BIO 423 - Lab 8

Benjamin Blonder

Spring 2019

Learning outcomes

Content goals:

- Summarize trait variation across sites
- Determine how environmental variation drives trait variation

R goals:

- Become more familiar with `tidyverse` functions
- Become more familiar with join operations
- Make predictions using linear models

This week we will learn how to work with trait-environment relationships, a key foundation of functional ecology. Our datasets will come from the Smithsonian Tropical Research Institute's Barro Colorado Island (BCI), a site that has been intensively studied for many years. On BCI there is a 50-hectare forest plot in which all individual plant stems are censused. This plot is divided up into 50 1-hectare subplots. We will be working with these data pre-summarized at subplot level for a single year's census.

We will also be working with spatial data for environmental variation across the subplots - both physiographic factors such as elevation, slope, and aspect, as well as soil mineral factors such as the concentration of various trace elements, as well as soil pH.

We will be seeking to understand how the environment predicts the functionining and identity of species on the island.

Data come from:

`http://www.sciencemag.org/cgi/content/full/295/5555/666/DC1`

`https://datadryad.org/handle/10255/dryad.81868`

`https://www.davidzeleny.net/anadat-r/doku.php/en:data:bci`

and are encapsulated also in the `vegan` and `BiodiversityR` packages.

Loading in the census data

```
library(vegan)
```

```
## Loading required package: permute
```

```
## Loading required package: lattice
```

```
## This is vegan 2.5-3
```

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------- tidyverse 1.2.1 --
```

```
## v ggplot2 3.1.0     v purrr   0.2.5
## v tibble  1.4.2     v dplyr   0.7.8
## v tidyr   0.8.2     v stringr 1.3.1
## v readr   1.3.0     v forcats 0.3.0
```

```
## -- Conflicts ------------------------------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(ggplot2)

# load the BCI data built-in from the vegan package
data(BCI)
BCI$Plot=as.numeric(row.names(BCI))
# note that the data are in 'wide' format (site x species matrix)
dim(BCI)
```

```
## [1]  50 226
```

```
# convert the data from long to short format
BCI_short = BCI %>%
              gather(key="Species",value="Abundance",-Plot)
# or using the reshape2 package (load it first), equivalently...
# BCI_short = melt(BCI, id.vars="Plot", variable.name="Species",value.name="Abundance")

head(BCI_short)
```

```
##   Plot            Species Abundance
## 1    1 Abarema.macradenia         0
## 2    2 Abarema.macradenia         0
## 3    3 Abarema.macradenia         0
## 4    4 Abarema.macradenia         0
## 5    5 Abarema.macradenia         0
## 6    6 Abarema.macradenia         0
```

```
# keep only species with non-zero abundance, and reorder for clarity
BCI_short_nonzero = BCI_short %>% filter(Abundance>0) %>% arrange(Plot, Species)

head(BCI_short_nonzero)
```

```
##   Plot                Species Abundance
## 1    1 Alchornea.costaricensis         2
## 2    1        Alseis.blackiana        25
## 3    1         Annona.spraguei         1
## 4    1           Apeiba.glabra        13
## 5    1        Apeiba.tibourbou         2
## 6    1    Astronium.graveolens         6
```

Load in trait data

To summarize data at community level, we can consider calculating community-weighted means (CWM) of traits. These can be represented as

$$CWM_t = \frac{\sum_i^n t_i N_i}{\sum_i^n N_i}$$

where t_i is the value of trait t for species i and N_i is the abundance of species i.

Such metrics bring the mean trait value closer to that of the most common species, and are useful for summarizing the functional properties of the overall community.

To calculate these, we will need to join the trait data with the by-site abundance data - look out for how this is done below!

