Spatial niche project – Summary

# Principal objective

Quantify to what extent tropical trees niche-differentiate by specializing to different soil resources.

# Subgoals

1. Infer and enumerate the “soil niches” in a forest.
2. Assess whether our inferred soil niches correlate with measured soil indicators (nutrient concentrations). If so, this lends credence to our inference.
3. Assess whether our inferred soil niches correlate with species traits. If so, this provides a link between species traits and the types of soil resources to which they specialize.
4. Obtain an affinity matrix to be compared with D’Andrea, Gibbs & O’Dwyer 2020 to make inferences about whether non-neutral behavior should be observed at the community level.

# Methods and Results

**Inferring and enumerating the soil niches**

1. First, we considered every pair of trees within the 50ha forest dynamics plot (FDP) on BCI and determined whether species A and B tended to occur in close proximity (i.e. d\_{tree1, tree2} < d\_cutoff) more than expected by chance. We used a distance cutoff of 20-30 m, as this is the estimated range of physical interaction between neighboring trees (cite literature). We also restricted the analysis to species above an abundance cutoff, because we lack statistical power to determine whether rare species are physically closer than chance.
2. We treat the FDP species as nodes in a network, with links between species that were found to be in proximity beyond chance.
3. We then use R package *igraph* to find “communities”, i.e. subsets of nodes that are unusually interconnected and relatively isolated from all other nodes in the network. This “community-finding” algorithm uses random walks to find these subsets (i.e. the notion that if you’re moving around the network randomly, you will tend to spend most of your time within the same community).
4. The community-finding algorithm reveals 3 such species groups (Fig. 1). We identify these groups as being the “soil niches” (“niches”), i.e. the three distinctive strategies regarding soil resources. Upon finishing this part of the analysis, each species has a niches label.
5. We can perform network-based analyses to determine how tight the classification is, and whether it is statistically significant. For example, I found that the *modularity* of this three-niches graph is significant by randomizing the species labels of the FDP trees, finding communities among the resulting graphs, and calculating the modularity of the results.
6. John checked that this result is also robust to using different cutoffs for the interaction distance and minimum abundance. Using distance cutoff from 2 to 40 meters, abundance threshold from 0 to 400, we consistently find three clusters, with some exceptions of 2 and 4 clusters. Upon visual inspection, the predicted soil landscape looks similar over a wide range of distance (10-30 m) and minimum abundance. We ultimately use the parameters: distance = 10m, abundance cutoff = 50 for the following analysis.

**Assessing whether our niches correlate with nutrient concentrations in the FDP**

1. Each of the 3 niches corresponds to a bias for preferentially recruiting in sites whose local soil conditions match the niches. In other words, we infer from the finding of three niches that the FDP has three ecologically relevant “soil types”, which are presumably a composite function of all the conditions that matter for plant recruitment (nutrients, water, pH, toxicity).
2. Given the coordinates of each tree in the FDP and the niches labels resulting from our graph analysis, we can infer the soil type under each tree via *kernel density estimation:* we suppose continuity/autocorrelation in the spatial distribution of soil types, and in particular we assume that if a point (x0, y0) has soil type 1, then points (x, y) separated by distance d have probability P\_1(d) of belonging to soil type 1 and 1-P of belonging to one of the other soil types. We let P be a Gaussian with width d. The bandwidth here acting as a smoothing parameter. Doing this for every data point returns a map of probabilities for the entire FDP. (John performed this analysis).
3. At this point we have options for what to do next. The simplest is to assign each pixel the soil type with the highest probability. A more complicated alternative is to instead work with the probabilities. I chose the simple route (Fig 2).
4. We have publicly available data (from R. John, Dalling, et al PNAS 2007) on the concentrations of several nutrients on BCI (namely P, B, K, Mg, Mn, Cu, Ca, pH, organic N, mineralized N, and Al) at the scale of 20m-by-20m quadrats spanning the whole FDP (1,250 such quadrats). We can then compare the raster of soil types with the raster of nutrient concentrations, looking for a correspondence. We expect such correspondence if our inferred soil types relate to biologically meaningful features of the environment.
5. One problem we run into with this question is that both the soiltype raster and the nutrient raster are autocorrelated (in part because autocorrelation is assumed in the productions of both rasters, and then used to extrapolate from discrete datapoints). The problem is that two autocorrelated rasters may match better than chance (here meaning expected matches between the cells of permutated versions of said rasters) simply because both rasters tend to be blobby: it is easier to match blobs than to match small cells. One solution to this problem is to compare the match not against fully randomized rasters, but against *partially randomized* rasters, where we permutate cells within a certain distance scale but keep the overall shape of the blobs (see Fig. 3). We then compare the match level against null models randomized at increasingly large distances.
6. Another thing to consider is that most of the nutrient concentrations are highly correlated: basically, the right side of the FDP is high in nutrients except P and organic N, the left side is low in nutrients and high in Al (which is toxic to plants), and the middle has the most P and organic N (Fig. 4). This suggests that we can simplify the analysis by running a *PCA* on it. Indeed, the 1st and 2nd principal components account for 93% of the variance in nutrient concentration. This is also obvious from plots of the cells on the PCA plane colored by nutrient concentration. The high and low nutrient areas are well separated on the PCA plane – as are the soil types! (Figs. 5, 6) This suggests very strongly that our inferred soil types are meaningfully connected to soil nutrients. Specifically, it is clear that soil type 1 (red) corresponds to alkaline soils with high N and P, soil type 2 (green) corresponds to low nutrients and high Al, and soil type 3 (blue) corresponds to high nutrients.
7. One interesting way to quantify this association is to train a machine-learning classifier on some of the nutrient data and use it to predict the soil types of unseen data. I used the *C5.0 decision tree / rule learner* algorithm to do this, and results are very good. Using repeated 10-fold cross-validation, I find that the soil nutrients predict the soil type with high accuracy (> 85%). To measure the quality of the prediction, I used *Cohen’s Kappa*, which is a measure of how the observed accuracy exceeds predicted accuracy (i.e. the proportion of correct guesses achieved by guessing at random). This index varies from 0 to 1, with values between 0.6-0.8 considered good agreement and > 0.8 considered very good agreement. For our data, I am finding Kappa ~ 0.83.
8. One could argue that our good agreement between nutrient concentrations and inferred soiltypes comes just from the fact that both the nutrient and soiltype rasters are spatially autocorrelated (i.e. blobby). The idea is that, since both rasters are blobby, substantial regions of similar nutrient concentrations will coincidentally overlap with a single soiltype, thus allowing a classifier to successfully but artifactually predict the latter from the former. To deal with this possibility, I performed two tests: A) Partially randomize the nutrient data in a way that maintains the autocorrelation at a certain scale but randomized the data within those scales. B) Train the classifier to predict randomly generated but similarly autocorrelated soiltypes.
9. When we train the classifier using partially randomized nutrient data at different scales, we see increasingly poor predictions (Fig. 7) I think the fact that the better-than-chance agreement doesn’t just collapse to zero indicates that spatial autocorrelation in both the nutrient and soiltype data plays a role. However, the fact that we see a large drop in predictive power when randomizing at scales smaller than the blob sizes indicates that our results are not solely due to this autocorrelation.
10. In addition, when I ran the C5.0 algorithm on null data where the soiltype is a discretized Gaussian random field with autocorrelation scale and other variogram parameters similar to the actual inferred soiltype raster (Fig. 8), results clearly show that a) The autocorrelated random data results in a positive Kappa (blue box), confirming that autocorrelated data will indeed result in artifactual correspondence/predictability between nutrients and soiltypes. b) However, the Kappa is substantially lower than what we get from our actual inferred soiltype raster, showing that our good results transcend this artifactual correspondence due to autocorrelation.
11. Overall, these results are at once reassuring and exciting. They’re exciting because we were able to predict high- and low-nutrient areas in the FDP indirectly via patterns of spatial proximity between species. It is reassuring because from the outset the project is founded on the idea that these patterns are created precisely because of soil heterogeneity, so it’s good that we are confirming that assumption post-facto.

