# workshop: Community ecology in R

## Background Information

This workshop is based on an online tutorial developed alongside data described in the following paper;

Kembel & Cahill (2011) Independent evolution of leaf and root traits within and among temperate grassland plant communities. PLoS ONE 6(6): e19992.

This document includes all the code for the workshop. I recommend completing it in the order presented here, but you may wish to skip ahead to sections of interest. If you do this make sure you have loaded the required packages and data files. The focus of this workshop is on completing the practical aspects of analysis and assumes at least an undergraduate background in biology and statistics. Therefore, it is recommended that you do your own reading on the theoretical aspects of any topics you are unfamiliar with.

Before we start it is good practice to clear your environment in R or R Studio. Start a new script for this workshop and save it from the start. I also recommend annotating your script - I have added my annotations but it’s good to write it in your own words. Use # to prevent the annotation from being read in R as code. Annotation will help you understand the commands as you use them and is handy as a future reference guide.

We need to make sure the different packages we are going to use are loaded. We will be using functions from the **ape**, **picante**, and **vegan** packages. Since **picante** depends on the other two packages, loading it will load the other two as well. If you do not have the package installed, you can use Tools>Install Packages to search and install it. Once it’s installed load it in R using the library() function.

library(picante)

Download the files from the Learning Hub in the folder Grassland Data. Make a note of the location of these files on your computer. To make it easier to load files, we can set our working directory to the folder containing the grassland data. The exact format of a filename will vary depending on your operating system. The format below works for Windows and is the path to where I have the file on my computer.

setwd("C://Users//Heather// OneDrive - Harper Adams University//Harper//HAU Teaching//C7049 Ecological Entomology//2021 Ecological Entomology//Community Ecology Workshop")

You may need to change this format for Mac or Linux. Remember you will also need to change the location to wherever you put the files on your system. Alternatively, you could use the file.choose() function for each of the file-reading commands in this tutorial to interactively select files to load, rather than setting the working directory and writing out the filenames.

## Working with multiple data sets

We are going to analyse data on the biodiversity of grassland plants in Alberta. However, the same process could be applied to a data set on any organism in any location. Community analysis often combines multiple data sets. This workshop will use the following four data files.

1. grassland.community.csv

This is the basic grassland community data set. These data are usually a site x species matrix and can contain any measure of abundance or presence/absence. In this case it is percentage cover of different plant species in 20x20m quadrats in grasslands in different habitat types.

2. species.traits.csv

Trait data can be used to infer the function or adaptation of organisms to their environment. In this case the data are leaf and root traits of each species.

3. plot.metadata.csv

Metadata consists of any additional information that has been measured about specimens collected or the environment of the sample sites. In this example the metadata contains information on the habitat and site of quadrats and basic environmental variables, such as slope and moisture.

4. grassland.phylogeny.newick

Ecological analysis and interpretation of data are strengthened by including an evolutionary perspective. In cases where taxa are well studied there may be an existing phylogeny or one could be reconstructed from available data. Here we will use a known phylogeny for the species found in the community data set.

We will begin the workshop by loading and exploring each of the data sets, before we move on to analysing them in combination with each other. Make notes on the individual and combination data files as you go along to help keep track of what information each one contains.

## Community data

Ecological community data consist of observations of the (relative) abundance of species in different samples. In our case, the abundance measure is percentage cover of different plant species in 20x20m quadrats in grasslands in different habitat types. The format for community data is a data frame with samples in the rows and species in the columns. Our data are already in this format so we can load them using the following command. Note that since we've set our working directory to the folder containing all the data files, we just type the filename.

#read community data

#use plot IDs as rownames (first column of data)

#use species names as colnames (default read.csv is header=TRUE)

#alternatively, replace filename with file.choose() to open interactive window

comm <- read.csv("grassland data/grassland.community.csv", header= TRUE, row.names = 1)

By reading the data in this way, we have set the species names as the column names, and the sample names as the row names. This is important to note - we didn't load these labels in as data - they are the names of the rows/columns. Later this will make it easier for us to link different data sets. Let's check to make sure our rows and columns have reasonable-looking names.

#first let’s find out the class of object

class(comm)

#get the dimensions (rows x columns)

dim(comm)

#find out the row and column names

rownames(comm)

head(colnames(comm))

#take a peek at the data (just the first five rows/columns)

comm[1:5, 1:5]

Each cell contains the percentage cover of a species in a sample. Many multivariate methods are sensitive to the total abundance in a sample, so we should probably convert these absolute abundance estimates to a relative abundance estimate. We can do this with a function from the **vegan** package.

#check total abundance in each sample

apply(comm, 1, sum)

#turn % cover to relative abundance by dividing each value by sample total abundance

comm <- decostand(comm, method = "total")

#check total abundance in each sample

apply(comm, 1, sum)

## Trait data

We also have information on the leaf and root traits of each species. We can load these data in the same way as the community data, but now we will have species in the rows and traits in the columns.

