was no option to give unique identifiers to the OTU clusters. So I made a script, which gives a label of your choice (here "OTU") plus a number, from the order in which the clusters were created.

```
In [ ]: ./addOTUtag.py otus_95_combo.fasta OTU otus_95_combo_relab.fasta
```

### Assign taxonomy

We'll do a mass, low-confidence assignment of taxonomy to our OTU clusters using UTAX and UNITE. These aren't to be used for any real analysis, just a quick first glance. If we become interested in an OTU, we'll need to take some time to look for a higher confidence taxonomic assignment.

```
In [ ]: ## get the suggested version of UNITE.
wget https://drive5.com/utax/data/utax_unite_v7.tar.gz
tar -xvf utax_unite_v7.tar.gz

## some useful shortcuts:
ITS1db=/home/daniel/Documents/submissions/taibioinfo/UNITE/utaxref/unite_v7/fa
ITS1tf=/home/daniel/Documents/submissions/taibioinfo/UNITE/utaxref/unite_v7/fa
OTUs=/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/OTUclu

## make a USEARCH database out of UNITE
usearch -makeudb_utax $ITS1db -taxconfsin $ITS1tf -output ITS1tax.udb |& tee

## use this and the
search -utax $OTUs -db ITS1tax.udb -fastaout OTUs_95_assTaxed.fasta -strand pl
```

## Make biom table

We can now assemble the biom table. We'll make the first generation format of biom tables, which was a json format. I prefer this to the later compiled file, hdf5-format, because it's human readable. Taxonomy assignments get thrown in for free during this step of biom table construction via uparse.

To parse correctly, we need to get rid of "-" dashes in our sample names, it doesn't parse with USEARCH. So... more sed...

```
In [ ]: allReads=/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom/OTU
sed -E '/>Dc-X/ s/(Dc)-(X)/\1_\2/g' $allReads |\
sed -E '/>Dc-PosG/ s/(Dc)-(PosG)/\1_\2/' |\
sed -E '/>Dc-PosI/ s/(Dc)-(PosI)/\1_\2/' |\
sed -E '/>Dc-Nea/ s/(Dc)-(Nea)/\1_\2/' > allReads.con.fasta
```

While we're at it, let's get rid of our unmatched reads:

```
In [ ]: sed -i '/>leafNotChim unmatched/.+1d' $allReadsCon
```

Make the table:

```
In [33]: allReadsCon=/home/daniel/Documents/submissions/taibioinfo/taiwan_combined_biom
usearch -usearch_global $allReadsCon -db OTUs_95_assTaxed.fasta -strand plus -
```

```
In [34]: cat makebiom.loa
```

```
usearch v8.1.1861_i86linux32, 4.0Gb RAM (12.1Gb total), 4 cores
(C) Copyright 2013-15 Robert C. Edgar, all rights reserved.
http://drive5.com/usearch (http://drive5.com/usearch)
```

License: danchurchthomas@gmail.com

```
00:00 44Mb 100.0% Reading ./assTax/OTUs_95_assTaxed.fasta
00:00 10Mb 100.0% Masking
00:00 11Mb 100.0% Word stats
00:00 11Mb 100.0% Alloc rows
00:00 19Mb 100.0% Build index
07:25 90Mb 100.0% Searching allReads_con.fasta, 100.4% matched
15442054 / 15456309 mapped to OTUs (99.9%)
07:25 90Mb Writing combo_otu.biom
07:25 90Mb Writing combo_otu.biom ...done.
```

Well, that's nice. But now we have to do a lot of cleaning up, and add metadata.