**Assessing whether our soil types are associated with species traits**

1. The results above are cool but expected. In fact if we didn’t find a correspondence, we should be concerned. However, it would be very exciting to find that our soil types predict species traits or vice versa, because then we’d be connecting traits to niches – which I’m all about!
2. We have trait data that Joe Wright at STRI shared with me for the BCI trait clustering project. First, I trim the data down to five trait categories: *vital rates* (dbh growth, mortality), *size traits* (dbh, height), *leaf traits* (all sorts of leaf density), *seed traits* (fruit and seed size), and *wood traits* (dry wood density). Then, I run a separate PCA on each of these categories, keeping the 1st principal component. This leaves us with 5 trait-based features on which to train the AI to predict the niches of the species. (Note: unlike the nutrient data, the trait data has *a lot* of NAs. We’re forced to use value imputation methods to deal with those NAs, otherwise we’d lose a lot of the 77 species in our analysis). I perform this category-specific PCA analysis for several reasons:
   1. We argue in our Translucent Windows paper that measured traits are imperfect proxies to the actual niche axes, and collating information from several related by different measures could yield better results as it could reduce the measurement noise. So instead of using 17 different leaf traits as predictors of the niches, we collapse all these measures into a single axis of leaf longevity/sturdiness.
   2. Traits of the same type are strongly correlated (Fig. 9).
   3. The first principal components of some trait types are also correlated (Fig. 10), albeit much more weakly.
   4. Correspondingly, reducing the covariate tally from over 20 to the 5 first components for each trait category greatly improves the AI predictions from Kappa = 0.2 to Kappa = 0.5.
3. Results here are weaker than with nutrients, but still definitely significant. The C5.0 rule learner returns a Kappa of about 0.5, which is considered moderate agreement. With randomized data you get Kappa = 0 (Fig. 11). One advantage of the C5.0 algorithm over neuro nets (which are more powerful and often more accurate) is interpretability of results: we get a set of rules such as *If mortality is below X and leaf density is above Y, then classify the species as niches 1*. (See Rules page after figures) Indeed, the results I’m getting suggest that niches 1 and 2 differ mostly by vital rates, in that niches 1 is the slow-and-steady strategy while niches 2 is the live-fast-die-young strategy. Niches 3 is a bit more subtle, with similar vital rates as niches 1 but lower leaf density and higher seed size. (Fig. 12)
4. One big achievement of this analysis would be to connect it to the nutrient analysis in a way that makes sense. So it would be great if we found that the slow-and-steady plants tend to occur in low-nutrient soils and vice-versa. However, we are finding the opposite! It looks like the slow-and-steady group occurs in the high-N high-P area, while the live-fast group occurs in low-nutrient areas, and group 3 occurs in the high-nutrient area. I can’t say I understand this. Would be interesting to hear Jim Dalling’s or Joe Wright’s opinion.
5. We also performed other ML methods. Random forest takes the average of an ensemble of decision trees. We trained Random Forest and got Kappa = 0.475 with max\_depth = 1, max\_features = 2 and min\_samples\_leaf = 5. We also tried gradient boosting which is a sequential method that takes an ensemble of decision trees to minimize the loss function. It doesn’t perform better than the random forest or CS5.0. The reason here is that our trait data set is so small that sophisticated classification method such as gradient boosting will overfit. All these confirms that decision tree method gives us both intuition and accuracy.

**The affinity matrix – putting our results in the context of D’Andrea et al PCB 2020**

1. Assuming the above results indicate that we are finding meaningful niches on BCI, and treating each putative soil type as a resource, we can then put together a 77 x 3 affinity matrix Cij quantifying the degree of preference of each species for each soil type.
2. I estimate Cij as the proportion of trees of species i found in sites with soil type j, relative to the total abundance of species i. This gives the matrix plotted in Fig. 13
3. In D’Andrea et al PCB 2020 we considered two consumer-resource scenarios: *specialists*, where each species has its preferred resource which it uses with proportion Cd (d for diagonal), and all non-preferred resources are used with proportion Co; and *generalists*, where the Cij is a uniform random matrix. In the first case, the degree of niche differentiation was measured as Cd / Co, and in the second, as the CV of the matrix.
4. Assuming the specialist case is more relevant here, I estimated Cd and Co by averaging over either same-niche or across-niche entries of the Cij (i.e. entries where the species is found in its preferred soil types vs non-preferred soil types). I get Cd = 0.5 and Co = 0.25, giving a ratio of 2 which is much smaller than the minimum required for non-neutral behavior to emerge at the community level. (Fig. 2E in that paper). I will note that we looked at 50 species and mean abundance = 100 trees/species in the paper, whereas here we have 77 species and mean abundance 230, which I think are comparable situations. If we believe both this estimation and the results in our PCB paper, we’d conclude that the PCB paper would predict neutral-like behavior on BCI based on our estimated Cd and Co, which is indeed the case.

# Next steps

I think the results reported above make for a publishable paper. I would be very interested to hear James’s feedback, and eventually once we’re more confident about the actuality of these results, run them by Joe Wright and Jim Dalling.

Re novelty of our paper, I think the finding of exactly 3 soil-based niches is new, and so is the association to traits. Dalling has looked at whether edaphic measures are correlated with spatial distribution of trees on BCI, but as far as I can tell not to this degree of detail of associating specific species to specific soil conditions. Methodologically, I don’t recall seeing papers that use graph-based methods or machine learning to find and describe niches.

Two things that can help us sell the paper are:

1. Tighten the story connecting traits to nutrients. Why are we finding that fast growers occur in depauperate areas and vice-versa, rather than the reverse?
2. Try venturing out to other forests. John et al 2007 discussed edaphic data not only on BCI but also other neotropical forests. If we can find trait data for those, and ended up finding consistent results, I think this could be a PNAS paper.

June 17, 2021 - James O’Dwyer feedback

1. I know you have gone way beyond them, but I want to compare still our results to that Volkov 2009 paper. Did they see something that would at least be consistent with three clusters at bci? It’s a bit hard to make a connection clearly, because they focus on the 20 most abundant species. And I remember we thought that their inferred interaction matric would be quite similar to CC^T for us, but they never thought about it in terms of a matrix C of preferences, so we don’t see any dimensionality of that matrix.

I guess you could say they were aiming for something similar to what we are, but they didn’t make these next steps that we are looking at here: connecting species clusters to soil; sniche to traits; and then assuming that C underlies dynamics rather than Lotka-Volterra.

R: I agree with what you said: we go beyond what they did in that we assume an underlying relationship to soil resources – rather than phenomenologically assigning coupling constants to be found by solving the master equation for equilibrium, and we scrutinize CC^T further than they do as we look for groups of species rather than just comparing the diagonal to off-diagonal elements.

Re can we say their results are consistent with 3 clusters at bci? I think we could try to download their CC^T matrix and use the same algorithm we used to find “communities”, i.e. subsets of species that are highly connected in space. According to our methodology, both their interaction matrix and their covariance matrix should show this 3-group structure.

Re 20 species, I forget now but I think either John or I tested our results for robustness against different abundance cutoffs. John, do you remember anything about this?

2. The connection from soil nutrients to clusters to traits is very cool and quite fascinating.

That seems like something that we must be able to make sense of and could be a compelling part of the paper. I agree that when we are ready feedback from e.g. Jim would be very useful in interpreting this.

Maybe we need to think some more on the causality here. Could it be that the slow life history species would tend to generate higher N, P in the soil over long periods of time?

This also got me wondering about fluctuations over time in species composition. My work with Ryan and others shows that there are large fluctuations in species abundances over the censuses…but I wonder whether they take place largely within the preferred soil types. I guess the soil types themselves would change relatively slowly.

R: You offer an interesting idea, namely that the causal arrow between soil conditions and plant occurrence can go both ways – with plants influencing the local soil nutrient concentrations in addition to responding to them. Considering this possibility leads to the opposite predictions we’ve been making: species that are not limited by soil resources (and therefore would have traits associated with slow growth) may actually release these resources as exudates, so one would expect the soil under them to be enriched. By contrast, species that are hungry for resources (i.e. fast-growth plants) will lap them up from the soil until resulting concentrations limit further growth, and therefore can be expected to be found in nutrient-poor soils.

Re temporal fluctuations in species composition, that is a great idea for a follow-up study. Our study predicts that most of the temporal turnover should occur within species grouped in the same sniche. Do we have enough data to test this?

3. I don’t think it is critical to this story, but I am interested…would the Cij you have inferred tend to be consistent with neutral-like SADs etc? You said in the draft something like “and indeed this [exhibiting neutral behavior] is the case”. but I wasn’t sure if that was a statement coming from just the general sense that BCI has been somewhat well-explained by neutral models in the past, or a more specific test.

I know the temporal census-to-census dynamics won’t really look neutral (see the work with Tak and Ryan on environmental stochasticity). I mean, it won’t look “niche-like” either. But it won’t follow the patterns in our PCB paper. The SAD has been proposed to be consistent with neutrality of course, but there are subtleties there, as the local SAD relies on immigration from outside, and as you know from my paper with Stephen C this is a bit murky if you try to fit the neutral mode with large-scale data.

So…just wondering (and maybe it is not entirely necessary for the main story here) how we close the loop with the connection to our PCB paper. Our inferred C\_ij would we think put us in the “neutral-like” regime, all else being equal…but what do we look at in the data to highlight that outcome?

R: I see your point. I wasn’t thinking that deeply about it. I just figured neutral theory’s famous fit to SADs (in the less careful, spatially agnostic sense of the test) is widely accepted as indication that the forest can behave neutrally even if it is *not* neutral. I suppose the further we can go is offer the same type of neutrality test using BCI data as the tests we ran in our PCB paper – namely, fitting a logseries. In our PCB paper, we showed that communities assembled based on Cij matrices with niche index as low as the one we get in the present study conform very well to logseries.