```
traits = read.csv("BCI.trait.csv")
str(traits)
```

```
## 'data.frame':    286 obs. of  15 variables:
##  $ genus_species    : Factor w/ 286 levels "Abarema_macradenia",..: 1 2 3 4 5 6 7 8 9 10 ...
##  $ FAMILY.          : Factor w/ 58 levels "Acanthaceae",..: 23 21 21 21 56 21 21 46 49 46 ...
##  $ GENUS.           : Factor w/ 175 levels "Abarema","Acalypha",..: 1 2 2 3 4 5 5 6 7 8 ...
##  $ SPECIES.         : Factor w/ 241 levels "aculeata","acuminata",..: 134 75 139 232 165 67 125 79 18
##  $ Shade_LaminaTough: num  NA 478 196 330 176 ...
##  $ Shade_VeinTough  : num  NA 2116 1654 1713 1504 ...
##  $ Sun_LaminaTough  : num  367 NA 222 303 NA ...
##  $ Sun_VeinTough    : num  2452 NA 1217 2396 NA ...
##  $ SEED_DRY         : num  NA NA NA 0.01225 0.00115 ...
##  $ LEAFTHCK_AVI     : num  NA NA NA 121 170 ...
##  $ LEAFTHCK_AVD     : num  NA NA NA 152 NA ...
##  $ abundance        : int  1 530 42 219 77 226 2 379 122 8171 ...
##  $ sigma            : num  0 89.6 72.8 28.7 75.3 ...
##  $ omega            : num  NA 3.97 85.31 17.05 5.97 ...
##  $ N1995            : int  NA 530 42 219 77 227 2 379 NA 8171 ...
```

```
# create a Species column to match the Species column in BCI_short_nonzero

# this requires changing underscores to periods
# the fixed=TRUE argument means the substitution will interpret the
# find/replace exactly, rather than interpreting them as 'regular expressions'
# which allow matching of more complex patterns (e.g. all numbers, all spaces)
traits$Species = gsub("_",".",traits$genus_species,fixed=TRUE)

# now join the trait data and the abundance data
# we use inner join to keep only cases where species and traits are matched
BCI_merged = inner_join(BCI_short_nonzero, traits,by="Species")

# calculate community weighted means
BCI_summary_trait = BCI_merged %>%
                      group_by(Plot) %>%
                      summarize(CWM.seed.mass=
                                  sum(SEED_DRY*Abundance,na.rm=T)/sum(Abundance,na.rm=T)
                                )
```

Load environmental data

```
# load environmental data as a tab-separated file
BCI.env = read.csv('BCI.env.txt',sep='\t')
BCI.env$Plot = as.numeric(row.names(BCI.env))
# also add in soil data
BCI.soil = read.csv('BCI.soil.txt', sep='\t')
```

```
BCI.soil$Plot = as.numeric(row.names(BCI.soil))
# make a combined environmental dataset
BCI_env_combined = inner_join(BCI.env,BCI.soil,by="Plot")
head(BCI_env_combined)
```