## Formatting Biom table and adding metadata

### Change biom taxonomy metadata format

Let's change the standard USEARCH taxonomy formatting to Greengenes database formatting. Our usearch output looks like this:

```
In [35]: arep rows combo_otu.biom -A 5
```

```
"rows": [
    {"id": "OTU19:leafNotChim_100", "metadata": {"taxonomy": "d:Fungi,p:Ascomycota(0.3128),c:Eurotiomycetes(0.2065),o:Onygenales(0.1099),f:Arthrodermataceae(0.0449),g:Arthroderma(0.0202),s:Arthroderma_melis_SH007610.07FU(0.0022)"},,
    {"id": "OTU108:leafNotChim_100", "metadata": {"taxonomy": "d:Fungi,p:Ascomycota(0.1084),c:Eurotiomycetes(0.0620),o:Pyrenulales(0.0299),f:Massariaceae(0.0138),g:Massaria(0.0060)"},,
    {"id": "OTU1:leafNotChim_100", "metadata": {"taxonomy": "d:Fungi,p:Ascomycota(0.9897),c:Sordariomycetes(0.7674),o:Hypocreales(0.5816),f:Hypocreales_fam_Incertae_sedis(0.3979),g:Myrothecium(0.2279)"},,
    {"id": "OTU202:leafNotChim_100", "metadata": {"taxonomy": "d:Fungi,p:Ascomycota(0.4325),c:Dothideomycetes(0.2853),o:Dothideomycetidae_ord_Incertae_sedis(0.1601),f:Strigulaceae(0.0654),g:Strigula(0.0309),s:Strigula_smargdula_SH211054.07FU(0.0045)"},,
    {"id": "OTU426:leafNotChim_100", "metadata": {"taxonomy": "d:Fungi,p:Ascomycota(0.2072),c:Dothideomycetes(0.1317)"},,
```

Let's edit this with SED:

```
In [38]: sed '/taxonomy/ s/([0-1]\.[0-9]*)//g' combo_otu.biom | \
sed -E 's/("taxonomy")(:)/\1:[/ | \
sed -E 's/"}}/,]}}/' | \
sed -E '/taxonomy/ s/(d:)([^,]*)/"k_\2"/' | \
sed -E '/taxonomy/ s/(p:)([^,]*)/"p_\2"/' | \
sed -E '/taxonomy/ s/(c:)([^,]*)/"c_\2"/' | \
sed -E '/taxonomy/ s/(o:)([^,]*)/"o_\2"/' | \
sed -E '/taxonomy/ s/(f:)([^,]*)/"f_\2"/' | \
sed -E '/taxonomy/ s/(g:)([^,]*)/"g_\2"/' | \
sed -E '/taxonomy/ s/(s:)([^,]*)/"s_\2"/' | \
sed -E '/taxonomv/ s/.1}\}/1}\}/' > combo_otu_relab.biom
```

```
In [39]: arep rows combo_otu_relab.biom -A 5
```

```
"rows": [
    {"id": "OTU19:leafNotChim_100", "metadata": {"taxonomy": ["k_Fungi","p_Ascomycota","c_Eurotiomycetes","o_Onygenales","f_Arthrodermataceae","g_Arthroderma","s_Arthroderma_melis_SH007610.07FU"]}},,
    {"id": "OTU108:leafNotChim_100", "metadata": {"taxonomy": ["k_Fungi","p_Ascomycota","c_Eurotiomycetes","o_Pyrenulales","f_Massariaceae","g_Massaria"]}},,
    {"id": "OTU1:leafNotChim_100", "metadata": {"taxonomy": ["k_Fungi","p_Ascomycota","c_Sordariomycetes","o_Hypocreales","f_Hypocreales_fam_Incertae_sedis","g_Myothecium"]}},,
    {"id": "OTU202:leafNotChim_100", "metadata": {"taxonomy": ["k_Fungi","p_Ascomycota","c_Dothideomycetes","o_Dothideomycetidae_ord_Incertae_sedis","f_Strigulaceae","g_Strigula","s_Strigula_smargdula_SH211054.07FU"]}},,
    {"id": "OTU426:leafNotChim_100", "metadata": {"taxonomy": ["k_Fungi","p_Ascomycota","c_Dothideomycetes"]}},,
```