# Figures

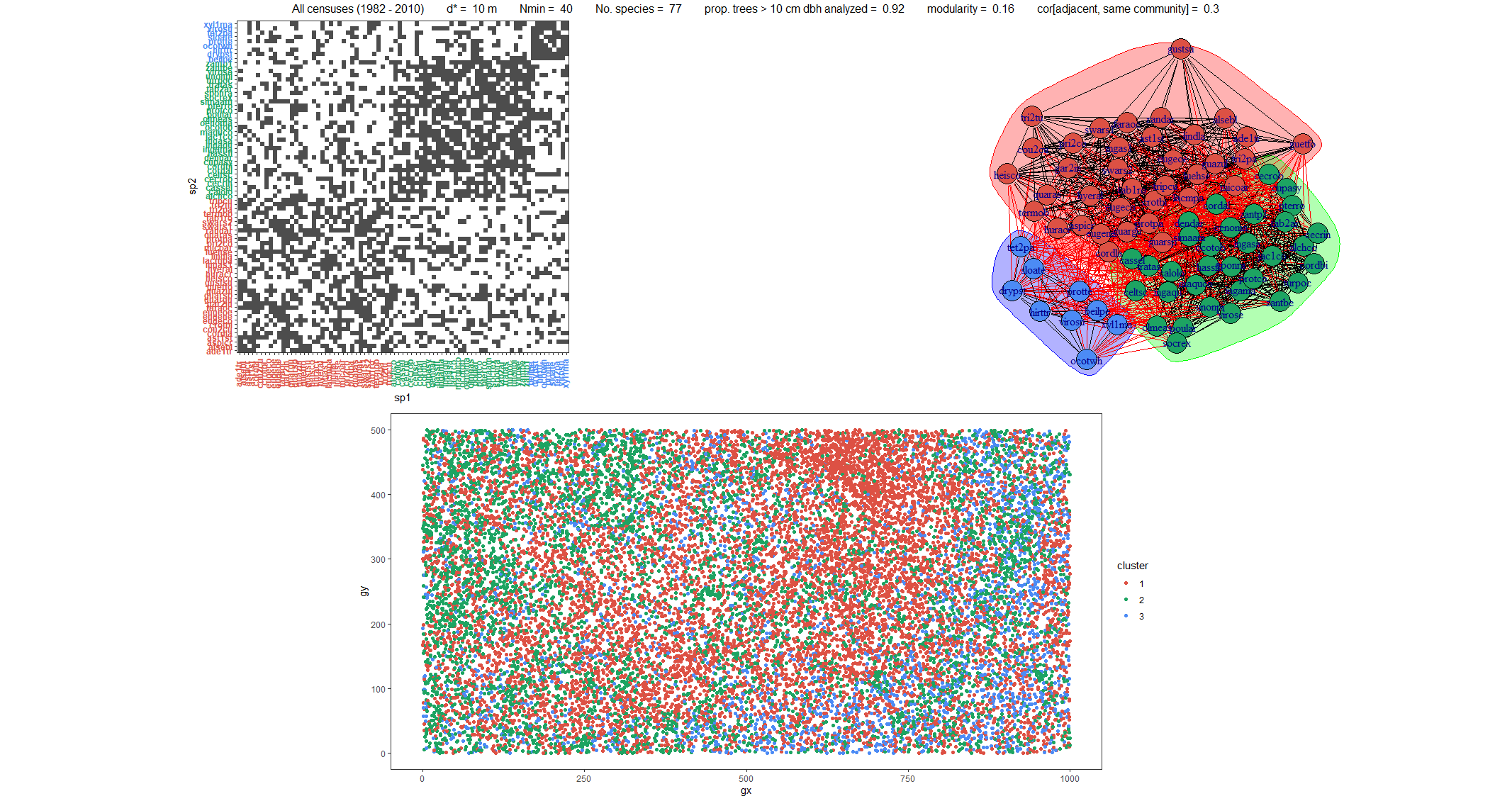


Figure 1. Top left: BCI FDP species organized by niches. Dark pixels indicate the species pair appears in close proximity more often than chance. Top right: same information represented as a graph, with the niches represented as colored blobs. Bottom: geographic distribution of the FDP species, colored by niches.

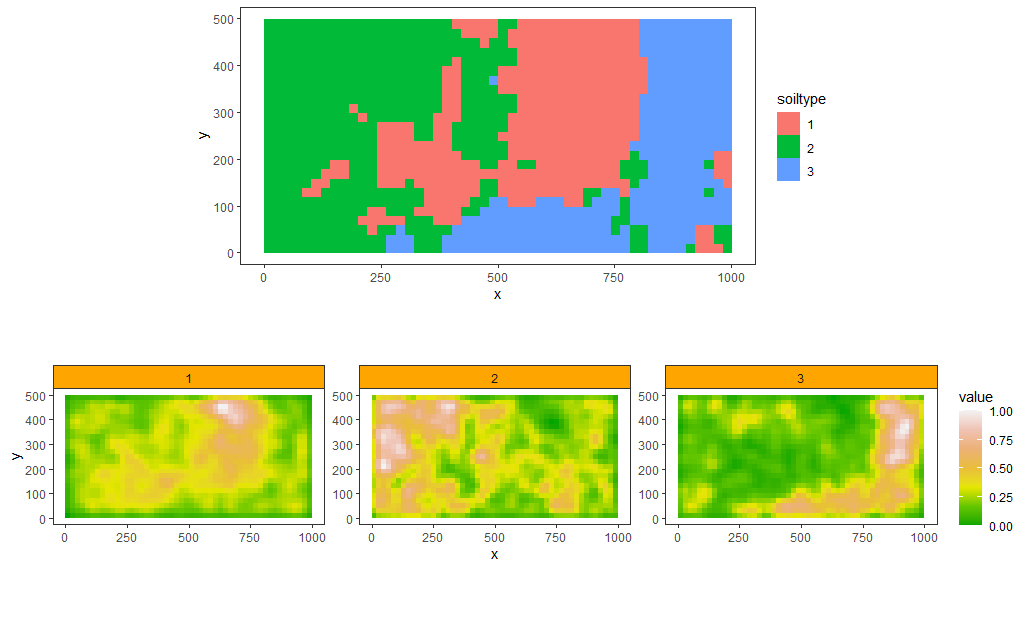


Figure 2. Top: Raster showing areas with each of the three soil types. Bottom: Rasters showing estimated local probability of each soil type across the FDP.

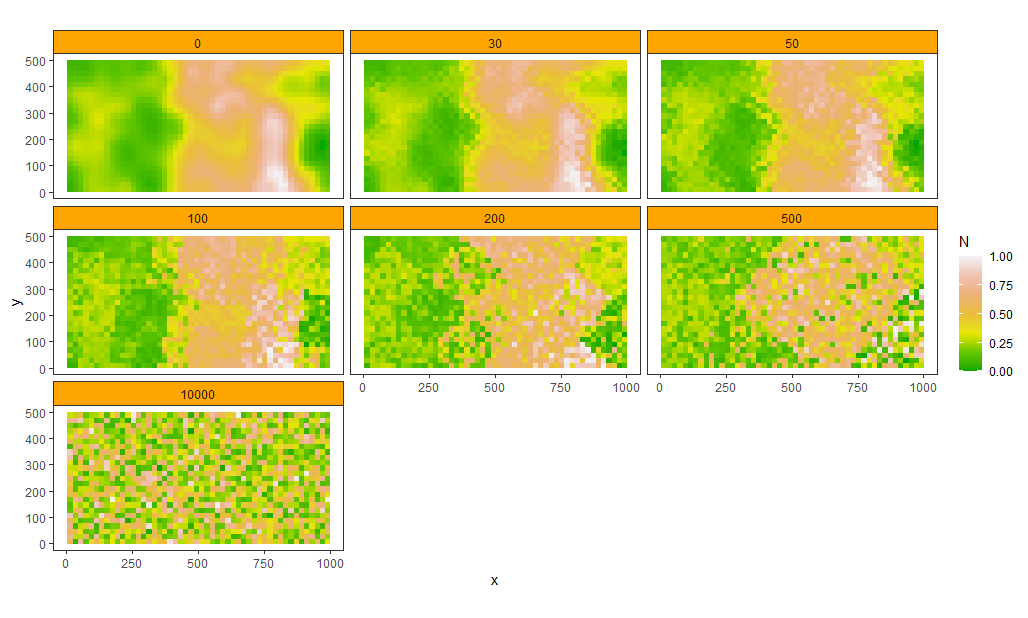


Figure 3. Illustration of the partial randomization process used to create null data to train the C5.0 classifier. Panels show different scales of randomization, such that the spatial autocorrelation is maintained at greater distances than that scale.