#remember, you could replace filename with file.choose() to open interactive window

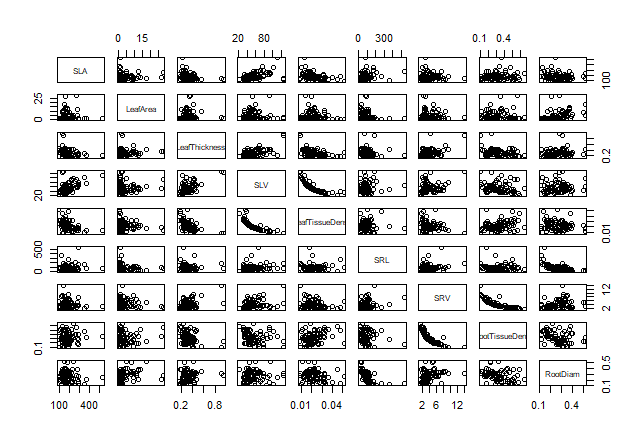
traits <- read.csv("grassland data/species.traits.csv", header = TRUE, row.names = 1)

#take a peek at the data

head(traits)

#plot the data

pairs(traits)

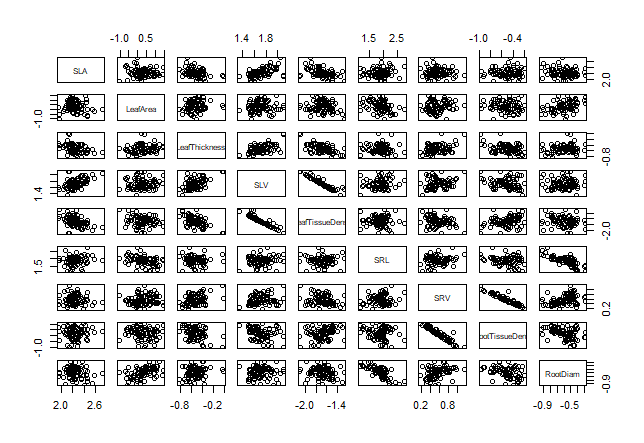


# some variables look skewed so we will log transform all variables

traits <- log10(traits)

# plot the transformed data – does this look better?

pairs(traits)



## Metadata

We have some information about the samples, including the habitat and site they were collected from, and a few basic environmental variables such as slope and moisture regime.

#load metadata file

metadata <- read.csv("grassland\_data/plot.metadata.csv", header = TRUE, row.names = 1)

#check the data

head(metadata)

## Phylogeny

If you have a phylogeny in the commonly used Newick or Nexus format it can be imported into R with the read.tree or read.nexus functions.

#load the phylogeny data and check the class of object

phy <- read.tree("grassland\_data/grassland.phylogeny.newick")

class(phy)

Our phylogeny is a special object of type phylo. It is a type of list object but has different elements such as tip labels and edge lengths. R knows how to summarize and plot a phylo object due to the way it is defined by the **ape** package.

#check what this object looks like

phy

#list the elements of our phylogeny

names(phy)

# what are the first few tip labels?

phy$tip.label[1:5]

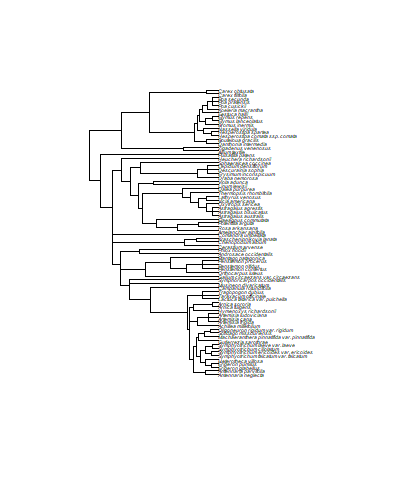
# how many tips does our phylogeny have?

Ntip(phy)

#plot our phylogeny

#change the cex argument to makes the labels smaller or larger

plot(phy, cex = 0.5)



## Cleaning and matching data sets

Our workspace now contains four data sets for community, trait, phylogeny, and metadata.

#list the objects in your workspace

ls()

The data sets we are using today have already been cleaned up so that they contain the same species and the same samples, but often when we are working with our own data, there will be mismatches among different types of data. For example, our community data might only contain a subset of the species in our phylogeny, or there might be some species for which we have trait information but no phylogenetic information. For some analyses, R will assume that species are in the same order in both the community data set and the phylogeny. Sometimes there might be a typo in the labels for a dataset, and we will want to catch those before proceeding.

There are several functions in **picante** that are designed to make sure different data sets match with one another. We should check that our phylogeny and community contain the same species, and that they are in the same order. The match.phylo.comm takes a community object and a phylo object, reports any species that are not present in both data sets, and outputs a version of each object in the same order and containing the same species.

#check for mismatches/missing species

combined <- match.phylo.comm(phy, comm)

#the resulting object is a list with $phy and $comm elements

#we then replace our original data with the sorted/matched data

phy <- combined$phy

comm <- combined$comm

We should do the same matching for our trait data. We can use match.phylo.data to match a data object and a phylo object.

combined <- match.phylo.data(phy, traits)

#the resulting object is a list with $phy and $data elements

#we then replace our original data with the sorted/matched data

phy <- combined$phy

traits <- combined$data

We should also check whether our community data and metadata are in the same order.

all.equal(rownames(comm), rownames(metadata))

#they all match - if they didn't we could sort them to the same order

#this is how we would sort metadata rows to be in the same order as community rows

metadata <- metadata[rownames(comm), ]

We're done! All of our data are now ready for analysis. In each of the sections below we will explore different ways of analysing the community data for plants in Alberta grasslands.

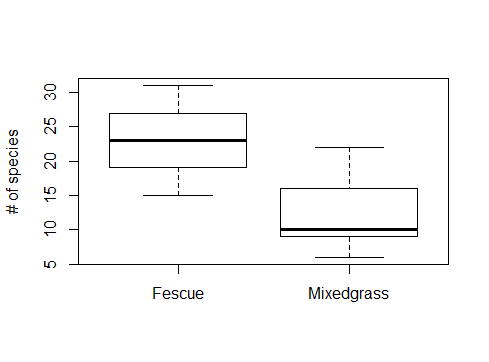
# Visualizing and summarizing biodiversity data

## Community richness and diversity

At a most basic level, we can ask about the overall taxonomic diversity of these grasslands. How many plant species are there? Do habitats differ in species richness?