Looks good. But the leafNotChim thing is annoying, especially with our sample ids. So change this:

```
In [40]: sed -E -i 's/(leaf)(NotChim )([0-9]*)/\3\1/a' combo_otu_relab.biom
```

```
In [41]: arep rows -A 10 combo otu relab.biom
```

```
"rows": [
    {"id": "OTU19:100leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota", "c_Eurotiomycetes", "o_Onygenales", "f_Arthrodermataceae", "g_Arthroderma", "s_Arthroderma_melis_SH007610.07FU"]}},
    {"id": "OTU108:100leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota", "c_Eurotiomycetes", "o_Pyrenulales", "f_Massariaceae", "g_Massaria"]}},
    {"id": "OTU1:100leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota", "c_Sordariomycetes", "o_Hypocreales", "f_Hypocreales_fam_Incertae_sedis", "g_Myrothecium"]}},
    {"id": "OTU202:100leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota", "c_Dothideomycetes", "o_Dothideomycetidae_ord_Incertae_sedis", "f_Strigulaceae", "g_Strigula", "s_Strigula_smaragdula_SH211054.07FU"]}},
    {"id": "OTU426:100leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota", "c_Dothideomycetes"]}},
    {"id": "OTU1905:100leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota"]}},
    {"id": "OTU10:100leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota", "c_Lecanoromycetes"]}},
    {"id": "OTU10433:17leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota", "c_Sordariomycetes", "o_Hypocreales", "f_Hypocreales_fam_Incertae_sedis", "g_Myrothecium"]}},
    {"id": "OTU17:100leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota", "c_Dothideomycetes", "o_Pleosporales"]}},
    {"id": "OTU429:100leaf", "metadata": {"taxonomy": ["k_Fungi", "p_Ascomycota", "c_Sordariomycetes", "o_Hypocreales", "f_Hypocreales_fam_Incertae_sedis", "g_Myrothecium"]}},
```

```
In [42]: arep columns -A 10 combo otu relab.biom
```

```
"columns": [
    {"id": "100leaf", "metadata": null},
    {"id": "101leaf", "metadata": null},
    {"id": "102leaf", "metadata": null},
    {"id": "103leaf", "metadata": null},
    {"id": "104leaf", "metadata": null},
    {"id": "105leaf", "metadata": null},
    {"id": "106leaf", "metadata": null},
    {"id": "107leaf", "metadata": null},
    {"id": "108leaf", "metadata": null},
    {"id": "109leaf", "metadata": null},
```

Looks okay. Check to see if the biom package can find problems:

```
In [43]: biom validate-table -i combo otu relab.biom
```

```
Invalid format 'Biological Observation Matrix 1.0', must be '1.0.0'
'id' in {'metadata': {'taxonomy': ['k_Fungi', 'p_Ascomycota', 'c_Sordariomycetes', 'o_Ophiostomatales', 'f_Ophiostomataceae', 'g_Pesotum']}, 'id': ''}
} appears empty
Bad value at idx 0: [0, 0, 81511]
Timestamp does not appear to be ISO 8601
The input file is not a valid BIOM-formatted file.
```

Don't know why this name was left off, but this happened before. Find the OTU and fill in the name manually:

```
In [44]: arep "a:Pesotum" combo otu.biom -A 1 -B 1

        {"id":"OTU10613:leafNotChim_111", "metadata":{"taxonomy":"d:Fungi"},},
        {"id":"","metadata":{"taxonomy":"d:Fungi,p:Ascomycota(0.3128),c:Sordariomycetes(0.2065),o:Ophiostomatales(0.1099),f:Ophiostomataceae(0.0449),g:Pesotum(0.0202)"},},
        {"id":"OTU7760:leafNotChim_111", "metadata":{"taxonomy":"d:Fungi,p:Ascomycota(0.2072),c:Schizosaccharomycetes(0.1317),o:Schizosaccharomycetales(0.0620),f:Schizosaccharomycetaceae(0.0273),g:Schizosaccharomyces(0.0108)"},}
```