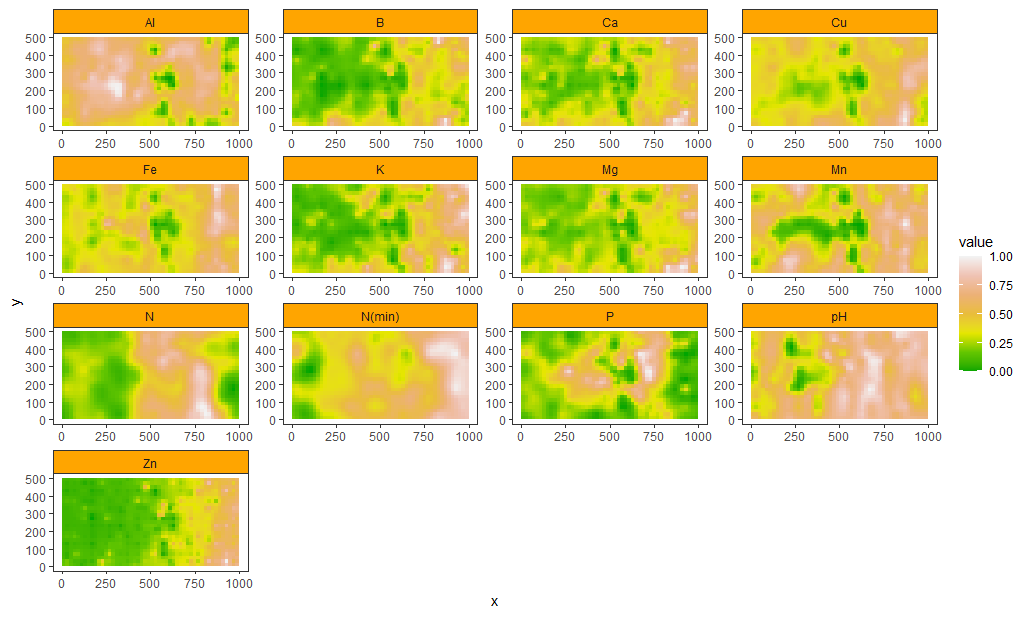


Figure 4. Rasters showing spatial distribution of each nutrient on the FDP.

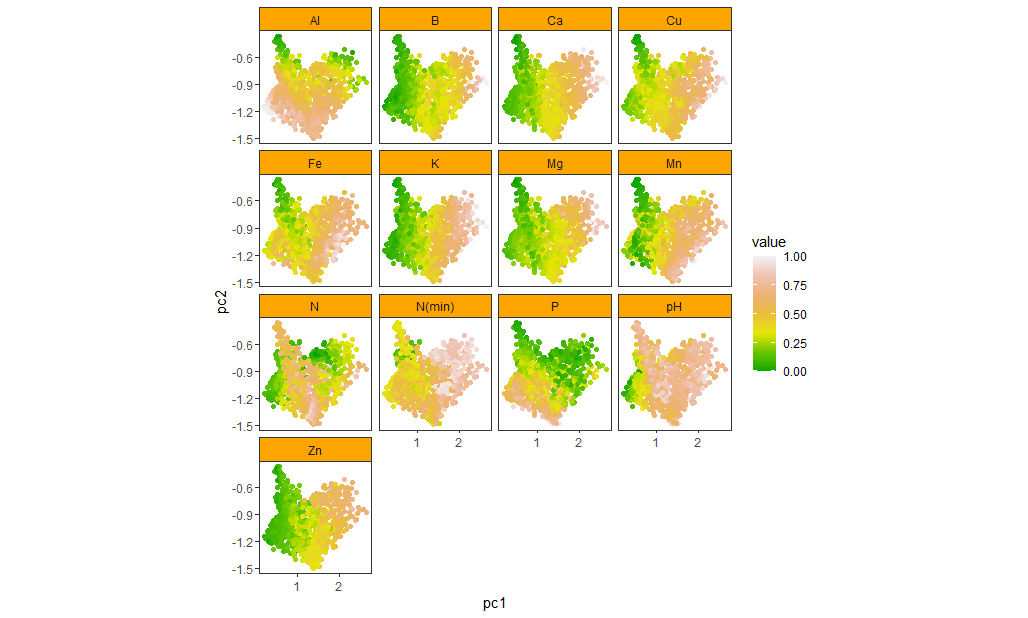


Figure 5. Representation of the nutrient concentration by the first two principal components. Each point represents a 20mx20m quadrat, colored by the local nutrient concentration. We see a clear separation between areas of the plot, and also substantial correlation between concentrations of different nutrients.

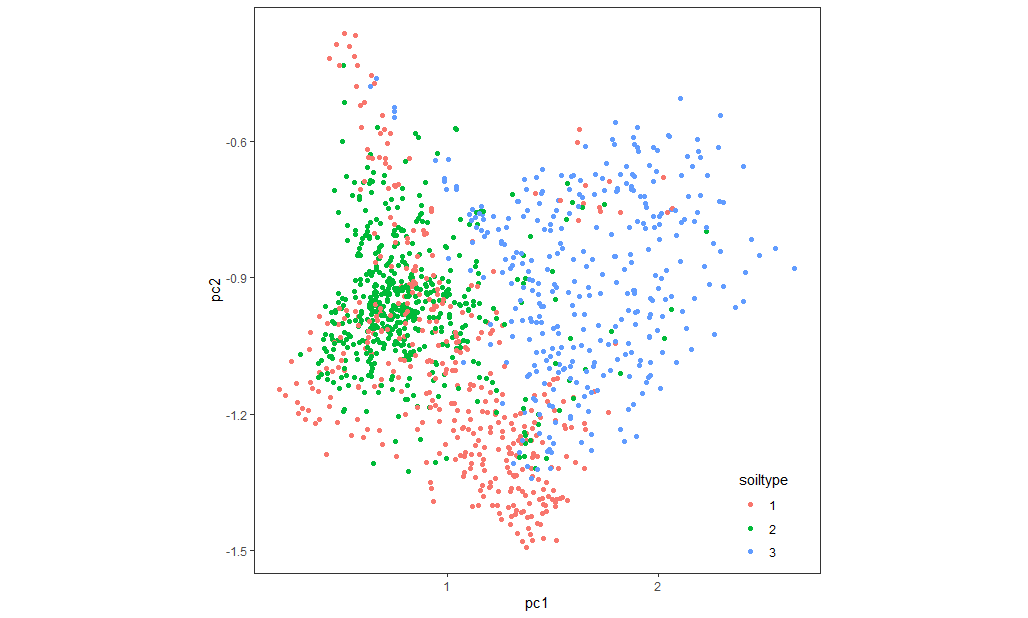


Figure 6. Same as Fig. 5, but with colors representing the quadrats’ respective soil type as per Fig. 2 (top). We see a clear link to nutrient concentration. Soiltype 1 is associated with high organic N and P, soiltype 2 with low nutrient areas, and soiltype 3 with high nutrient areas.

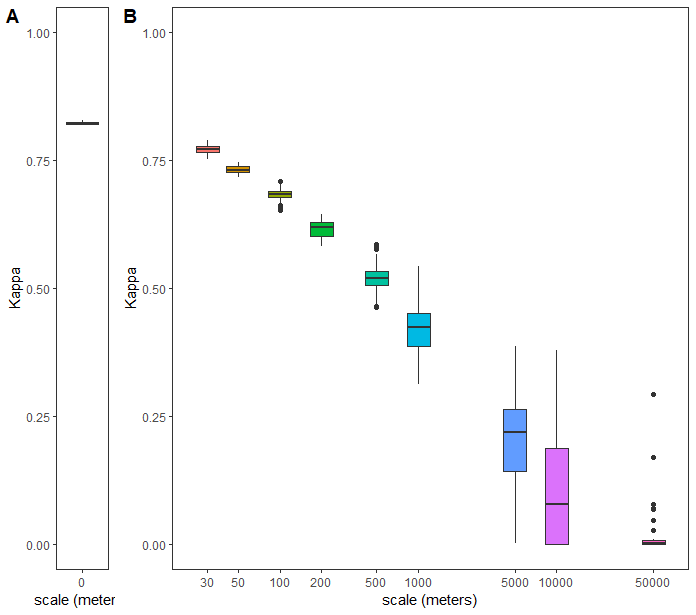


Figure 7. Cohen’s Kappa, which indicates the degree to which the classifier correctly predicts the soiltype of a quadrat, is much higher than its null expectation of 0. Kappa decreases as we train the classifier on partially randomized nutrient concentrations, with the decline being stronger at larger scales of randomization.