#compare species richness between fescue and mixed grass habitats

boxplot(specnumber(comm) ~ metadata$habitat, ylab = "# of species")



#use a statistical test to analyse the difference

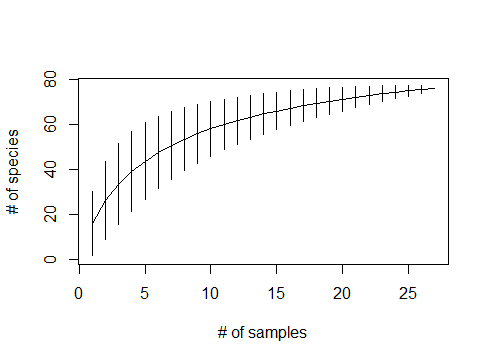
#for simplicity we will use a t-test this time

t.test(specnumber(comm) ~ metadata$habitat)

You might want to know if you did we do a good job of sampling the diversity. We can look at a species accumulation curve to assess this.

#plot a species accumulation curve across samples

plot(specaccum(comm), xlab = "# of samples", ylab = "# of species")



There are further analyses that could be performed to look at taxonomic biodiversity, but for now we will move on to look at community composition as well as phylogenetic and trait diversity.

# Multivariate community analysis

Basic information, such as species richness, may not reveal the full picture of a community. We may be more interested in whether the composition of communities varies across different locations, sites, or treatments. We may also want to incorporate an analysis that examines how habitat type and environmental variables are related to community composition.

In this case we can use multivariate ordination methods to explore community structure in more detail. These methods are available in the **vegan** package, which also includes excellent documentation and tutorials for these methods. The book “Numerical Ecology in R” by Borcard et al. gives a great overview of multivariate analysis methods.

## Hierarchical clustering

We can cluster together plots based on their overall community composition. We will calculate Bray-Curtis dissimilarity among all the samples. This is an abundance-weighted measure of how similar two communities are in terms of their species composition. We will then cluster together communities that are similar using an agglomerative hierarchical clustering algorithm.

#calculate Bray-Curtis distance among samples

comm.bc.dist <- vegdist(comm, method = "bray")

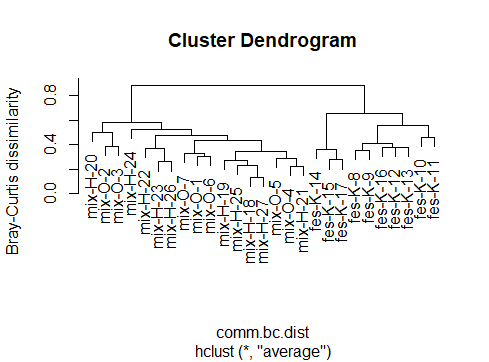
#cluster communities using average-linkage algorithm

comm.bc.clust <- hclust(comm.bc.dist, method = "average")

#we will later use the results of this cluster analysis to add to our ordination plot

#for now use the analysis to plot a cluster diagram

plot(comm.bc.clust, ylab = "Bray-Curtis dissimilarity")



How would you interpret this cluster diagram? It looks like mixed grass and fescue habitats contain different plant community types - the two main clusters separate fescue samples from all other samples.

## Ordination

There are numerous ordination methods available in R. It is possible to do a constrained ordination such as constrained correspondence analysis (CCA) or redundancy analysis (RDA), where trait or environmental data are incorporated directly into the ordination. These methods are implemented in the functions cca and rda in **vegan**.

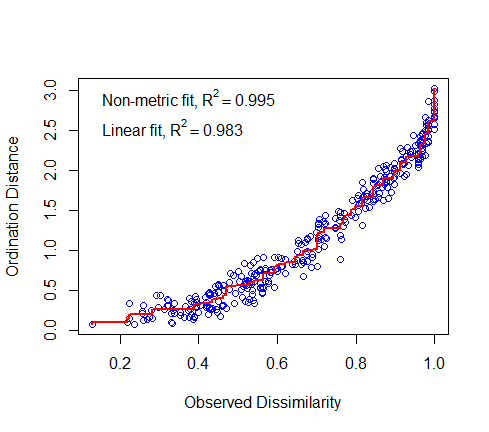
There is a large volume of literature on selecting the most appropriate type of ordination for your data so this should be considered before attempting your own analysis. For now, let's use non-metric multidimensional scaling to visualize the multivariate structure of these communities.

#The metaMDS function automatically transforms data, runs the NMDS and checks solution robustness

comm.bc.mds <- metaMDS(comm, dist = "bray")

#A stress plot can then be used to assess goodness of ordination fit

stressplot(comm.bc.mds)



We can plot the ordination results in a variety of different ways.

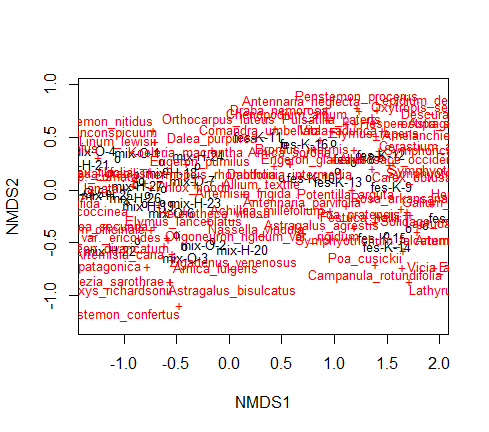
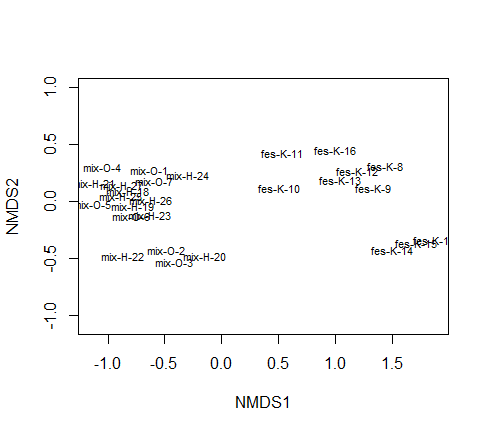
#plot site scores as text

ordiplot(comm.bc.mds, display = "sites", type = "text")