```
##   UTM.EW  UTM.NS elevation   convex    slope    aspectEW   aspectNS Plot
## 1 625754 1011569  130.2525  -7.8725 6.694828 -0.89108252 -0.4538413    1
## 2 625754 1011669  136.8100 -10.7000 5.086842 -0.21903766 -0.9757164    2
## 3 625754 1011769  143.6775 -14.6675 3.104794  0.03051372 -0.9995343    3
## 4 625754 1011869  147.0075 -16.7575 1.872813 -0.86414183 -0.5032483    4
## 5 625754 1011969  144.3850 -12.4850 5.118725 -0.67148116  0.7410216    5
## 6 625854 1011569  136.8750  -9.6850 2.945532 -0.86532324 -0.5012142    6
##     x   y        Al       B       Ca      Cu       Fe         K       Mg
## 1  50  50  901.0908 0.79448 1680.021 6.20312 135.2870 141.88128 279.1291
## 2  50 150  954.2488 0.66968 1503.365 6.03148 141.8080 137.23932 280.4524
## 3  50 250 1114.1122 0.59516 1182.311 6.79768 157.0878  98.69056 230.3973
## 4  50 350 1023.5793 0.56780 1558.020 6.63400 153.1746  98.36412 228.9468
## 5  50 450 1001.8848 0.39876 1242.251 6.44428 149.2509  94.07208 202.6820
## 6 150  50 1091.4672 0.73120 1441.977 6.49552 173.8682 131.89280 276.5010
##         Mn       P      Zn        N   N.min.      pH
## 1 266.9997 1.95248 2.96948 18.46500 -3.88544 4.32432
## 2 320.4786 2.24740 2.53208 21.59896  5.64388 4.37548
## 3 445.0708 1.95484 2.24672 20.24516 -4.06408 4.34700
## 4 407.7580 2.63444 2.44284 20.84232  7.89012 4.46112
## 5 250.5403 1.86356 2.13748 16.94500  8.53716 4.40128
## 6 477.3249 1.61612 2.63148 20.29812  4.38948 4.57252
```

Begin analyses

We now have several datasets: `BCI_env_combined`, which contains information on the abiotic environment and the spatial locations of subplots 1-50, as well as `BCI_summary_trait`, which contains information on CWM trait values for subplots 1-50, as well as the raw abundance and trait datasets.

We can now make a combined trait-environment dataframe, the heart of the work that will come:

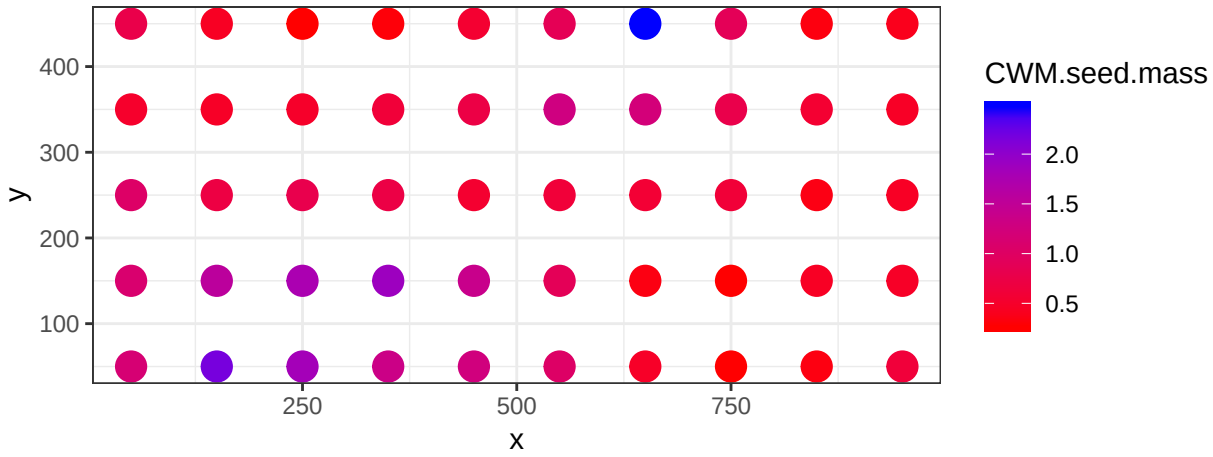
```
BCI_summary_trait_env = inner_join(BCI_summary_trait, BCI_env_combined, by="Plot")
str(BCI_summary_trait_env)
```

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    50 obs. of  24 variables:
##  $ Plot         : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ CWM.seed.mass: num  1.159 1.116 1.034 0.51 0.773 ...
##  $ UTM.EW       : num  625754 625754 625754 625754 625754 ...
##  $ UTM.NS       : num  1011569 1011669 1011769 1011869 1011969 ...
##  $ elevation    : num  130 137 144 147 144 ...
##  $ convex       : num  -7.87 -10.7 -14.67 -16.76 -12.48 ...
##  $ slope        : num  6.69 5.09 3.1 1.87 5.12 ...
##  $ aspectEW     : num  -0.8911 -0.219 0.0305 -0.8641 -0.6715 ...
##  $ aspectNS     : num  -0.454 -0.976 -1 -0.503 0.741 ...
##  $ x            : int  50 50 50 50 50 150 150 150 150 150 ...
##  $ y            : int  50 150 250 350 450 50 150 250 350 450 ...
##  $ Al           : num  901 954 1114 1024 1002 ...
##  $ B            : num  0.794 0.67 0.595 0.568 0.399 ...
##  $ Ca           : num  1680 1503 1182 1558 1242 ...
```