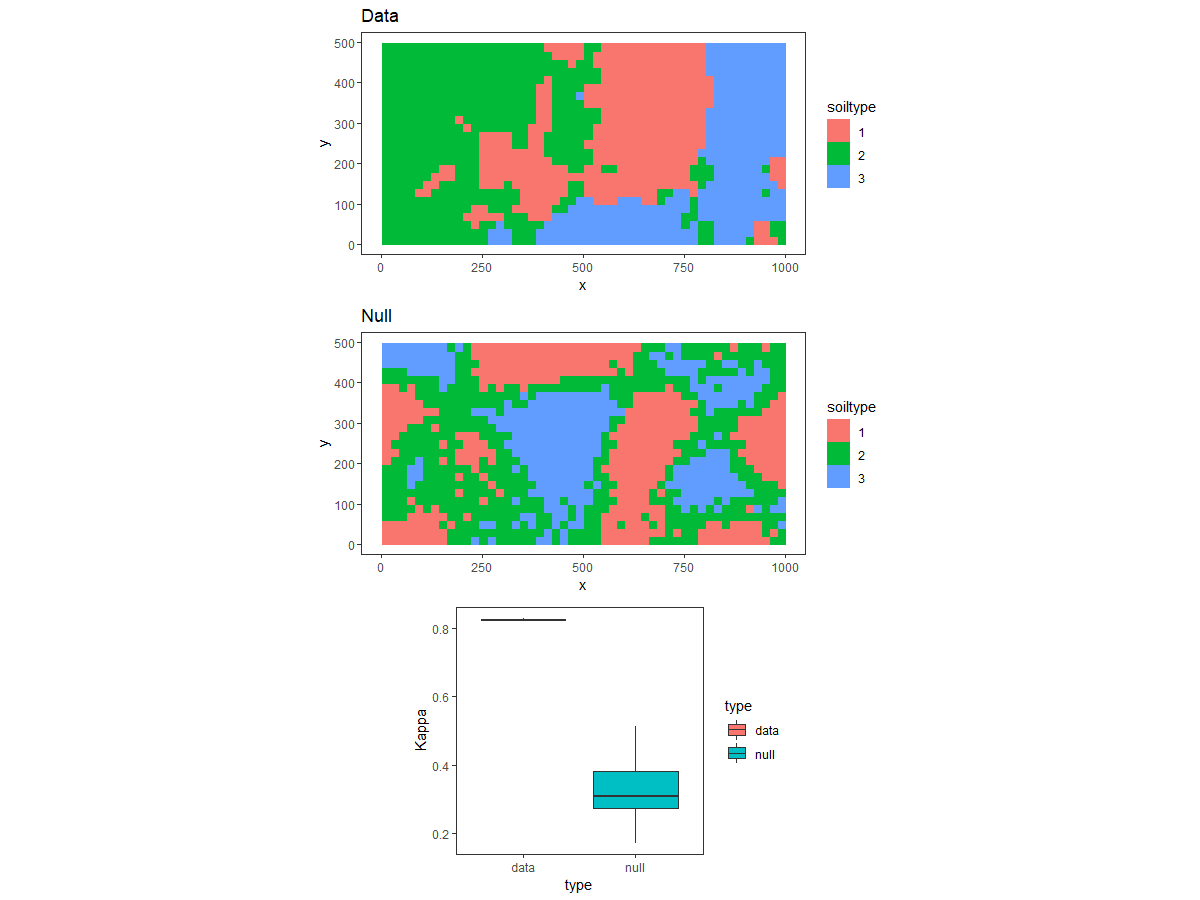


Figure 8. Top two panels show the raster of our inferred soiltypes (top) and an example null soiltype raster (middle). Both are autocorrelated with similar scales of autocorrelation and noise levels (variogram parameters are fit to the Data and then used to generate the Null). Bottom panel shows the distribution of Cohen’s Kappa over 46 (data) and 100 (null) iterations of the C5.0 classifier on both the data (red box) and null data (blue box). In both cases, Kappa > 0, signifying better-than-chance prediction of soiltypes from the nutrient concentrations even when the soiltypes are random data, and therefore confirming that the autocorrelation alone leads to accidental matches between nutrient data and soiltypes. However, there’s a very clear gap in the quality of predictions when using the real data.

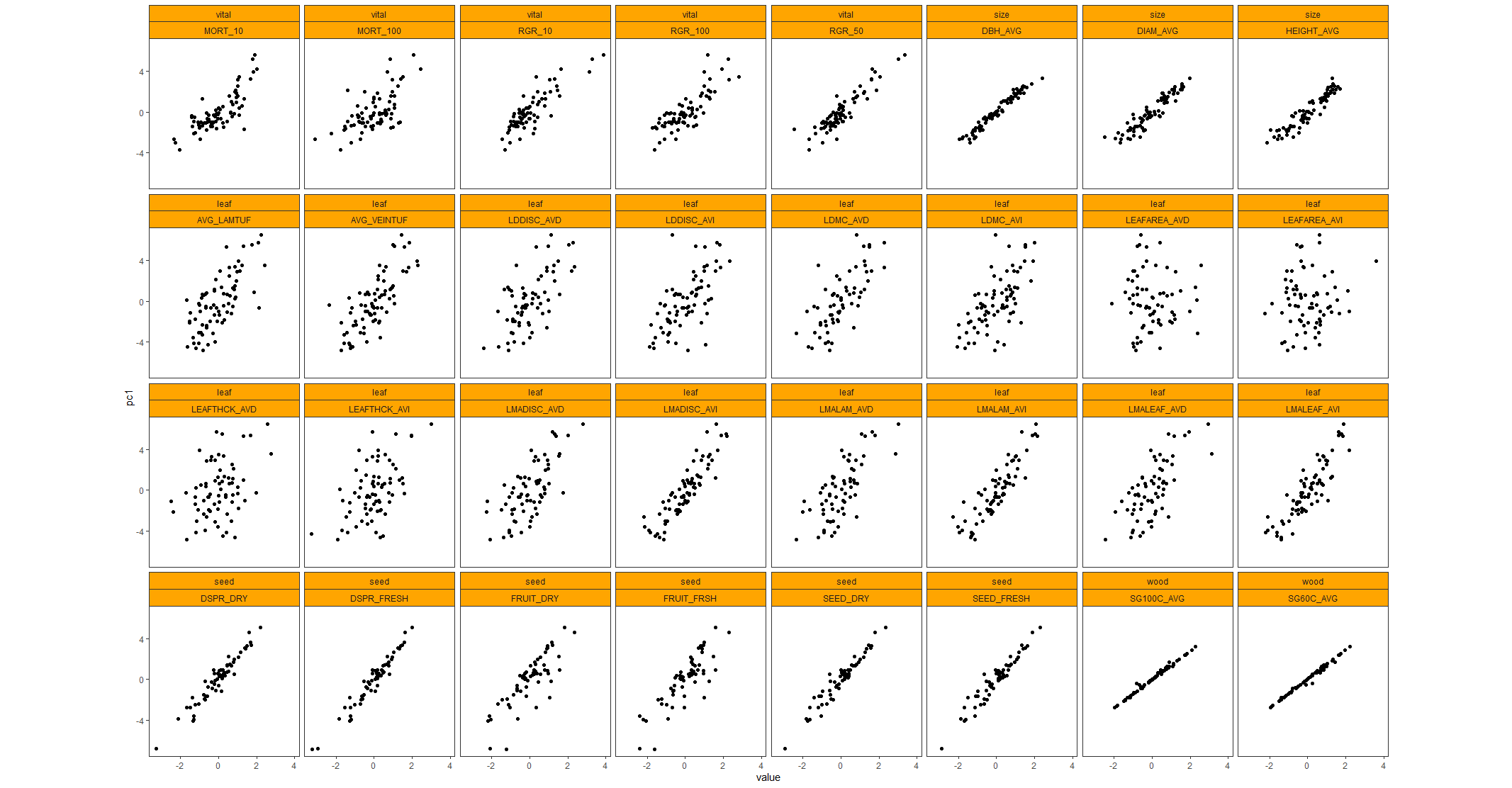


Figure 9. Correlation between the trait values and the first principal component of each trait category (vital rate, size trait, leaf trait, seed trait, wood trait). All traits correlate very strongly to its 1st component, indicating both that these traits are redundant and that PC1 is a good summary of the information contained in these trait measurements.