#use automated plotting of results to try and eliminate overlapping labels

#this may take a while to run

ordipointlabel(comm.bc.mds)



#the previous plot isn’t easy to understand but ordination plots are highly customizable

#set up the plotting area but don't plot anything yet

mds.fig <- ordiplot(comm.bc.mds, type = "none")

#plot just the samples

#colour by habitat

#pch=19 means plot a circle

points(mds.fig, "sites", pch = 19, col = "green", select = metadata$habitat == "Fescue")

points(mds.fig, "sites", pch = 19, col = "blue", select = metadata$habitat == "Mixedgrass")

# add confidence ellipses around habitat types

ordiellipse(comm.bc.mds, metadata$habitat, conf = 0.95, label = TRUE)

# overlay the cluster results we calculated earlier

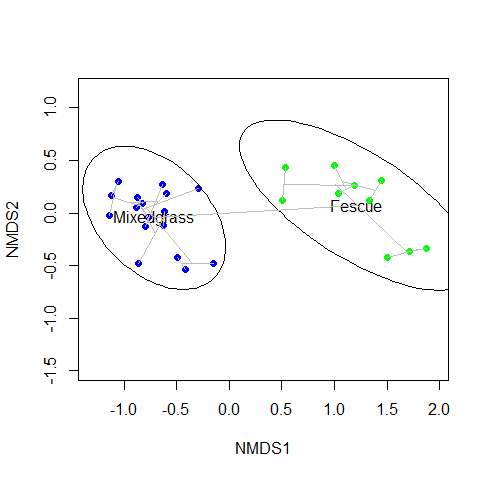
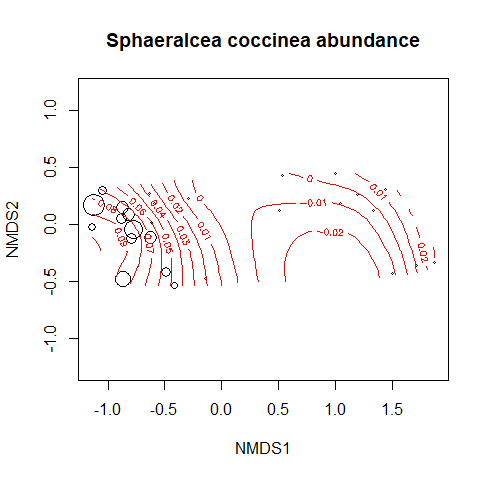
ordicluster(comm.bc.mds, comm.bc.clust, col = "gray")

We can also visualize the abundance of species. The ordisurf function fits a smooth surface to estimates of species abundance. Understanding this function is too complex for the scope of this workshop but I have included it here for completeness. I recommend reading the following blogpost for some insight into what the ordisurf function is doing; https://fromthebottomoftheheap.net/2011/06/10/what-is-ordisurf-doing/

#plot Sphaeralcea abundance

#cex increases the size of bubbles

ordisurf(comm.bc.mds, comm[, "Sphaeralcea\_coccinea"], bubble = TRUE, main = "Sphaeralcea coccinea abundance", cex = 3)

## Adding environmental and trait data to ordinations

How are environmental variables correlated with the ordination axes?

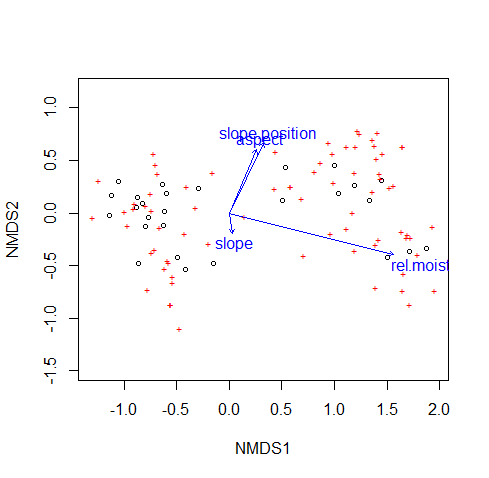
#begin by plotting the ordination

ordiplot(comm.bc.mds)

#calculate and plot environmental variable correlations with the axes

#use the subset of metadata that are environmental data

plot(envfit(comm.bc.mds, metadata[, 3:6]))



# Trait evolution

## Phylogenetic signal

The idea of phylogenetic niche conservatism (the ecological similarity of closely related species) has attracted a lot of attention recently, for example in the widely used framework of inferring community assembly processes based on knowledge of community phylogenetic structure plus the phylogenetic conservatism of traits. (Webb et al. 2002).

Phylogenetic signal is a quantitative measure of the degree to which phylogeny predicts the ecological similarity of species. The K statistic is a measure of phylogenetic signal that compares the observed signal in a trait to the signal under a Brownian motion model of trait evolution on a phylogeny (Blomberg et al. 2003). K values of 1 correspond to a Brownian motion process, which implies some degree of phylogenetic signal or conservatism. K values closer to zero correspond to a random or convergent pattern of evolution, while K values greater than 1 indicate strong phylogenetic signal and conservatism of traits. The statistical significance of phylogenetic signal can be evaluated by comparing observed patterns of the variance of independent contrasts of the trait to a null model of shuffling taxa labels across the tips of the phylogeny. These tests are implemented in the Kcalc, phylosignal, and multiPhylosignal functions.