Let's find this in our OTU clusters that have had taxonomy assignments:

```
In [45]: arep "a:Pesotum(0.0202)" OTUs_95_assTaxed.fasta

>OTU6797:leafNotChim_111;size=16;tax=d:Fungi,p:Ascomycota(0.3128),c:Sordariomycetes(0.2065),o:Ophiostomatales(0.1099),f:Ophiostomataceae(0.0449),g:Pesotum(0.0202);
```

Fill this into our OTU table with sed:

```
In [46]: sed '/"id":""/ s/"id":""./"id":"OTU6797:111leaf"./' combo otu relab.biom -i
```

```
In [47]: biom validate-table -i combo otu relab.biom

Invalid format 'Biological Observation Matrix 1.0', must be '1.0.0'
Bad value at idx 0: [0, 0, 81511]
Timestamp does not appear to be ISO 8601
The input file is not a valid BIOM-formatted file.
```

Beh. Minor stuff. Onward...

## Add sample metadata

We have a spreadsheet with our sample metadata on it. We can use this to assign useful info about each of our samples, with the `biom add-metadata` command:

```
In [17]: head meta 2018.06.14.tsv
```

#SampleID	Library	SorC	X	Y	Forest_Type	Host_family	Host_genus	Host_species	stream_distan
ce	vegcom								
Dc_X	W	Control	NA	NA	NA	NA	NA	NA	NA
	NA	NA							
Dc_PosG	W	Control	NA	NA	NA	NA	NA	NA	NA
	NA	NA							
Dc_PosI	W	Control	NA	NA	NA	NA	NA	NA	NA
	NA	NA							
Dc_Neg	W	Control	NA	NA	NA	NA	NA	NA	NA
	NA	NA							
1w	W	Sample	360	220	7	Juglandaceae	Engelhardtia		
	roxburghiana		Engelhardtia_roxburghiana			24.11897		2	
2w	W	Sample	360	221	7	Theaceae		Pyrenaria	
	shinkoensis		Pyrenaria_shinkoensis			23.22664		2	
3w	W	Sample	361	221	7	Proteaceae		Helicia formo	
sana	Helicia_formosana			22.77525					
4w	W	Sample	361	220	7	Theaceae		Pyrenaria	
	shinkoensis		Pyrenaria_shinkoensis			23.66758		2	
5w	W	Sample	363	220	7	Theaceae		Pyrenaria	
	shinkoensis		Pyrenaria_shinkoensis			22.7648	2		

```
In [1]: biom add-metadata -i combo_otu_relab.biom -o combo_otu_wMeta.biom --sample-met
```

```
In [2]: ## just checking:  
biom validate-table -i combo_otu_wMeta.biom
```

The input file is a valid BIOM-formatted file.