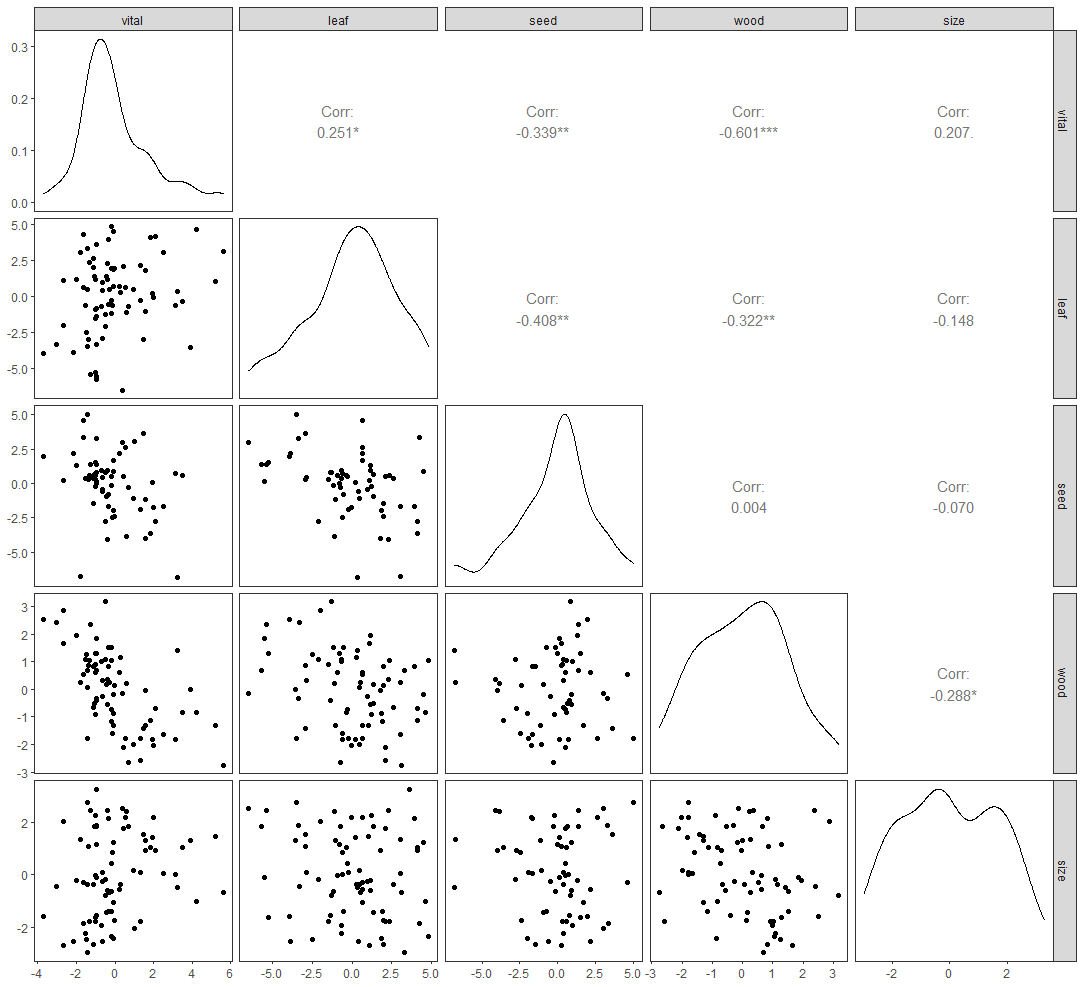


Figure 10. Correlations among first principal components of each trait type.

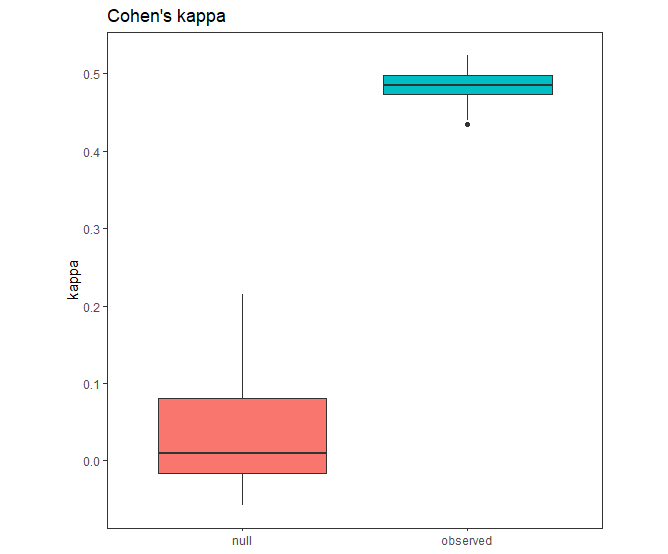


Figure 11. Cohen’s Kappa of the classifier predictions for observed trait data (blue boc) versus randomized trait data (red box). Distributions show 100 iterations of the classifier.

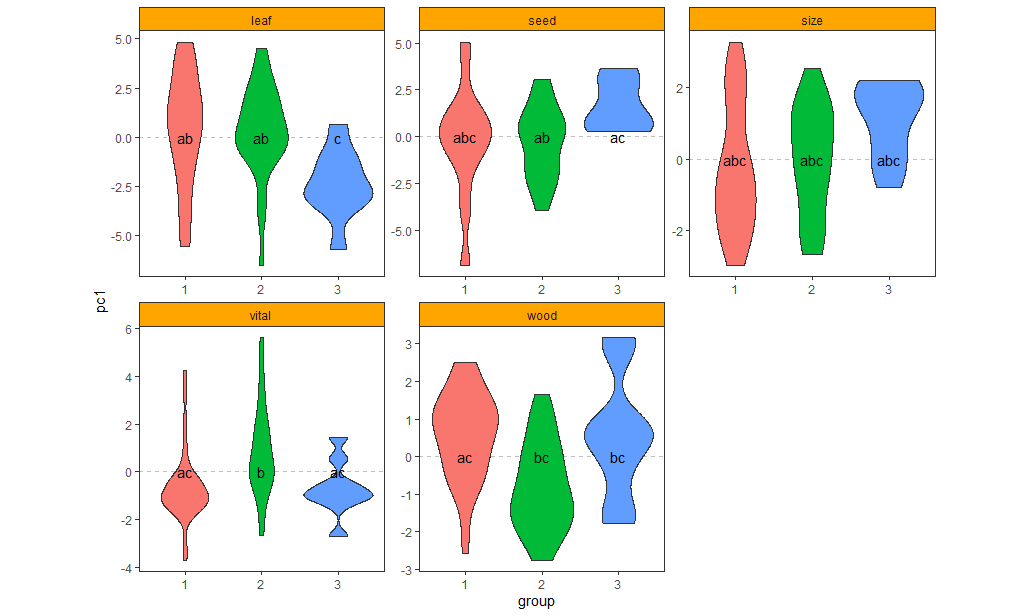


Figure 12. Trait distribution of each niches according to each trait type.

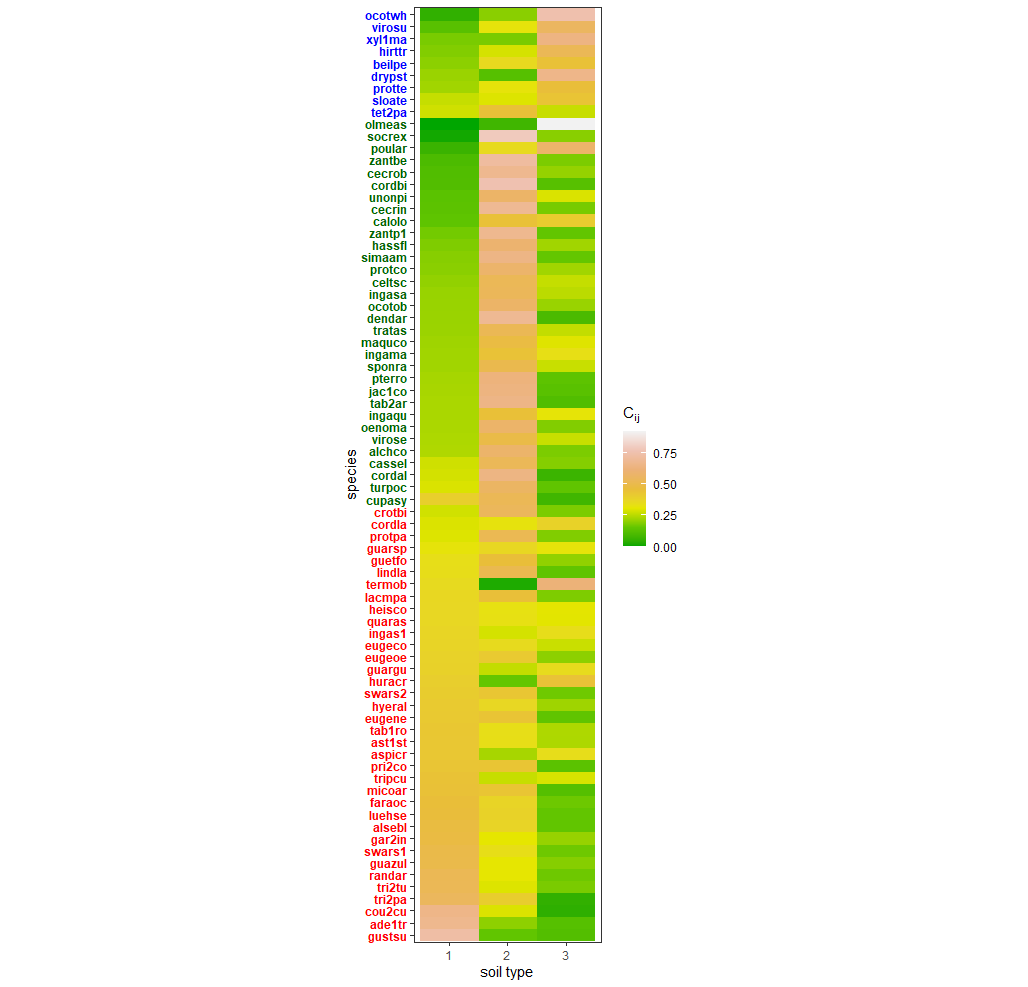


Figure 13. Affinity matrix reflecting how often each species occurs on each inferred soil type. Species codes are colored by niches.