Let's measure phylogenetic signal in these data.

#one way to do it - apply the Kcalc function to each column of the data.frame

apply(traits, 2, Kcalc, phy)

#another way to do it - with significance testing

#we have to convert the tree to be dichotomous before calculating P-values

multiPhylosignal(traits, multi2di(phy))

In the output, K is the K statistic (magnitude of signal vs. Brownian motion), and PIC.variance.P is the P-value of the test for non-random signal. Most variables show more phylogenetic signal than expected by chance.

## Visualizing trait evolution

We can visualize trait values on the phylogeny by plotting a different colour or size of symbol for each trait value. Let's visualize leaf area, the trait with the strongest phylogenetic signal. The cex argument to the tiplabels function adjusts the size of the trait symbols - some tinkering around with the scaling of the symbol sizes is required depending on the trait.

#plot phylogeny facing upwards

#show node labels but not tip labels

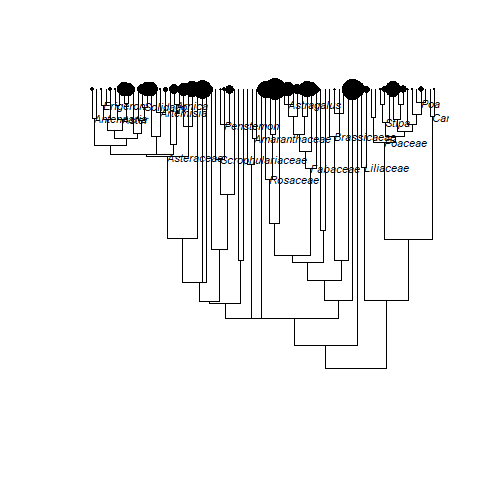
#use cex to shrink/enlarge labels

plot(phy, direction = "up", show.tip.label = FALSE, show.node.label = TRUE, cex = 0.7)

#plot leaf area on the phylogeny

#cex argument scales symbol size by trait value

tiplabels(pch = 19, col = "black", cex = 3 \* (traits[, "LeafArea"]/max(traits[, "LeafArea"])))



## Phylogenetic analysis of trait relationships

Phylogenetic signal means that closely related species have similar traits. This violates the assumption of independence of data points that is inherent in many methods including correlation and regression (Felsenstein 1985). We can account for non-independence due to phylogenetic signal using methods including phylogenetically independent contrasts and phylogenetic generalised least squares (pGLS). Generalised least squares methods work just like an ANOVA or linear model - we can test for relationships between categorical or continuous values, optionally taking phylogenetic relatedness into account.

Let's test for a relationship between specific root length (SRL) and root tissue density, taking phylogenetic relationships among species into account.

#GLS of root tissue density as a function of SRL

#this is the non-phylogenetic model

root.gls <- gls(RootTissueDens ~ SRL, data = traits)

anova(root.gls)

#then run a phylogenetic GLS – this adds the effect of phylogeny to the model

root.pgls <- gls(RootTissueDens ~ SRL, correlation = corBrownian(value = 1, phy),

data = traits)

anova(root.pgls)

#plot the relationship

plot(RootTissueDens ~ SRL, data = traits, xlab = "SRL (specific root length)",

ylab = "Root tissue density")

#add model fit lines

#use different colours for the phylogenetic and non-phylogenetic models

#coef is the model fit coefficients

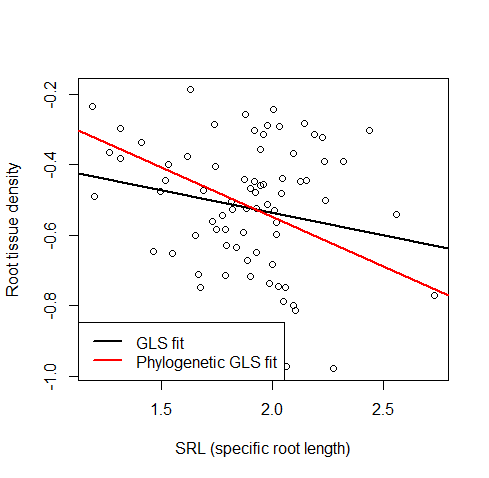
#lwd increases line width

abline(coef(root.gls), lwd = 2, col = "black")

abline(coef(root.pgls), lwd = 2, col = "red")

legend("bottomleft", legend = c("GLS fit", "Phylogenetic GLS fit"), lwd = 2,

col = c("black", "red"))



There is a weak relationship between SRL and root tissue density. The relationship is not significant if we do not take phylogenetic relatedness into account. We see a stronger and significant relationship between SRL and root tissue density after taking phylogenetic relatedness into account.

# Phylogenetic and trait diversity

## Phylogenetic diversity

One of the earliest measures of phylogenetic relatedness in ecological communities was the phylogenetic diversity (PD) index proposed by Faith. Faith's PD is defined as the total branch length spanned by the tree including all species in a local community, optionally including the root node of the phylogeny. The pd function returns two values for each community, Faith's PD and species richness (SR).