This command adds metadata to each of our sites, but it mashes our biom file into a single line, making it really hard to read.

```
In [3]: head -c 1000 combo otu wMeta.biom; tail -c 1000 combo otu wMeta.biom
```

```
{"id": "None", "format": "Biological Observation Matrix 1.0.0", "format_url": "http://biom-format.org", "matrix_type": "sparse", "generated_by": "BIOM-Format 2.1.6", "date": "2018-06-17T13:56:51.200393", "type": "OTU table", "matrix_element_type": "float", "shape": [11588, 232], "data": [[0, 0, 81511.0], [0, 1, 178.0], [0, 3, 1145.0], [0, 4, 3.0], [0, 27, 1.0], [0, 35, 226.0], [0, 37, 939.0], [0, 51, 1.0], [0, 54, 2.0], [0, 93, 2.0], [0, 114, 57.0], [0, 122, 1.0], [0, 125, 43.0], [0, 127, 13282.0], [0, 128, 2.0], [0, 129, 1.0], [1, 0, 26184.0], [1, 3, 7.0], [1, 4, 3504.0], [1, 33, 1.0], [1, 128, 2.0], [2, 0, 2735.0], [2, 1, 6795.0], [2, 2, 586.0], [2, 3, 4290.0], [2, 4, 2898.0], [2, 5, 5424.0], [2, 6, 1.0], [2, 7, 3.0], [2, 8, 859.0], [2, 9, 504.0], [2, 10, 435.0], [2, 11, 1191.0], [2, 12, 2.0], [2, 13, 259.0], [2, 14, 9.0], [2, 15, 7.0], [2, 16, 530.0], [2, 17, 25.0], [2, 18, 6454.0], [2, 19, 8038.0], [2, 20, 2138.0], [2, 21, 6263.0], [2, 22, 6891.0], [2, 23, 3186.0], [2, 24, 138.0], [2, 25, 6137.0], [2, 26, 7641.0], [2, 27, 1.0], [2, 28, 65417.0], [2, 29, 3333.0], [2, 30, 848.0], [2, 31, 886.0], [2, 32, 5517.0], [2, 33, 6203.0], [2, 34, 484.0], [2, 35, 742.071]}, {"id": "133w", "metadata": {"vegcom": "3", "stream_distance": "22.16354", "Host_genus": "Helicia", "Host_genus_species": "Helicia_formosana", "Library": "W", "Forest_Type": "7", "Host_species": "formosana", "X": "47", "Host_family": "Proteaceae", "SorC": "Sample", "Y": "47"}}, {"id": "Neg", "metadata": {"vegcom": "NA", "stream_distance": "NA", "Host_genus": "NA", "Host_genus_species": "NA", "Library": "W", "Forest_Type": "NA", "Host_species": "NA", "X": "NA", "Host_family": "NA", "SorC": "Control", "Y": "NA"}}, {"id": "PosG", "metadata": {"vegcom": "NA", "stream_distance": "NA", "Host_genus": "NA", "Host_genus_species": "NA", "Library": "W", "Forest_Type": "NA", "Host_species": "NA", "X": "NA", "Host_family": "NA", "SorC": "Control", "Y": "NA"}}, {"id": "PosI", "metadata": {"vegcom": "NA", "stream_distance": "NA", "Host_genus": "NA", "Host_genus_species": "NA", "Library": "W", "Forest_Type": "NA", "Host_species": "NA", "X": "NA", "Host_family": "NA", "SorC": "Control", "Y": "NA"}]}
```

Let's use [js-beautify \(<https://www.npmjs.com/package/js-beautify>\)](https://www.npmjs.com/package/js-beautify) to re-render into a multi-line json:

```
In [4]: is-beautifyv combo otu wMeta.biom > combo otu wMeta prettv.biom
```

```
In [5]: biom validate-table -i combo otu wMeta prettv.biom
```

The input file is a valid BIOM-formatted file.

```
In [6]: head -n 20 combo otu wMeta prettv.biom
```

```
{
  "id": "None",
  "format": "Biological Observation Matrix 1.0.0",
  "format_url": "http://biom-format.org",
  "matrix_type": "sparse",
  "generated_by": "BIOM-Format 2.1.6",
  "date": "2018-06-17T13:56:51.200393",
  "type": "OTU table",
  "matrix_element_type": "float",
  "shape": [11588, 232],
  "data": [
    [0, 0, 81511.0],
    [0, 1, 178.0],
    [0, 3, 1145.0],
    [0, 4, 3.0],
    [0, 27, 1.0],
    [0, 35, 226.0],
    [0, 37, 939.0],
    [0, 51, 1.0],
    [0, 54, 2.0]
```