Rules:

Rule 1: (10, lift 2.0)

vital <= -0.3237651

leaf > -0.8982342

seed <= 0.1063058

-> class 1 [0.917]

Rule 2: (5, lift 1.8)

vital <= -0.3237651

size > 2.213931

-> class 1 [0.857]

Rule 3: (4, lift 1.8)

vital > -0.3237651

wood > 0.8139341

-> class 1 [0.833]

Rule 4: (26/7, lift 1.5)

vital <= -0.3237651

size <= -0.1007147

-> class 1 [0.714]

Rule 5: (3, lift 1.9)

leaf > -0.8982342

seed > 0.1063058

seed <= 0.4120565

-> class 2 [0.800]

Rule 6: (36/10, lift 1.7)

vital > -0.3237651

-> class 2 [0.711]

Rule 7: (6, lift 7.5)

vital <= -0.3237651

leaf <= -0.8982342

size > -0.1007147

size <= 2.213931

-> class 3 [0.875]

Default class: 1

Evaluation on training data (77 cases):

Rules   
 ----------------  
 No Errors  
 7 10(13.0%)

(a) (b) (c) <-classified as

---- ---- ----

32 4 (a): class 1

3 29 (b): class 2

1 2 6 (c): class 3

Attribute usage:

98.70% vital

48.05% size

24.68% leaf

16.88% seed

5.19% wood

# July 22, 2021 Update

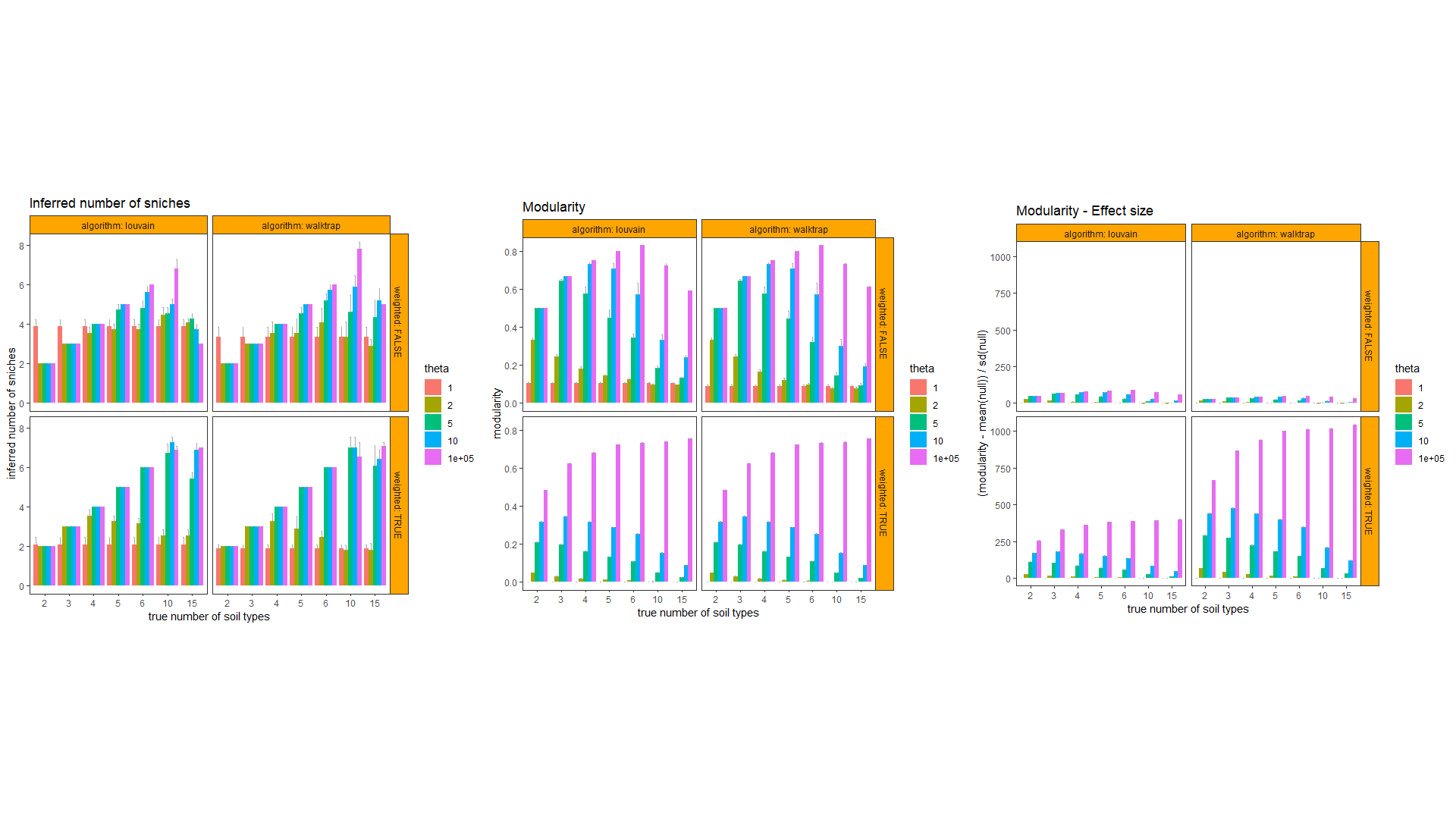


Figure 14. Validating our clustering algorithm

Fig. 14 shows results of applying our clustering method to communities that were generated via dynamic simulations. The dynamic step corresponds to an inter-census interval, and consists of killing about 10% of the trees (based on observed inter-census turnover rates on BCI) at random, followed by recruitment of a similar (but not necessarily identical) number of trees into the adult class (again, mirroring BCI). The recruitment stage is weighted by species abundance (as in the neutral scenario) AND preference for local soil condition. The latter is what connects the simulation to our ecological assumptions of sniches. Prior to the simulation start, we decide how many soil types (aka sniches) there are, and then each species is assigned a favorite soil type -- ie each species is given a sniche. When recruiting, a species will have odds of theta to 1 of recruiting in their preferred soil type relative to all other soil types (making soil types a categorical rather than an ordinal characteristic). So basically our simulated data has two free parameters:

1. true number of sniches
2. degree of sniche specialization

I seed the community like a normal dynamic step, starting with the recruitment stage and assuming identical initial abundances. Then I simulate each community for 100 census. There are 105 species initially, and the initial community size is 20,678 trees, matching the mean community size across censuses on BCI for trees > 10 cm dbh.

**Results**

On the left I am plotting the inferred number of sniches. In the middle I am plotting the observed modularity for that best number of sniches. On the right I am plotting the modularity *effect size*, defined as (mod - mean(null\_mod) ) / sd(null\_mod)  where null\_mod is a set of null values for the modularity. Those null values were obtained from the neutral runs -- i.e. by simulating the community with theta = 1.The plots are paneled by clustering algorithm (columns) and whether the adjacency matrix is weighted or binary (rows).

The values of theta I used are meant to span the realm of possibilities, from theta = 1 reflecting neutrality and theta = 1e5 representing full specialization (very unlikely for a species to recruit anywhere other than their matching soil type).

We see many interesting things.  
  
**Left plot**

* For some reason, the clustering algorithms like to assign 4 clusters to neutral data when using a binary adjacency matrix, and 2 when using a weighted one. I'm not sure what my prior expectation should have been --- A uniform (or normal) distribution of # of sniches across censuses, or a specific # of sniches? If the latter, why 4 (2)?
* Even under perfect specialization, the algorithms fail to find the true # of sniches beyond a certain value.
* Using weighted adjacency matrices usually gives more accurate inferences on the # of sniches.
* In terms of louvain vs walktrap, louvain is more accurate when theta is low and # of sniches is high (not in terms of getting it right more often, but of underestimating the true # of sniches by less). Meanwhile, walktrap is more accurate when theta is high and # of sniches is high. When true # of sniches is low (<= 5), they have similar accuracy.

**Middle plot**

* Observed modularity is overall higher when using binary adjacency matrices.
* Modularity goes up as the true # of sniches increases from 2 upwards, then peaks, then falls back down for ever increasing # of sniches. The position of the peak depends on the degree of sniche specialization, occurring at higher # of sniches when theta is higher.

**Right plot**

* In terms of the modularity effect size -- i.e. contrasting observed modularity against null expectations, the picture is reversed in terms of weighted vs binary matrices: we get much higher distinction from null data when using the weighted matrix. This suggests a bias towards higher modularity in binary vs weighted networks. The fact that the effect size is higher for weighted matrices indicates that the weighted  test is more accurate, and we should stick with it.
* Like absolute modularity, we see a humped shape in effect size relative to the true # of sniches. This shows the ability of our algorithm to detect the underlying clustered structure saturates at an intermediate number of sniches, then drops.
* Walktrap does a better job at distinguishing the simulations from null data than louvain.