#calculate Faith's PD

comm.pd <- pd(comm, phy)

head(comm.pd)

#plot Faith's PD by habitat

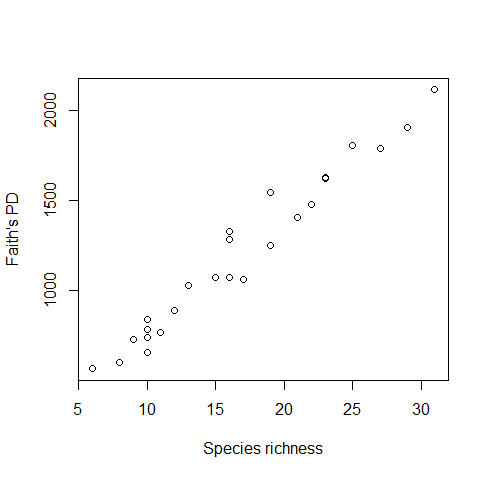
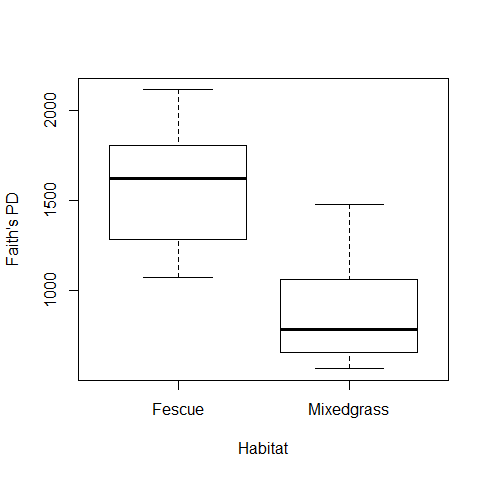
boxplot(comm.pd$PD ~ metadata$habitat, xlab = "Habitat", ylab = "Faith's PD")

#test for PD differences among habitats

t.test(comm.pd$PD ~ metadata$habitat)

#compare PD and species richness

plot(comm.pd$PD ~ comm.pd$SR, xlab = "Species richness", ylab = "Faith's PD")



Faith's PD is lower in mixed grass habitats than in fescue habitats. But Faith's PD is highly correlated with species richness, and we already know that there are fewer species in mixed grass habitats, so we need some way to compare phylogenetic diversity that takes this fact into account.

## Community phylogenetic structure

Another way of thinking about the phylogenetic relatedness of species in a community is to ask 'how closely related are the average pair of species or individuals in a community', and relate the patterns we observe to what we'd expect under various null models of evolution and community assembly. These types of questions are addressed by the measures of community phylogenetic structure such as MPD and MNTD, described by Webb et al. and implemented in Phylocom. Phylocom is a piece of software for the analysis of phylogenetic community structure and character evolution. For now, we’ll stick with R!

The function mpd will calculate the mean pairwise distance between all species or individuals in each community. Similarly, the mntd function calculates the mean nearest taxon distance, the average distance separating each species or individual in the community from its closest heterospecific relative. The mpd and mntd functions differ slightly from the pd function in the last section because they take a distance matrix as input rather than a phylogeny object. A phylo object can be converted to an interspecific phylogenetic distance matrix using the cophenetic function. Since the mpd and mntd functions can use any distance matrix as input, we can easily calculate trait diversity measures by substituting a trait distance matrix for the phylogenetic distance matrix. We'll return to this idea of analysing trait diversity shortly.

Measures of 'standardized effect size' of phylogenetic community structure can be calculated for MPD and MNTD by comparing observed phylogenetic relatedness to the pattern expected under some null model of phylogeny or community randomization. Standardized effect sizes describe the difference between average phylogenetic distances in the observed communities versus null communities generated with some randomization method. They are standardized by the standard deviation of phylogenetic distances in the null data.

Several different null models can be used to generate the null communities. These include randomizations of the tip labels of the phylogeny, and various community randomizations that can hold community species richness and/or species occurrence frequency constant. If the community data represent abundance measures then abundance data can be taken into account. Doing so changes the interpretation of these metrics from the average distance among two randomly chosen species from a community, to the average distance among two randomly chosen individuals in a community. For this workshop we will ignore abundance information and use a simple null model of randomly drawing species while keeping sample species richness constant. Let's calculate some of these measures of community phylogenetic structure for our example data set.

# convert phylogeny to a distance matrix

phy.dist <- cophenetic(phy)

# calculate ses.mpd – standardised effect size for the mean pairwise distance

comm.sesmpd <- ses.mpd(comm, phy.dist, null.model = "richness",

abundance.weighted = FALSE, runs = 999)

head(comm.sesmpd)

# calculate ses.mntd – standardised effect size for the mean nearest taxon distance

comm.sesmntd <- ses.mntd(comm, phy.dist, null.model = "richness",

abundance.weighted = FALSE, runs = 999)

head(comm.sesmntd)

The output includes the following columns:

* ntaxa - Number of taxa in community
* mpd.obs - Observed mpd in community
* mpd.rand.mean - Mean mpd in null communities
* mpd.rand.sd - Standard deviation of mpd in null communities
* mpd.obs.rank - Rank of observed mpd vs. null communities
* mpd.obs.z - Standardized effect size of mpd vs. null communities (equivalent to -NRI)
* mpd.obs.p - P-value (quantile) of observed mpd vs. null communities (= mpd.obs.rank / runs + 1)
* runs - Number of randomizations

Positive SES values (mpd.obs.z > 0) and high quantiles (mpd.obs.p > 0.95) indicate phylogenetic evenness, while negative SES values and low quantiles (mpd.obs.p < 0.05) indicate phylogenetic clustering, relative to the null model. MPD is generally thought to be more sensitive to tree-wide patterns of phylogenetic clustering and eveness, while MNTD is more sensitive to patterns of evenness and clustering closer to the tips of the phylogeny.