Our sample metadata is here:

```
In [7]: arep columns combo_otu_wMeta_pretty.biom -A 20
```

```
"columns": [{"  
    "id": "100leaf",  
    "metadata": {  
        "vegcom": "2",  
        "stream_distance": "25.97654",  
        "Host_genus": "Helicia",  
        "Host_genus_species": "Helicia_formosana",  
        "Library": "L",  
        "Forest_Type": "7",  
        "Host_species": "formosana",  
        "X": "183",  
        "Host_family": "Proteaceae",  
        "SorC": "Sample",  
        "Y": "20"  
    }  
, {  
    "id": "101leaf",  
    "metadata": {  
        "vegcom": "3",  
        "stream_distance": "18.36984"  
    }  
}, {  
    "id": "102leaf",  
    "metadata": {  
        "vegcom": "3",  
        "stream_distance": "21.3725"  
    }  
}, {  
    "id": "103leaf",  
    "metadata": {  
        "vegcom": "3",  
        "stream_distance": "11.08831"  
    }  
}, {  
    "id": "104leaf",  
    "metadata": {  
        "vegcom": "3",  
        "stream_distance": "1.409998"  
    }  
}, {  
    "id": "105leaf",  
    "metadata": {  
        "vegcom": "3",  
        "stream_distance": "22.46722"  
    }  
}, {  
    "id": "106leaf",  
    "metadata": {  
        "vegcom": "2",  
        "stream_distance": "82.49734"  
    }  
}, {  
    "id": "107leaf",  
    "metadata": {  
        "vegcom": "1",  
        "stream_distance": "64.85876"  
    }  
}, {  
    "id": "108leaf",  
    "metadata": {  
        "vegcom": "1",  
        "stream_distance": "19.02113"  
    }  
}, {  
    "id": "109leaf",  
    "metadata": {  
        "vegcom": "3",  
        "stream_distance": "13.46815"  
    }  
}]
```

So we should be good - biom table is constructed, clean of major errors, with taxonomic and sample metadata attached.

We will do most of our downstream manipulations of this biom table with [phyloseq \(https://joey711.github.io/phyloseq\)](https://joey711.github.io/phyloseq/), a package made for handling microbial community data in R.

Let's check to see if phyloseq likes our biom table:

```
In [1]: library("phyloseq")  
  
biom95_meta <- import_biom("combo_otu_wMeta.biom", parseFunction=parse_taxonom  
  
In [3]: sample_data(biom95_meta)
```

	vegcom	stream_distance	Host_genus	Host_genus_species	Library	Forest_Type	Host_
100leaf	2	25.97654	Helicia	Helicia_formosana	L	7	for
101leaf	3	18.36984	Helicia	Helicia_formosana	L	7	for
102leaf	3	21.3725	Cleyera	Cleyera_japonica	L	7	.
103leaf	3	11.08831	Helicia	Helicia_formosana	L	7	for
104leaf	3	1.409998	Helicia	Helicia_formosana	L	7	for
105leaf	3	22.46722	Limlia	Limlia_uriana	L	7	
106leaf	2	82.49734	Helicia	Helicia_formosana	L	3	for
107leaf	1	64.85876	Blastus	Blastus_cochinchinensis	L	3	cochinc
108leaf	1	19.02113	Cleyera	Cleyera_japonica	L	3	.
109leaf	3	13.46815	Meliosma	Meliosma_squamulata	L	7	squ

```
In [4]: otu_table(biom95 meta)
```

	100leaf	101leaf	102leaf	103leaf	104leaf	105leaf	106leaf	107leaf	108leaf	109leaf	...	124v
<b>OTU19:100leaf</b>	81511	178	0	1145	3	0	0	0	0	0	0	0
<b>OTU108:100leaf</b>	26184	0	0	7	3504	0	0	0	0	0	0	0
<b>OTU1:100leaf</b>	2735	6795	586	4290	2898	5424	1	3	859	504	...	0
<b>OTU202:100leaf</b>	3214	0	0	6330	1	0	0	0	0	0	0	0
<b>OTU426:100leaf</b>	5943	0	0	0	38	0	0	0	0	0	0	0
<b>OTU1905:100leaf</b>	98	0	71	35	0	0	0	0	0	0	0	0
<b>OTU10:100leaf</b>	6618	22297	36	6276	12852	6258	1	0	1	81	...	0
<b>OTU10433:17leaf</b>	1	0	0	1	0	0	0	0	0	0	0	0
<b>OTU17:100leaf</b>	2877	2600	6	3782	5298	491	0	0	51	2	...	0
<b>OTU429:100leaf</b>	559	0	0	0	0	0	0	0	0	0	0	0

Seems okay. We'll work on the revised stats pipeline in a separate notebook.