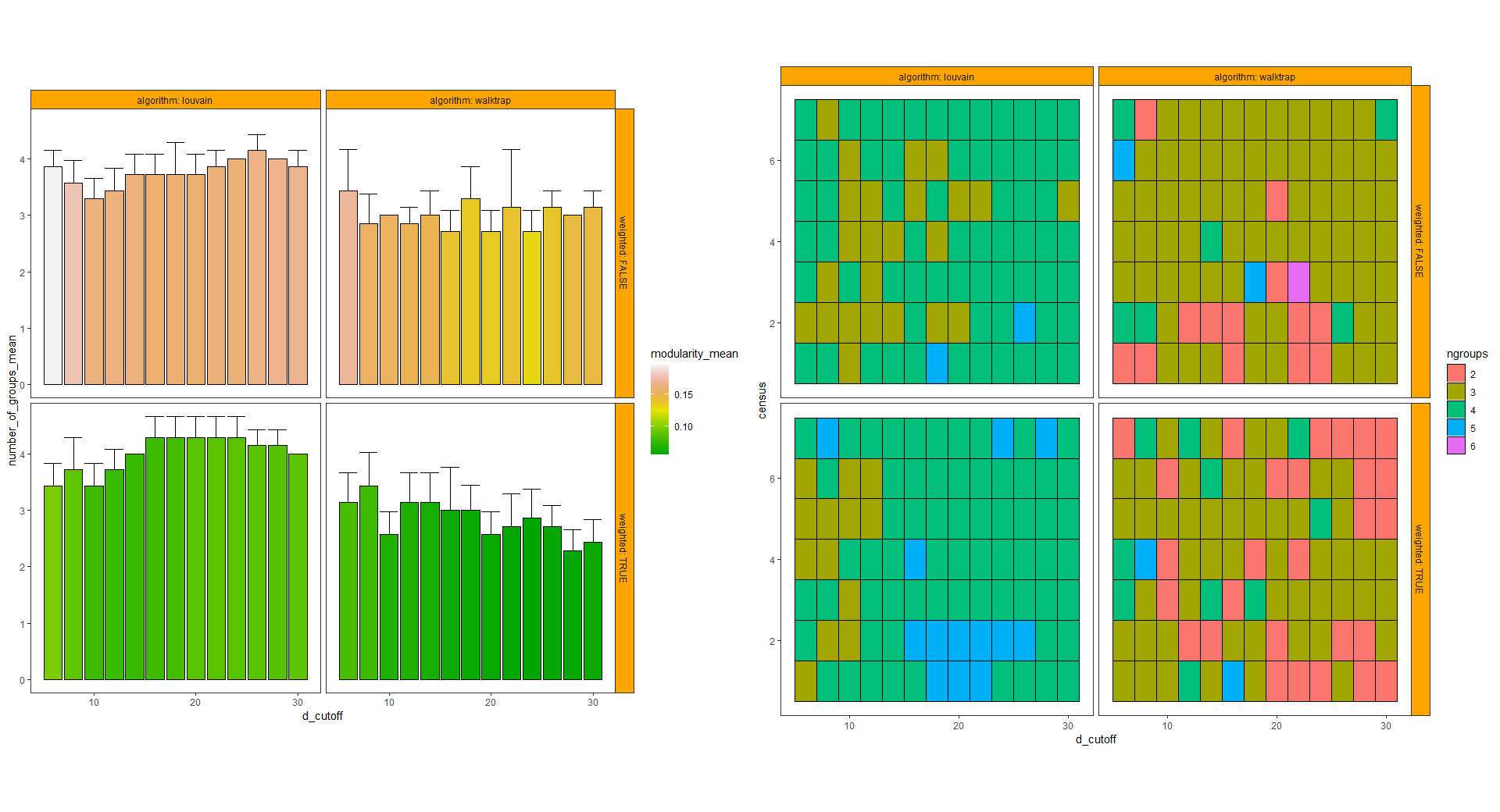


Figure 15. BCI results

**Left plot**  
The bars show the mean (across censuses) inferred # of sniches by distance cutoff used in the analysis. Error bars are 2 \* standard error of the mean. Bars are colored by the mean modularity (across censuses).

* We see that the inferrence is relatively stable for a range of d\_cutoff.
* We also see that louvain and walktrap disagree as to the # of sniches. Louvain likes 4, walktrap likes 3.
* When we use weighted adjacency matrices we get lower modularity.

**Right plot**  
You've seen this plot before. The grid is census vs d\_cutoff, and the colors show inferred # of sniches. So this shows the same data as the left plot in more granular fashion (and without the modularity).

* The point here is to see that despite some variation in inferred #sniches, that quantity is pretty consistent across censuses (meaning it doesn't swing from 3 to 10), especially under louvain.

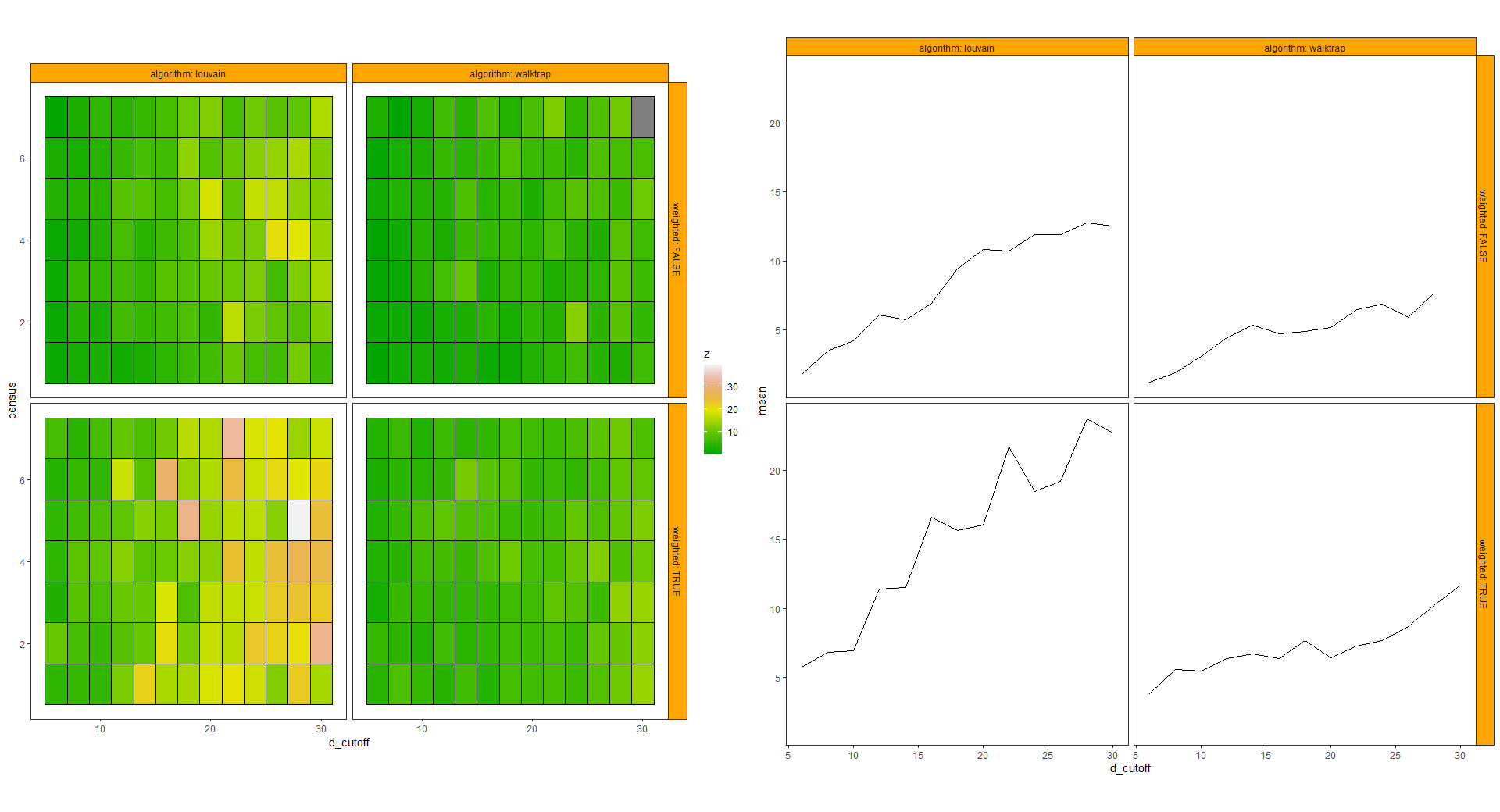


Figure 16. BCI results

**Left plot**  
Same scheme as the Fig. 15, but now colored by effect size.

* We see clearly that we get higher effect  size when adjacency matrix is weighted (vs binary) and when using louvain vs walktrap.

**Right plot**

* The average effect size (across censuses) increases with d\_cutoff. I was hoping to use the effect size as the index to tell us what cutoff to use -- as I expected it would peak at d\_cutoff ~= 10 to 20 --- but alas that is not what we see. Rather, for some reason it seems to increase monotonically with d\_cutoff.