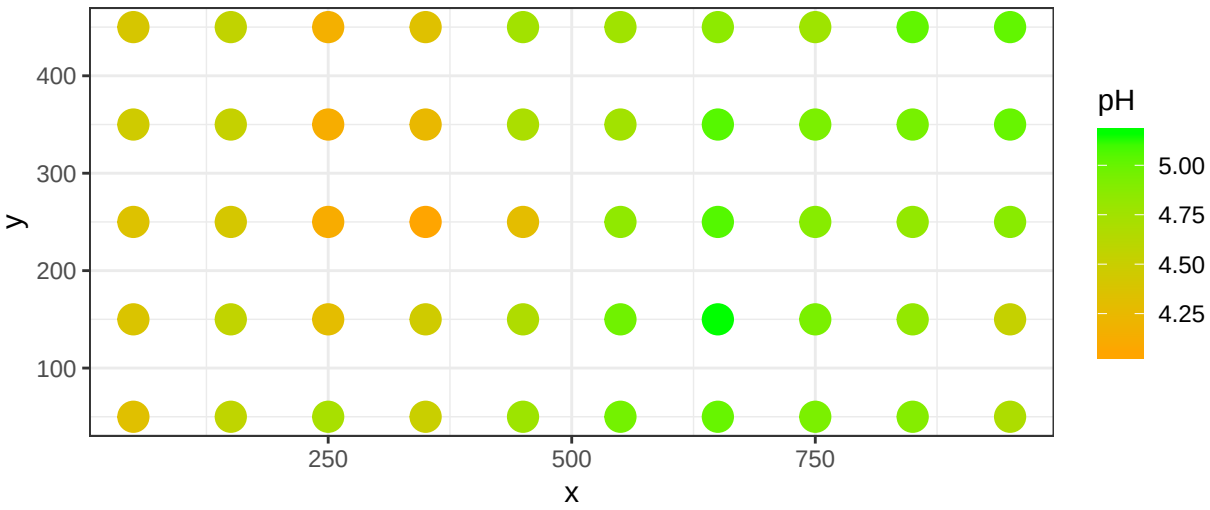
```
##  $ Cu           : num  6.2 6.03 6.8 6.63 6.44 ...
##  $ Fe           : num  135 142 157 153 149 ...
##  $ K            : num  141.9 137.2 98.7 98.4 94.1 ...
##  $ Mg           : num  279 280 230 229 203 ...
##  $ Mn           : num  267 320 445 408 251 ...
##  $ P            : num  1.95 2.25 1.95 2.63 1.86 ...
##  $ Zn           : num  2.97 2.53 2.25 2.44 2.14 ...
##  $ N            : num  18.5 21.6 20.2 20.8 16.9 ...
##  $ N.min.       : num  -3.89 5.64 -4.06 7.89 8.54 ...
##  $ pH           : num  4.32 4.38 4.35 4.46 4.4 ...
```

To illustrate what we can do with these data, let's make a few quick maps - one of a trait, one of an environmental variable.

```
ggplot(BCI_summary_trait_env,aes(x=x,y=y,col=CWM.seed.mass)) +
  geom_point(size=5)+
  coord_equal()+
  scale_color_gradient(low = "red",high="blue") +
  theme_bw()
```




```
ggplot(BCI_summary_trait_env,aes(x=x,y=y,col=pH)) +
  geom_point(size=5)+
  coord_equal()+
  scale_color_gradient(low = "orange",high="green") +
  theme_bw()
```



Questions

1. Which species has the highest trait value for seed mass? Look up the species and include a photograph of this species' seed.
2. Make a map of CWM shade leaf toughness. At what coordinates is toughness highest?
3. Which of the soil trace minerals boron or phosphorus has the strongest effect on shade leaf vein toughness? Use a linear regression and explain how you decided what you meant by 'strongest effect'. Write a one sentence plain-English summary explaining the effect of this element on the trait values.
4. Suppose there is some local erosion and one of the subplots reaches a slope of 15 degrees. What do you predict the CWM seed mass would become? You should build a linear model, then either use `predict(mymodel, data.frame(myvariable=newvalue))` or extract the coefficients of a linear model using `coef` and do the calculation by hand using slope-intercept form.
5. What if there is a landslide and the slope reaches 25 degrees - what CWM seed mass do you predict? If this is an unrealistic prediction, explain why, and suggest (but do not implement) a strategy for how you could make a better model.
6. Consider the 15 degree slope case again. How many species in the total species pool have seed mass values below the mean prediction? What fraction of the total number of species (with trait measurements) is this? (Hint: use `filter(!is.na(SEED_DRY))`)
7. Use your answer to the previous question to explain how environmental filtering may influence community assembly at this site, in 2-3 sentences.

Optional questions for graduate students

- Do a principal components analysis to determine the leading axis of soil variation across the plot. How much of this variation is explained by elevation or slope?
- Repeat the trait analyses in question 3 and 4 but correcting for spatial autocorrelation, e.g. with generalized least squares regression (hint, use `gls` and `corClasses`).
- Calculate functional richness in each plot, e.g. using the `convhulln` function in the `geometry` package. You may want to rescale the trait data using `scale` first in order to make variables with different units/scales comparable to each other. Then determine which plot has the highest functional richness, and whether topographic factors are drivers of richness.

- Determine whether the observed functional richness in each plot is significantly higher or lower than expected by chance if species are randomly assembled from the species pool. You'll have to decide how to best make the null model - should you fix the total abundance of individuals? Species? Once you do the analysis, interpret the findings in terms of ecological processes (Hint: read Mayfield & Levine 2010 in Ecology Letters first).

What to hand in

- A single Word Document including:
- written answers (1-2 sentences) and figures for each question above
- A copy of your R script (the contents of your `.R` file pasted into the Word document)
- Author contribution statement