#we can now compare ses.mpd between habitats statistically

t.test(comm.sesmpd$mpd.obs.z ~ metadata$habitat)

#let’s plot the data to visualise the comparison

boxplot(comm.sesmpd$mpd.obs.z ~ metadata$habitat, xlab = "Habitat", ylab = "SES(MPD)")

#now do the same to compare ses.mntd between habitats

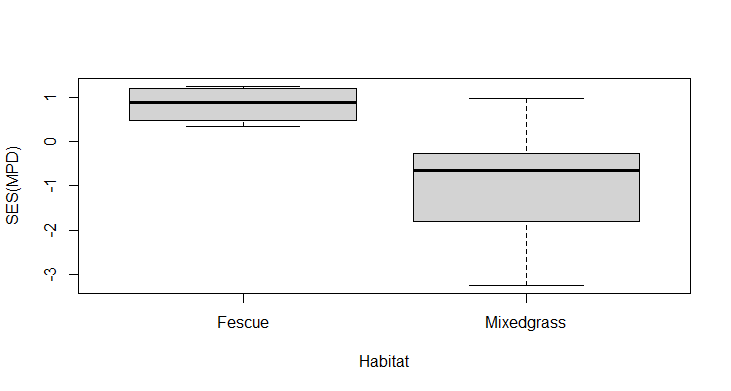
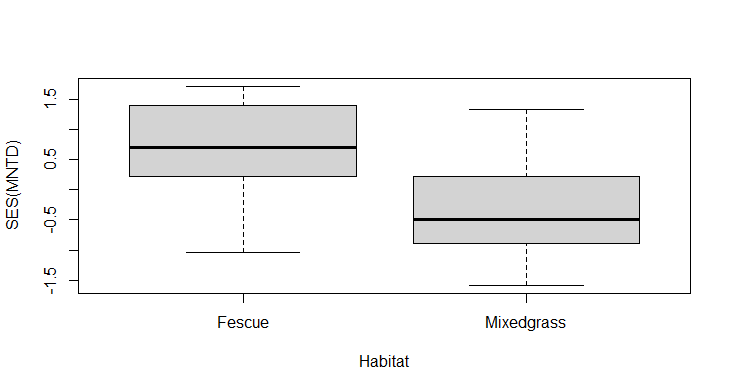
#start with the t-test

t.test(comm.sesmntd$mntd.obs.z ~ metadata$habitat)

#then plot the data

boxplot(comm.sesmntd$mntd.obs.z ~ metadata$habitat, xlab = "Habitat", ylab = "SES(MNTD)")

abline(h = 0, col = "gray")

It looks like plant communities from fescue habitats are phylogenetically even (more distantly related than expected by chance, \(SES > 0\)), and communities from mixed grass habitats are phylogenetically clustered (more closely related than expected by chance, \(SES < 0\)).

Let's look at the distribution of species from samples in these different habitats on the phylogeny.

#on the phylogeny plot species present in a fescue community

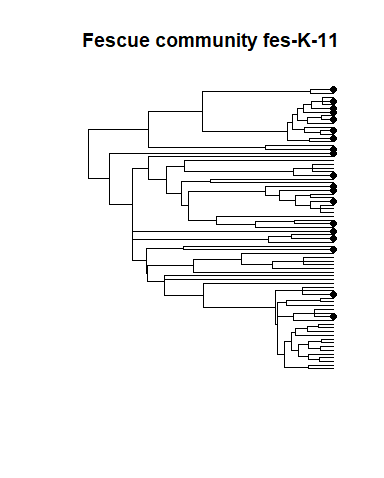
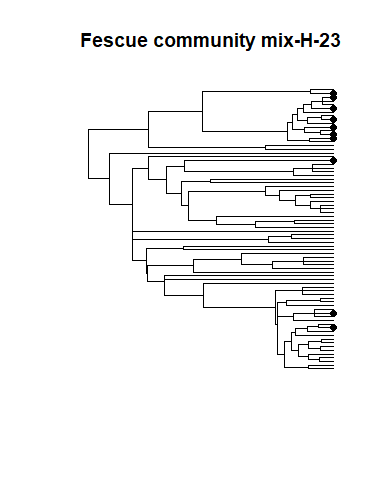
plot(phy, show.tip.label = FALSE, main = "Fescue community fes-K-11")

tiplabels(tip = which(phy$tip.label %in% colnames(comm)[comm["fes-K-11", ] > 0]), pch = 19)

#now plot species present in a mixed grass community on to the phylogeny

plot(phy, show.tip.label = FALSE, main = "Fescue community mix-H-23")

tiplabels(tip = which(phy$tip.label %in% colnames(comm)[comm["mix-H-23", ] > 0]), pch = 19)

What do these plots say about the phylogenetic relatedness of the plant species in each community? It appears that the fescue community 'fes-K-11' contains species that are phylogenetically even. In contrast the mixed grass community 'mix-H-23' contains species that are phylogenetically clumped.

## Trait diversity

We can calculate measures of trait diversity within communities in a manner analogous to the methods we used to calculate phylogenetic diversity. Let's calculate the standardized effect size of functional trait diversity by measuring trait dissimilarity among co-occurring species and comparing observed trait diversity to a null model.

#calculate trait distance - Euclidean distance among scaled trait values

# we want the full distance matrix

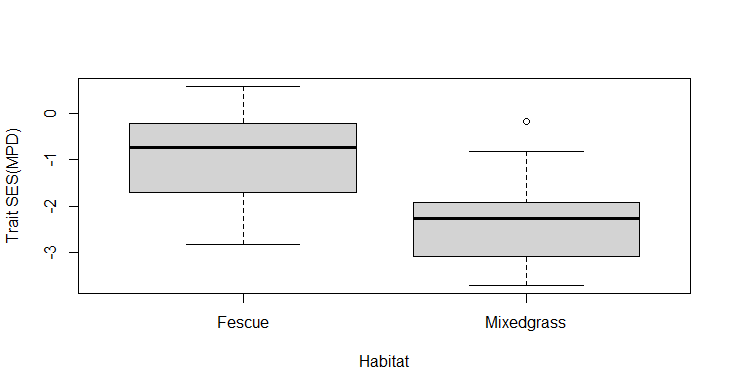
trait.dist <- as.matrix(dist(scale(traits), method = "euclidean"))

#calculate trait ses.mpd

comm.sesmpd.traits <- ses.mpd(comm, trait.dist, null.model = "richness", abundance.weighted = FALSE, runs = 999)

#compare trait ses.mpd between habitats

boxplot(comm.sesmpd.traits$mpd.obs.z ~ metadata$habitat, xlab = "Habitat", ylab = "Trait SES(MPD)")



In contrast to the pattern we saw for phylogenetic diversity, trait diversity is lower than expected in both habitats (\(SES\_{MPD} < 0\)), indicating that co-occurring plants have similar leaf and root traits, and this pattern of trait clustering is stronger in mixedgrass habitats.

The treedive function in **vegan** calculates a measure of functional trait diversity that is similar to Faith's PD.

## Phylogenetic beta-diversity

We can measure patterns of phylogenetic relatedness among communities in a manner similar to the within-community phylogenetic diversity measures described above. The unifrac and phylosor functions measure the among-community equivalent of Faith's PD, the total unique/shared branch length between communities. The comdist and comdistnt functions measure the among-community equivalent of MPD and MNTD, the mean pairwise distance or mean nearest taxon distance between pairs of species drawn from two distinct communities.

Let's compare a few different ways of measuring dissimilarity among communities. We've already calculated the Bray-Curtis distance among communities based on shared species (comm.bc.dist). Since the Bray-Curtis distance incorporates species abundances, we should use abundance information when calculating phylogenetic and trait diversity as well.

#calculate phylogenetic MNTD beta diversity

comm.mntd.dist <- comdistnt(comm, phy.dist, abundance.weighted = TRUE)

#calculate functional trait MNTD beta diversity

comm.mntd.traits.dist <- comdistnt(comm, trait.dist, abundance.weighted = TRUE)

#calculate Mantel correlation for taxonomic Bray-Curtis vs. phylogenetic MNTD diversity

mantel(comm.bc.dist, comm.mntd.dist)

#calculate Mantel correlation for taxonomic Bray-Curtis vs. trait MNTD diversity

mantel(comm.bc.dist, comm.mntd.traits.dist)

## Phylogeny/trait-based ordinations

Since we can calculate phylogeny- and trait-based measures of dissimilarity among samples, we can also perform an ordination of samples based on these metrics. Let's compare phylogeny- and trait-based ordinations with the species-based ordination we performed earlier.

#NMDS ordination of phylogenetic distances

#use monoMDS since we only have among-sample distances

comm.mntd.mds <- monoMDS(comm.mntd.dist)

#set up the plotting area but don't plot anything yet

mds.fig <- ordiplot(comm.mntd.mds, type = "none")

#plot just the samples, colour by habitat

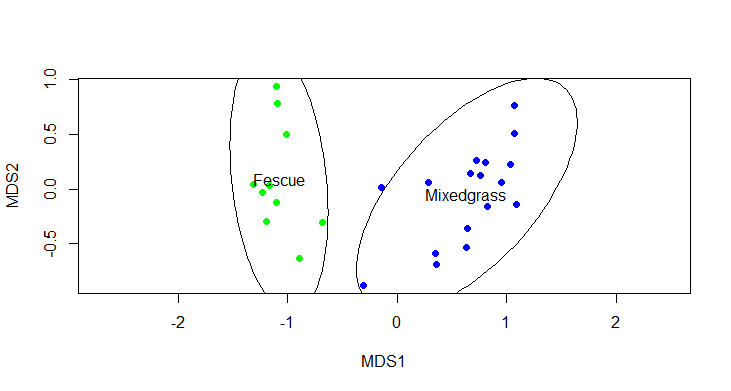
#pch=19 means plot a circle

points(mds.fig, "sites", pch = 19, col = "green", select = metadata$habitat == "Fescue")

points(mds.fig, "sites", pch = 19, col = "blue", select = metadata$habitat == "Mixedgrass")

#add confidence ellipses around habitat types

ordiellipse(comm.mntd.mds, metadata$habitat, conf = 0.95, label = TRUE)



#NMDS ordination of trait distances

#use monoMDS since we only have among-sample distances

comm.mntd.traits.mds <- monoMDS(comm.mntd.traits.dist)

#set up the plotting area but don't plot anything yet

mds.fig <- ordiplot(comm.mntd.traits.mds, type = "none")

#plot just the samples, colour by habitat

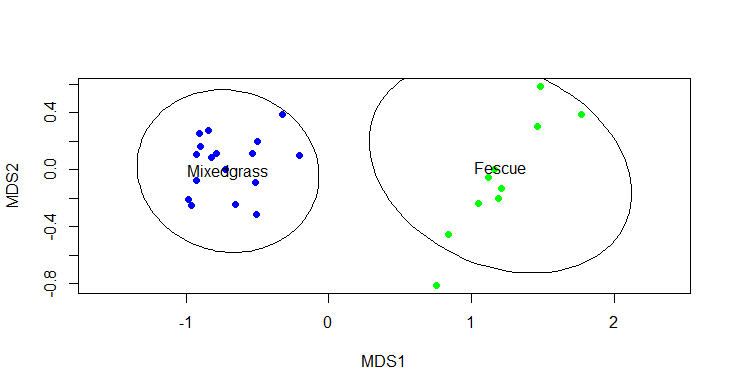
#pch=19 means plot a circle

points(mds.fig, "sites", pch = 19, col = "green", select = metadata$habitat == "Fescue")

points(mds.fig, "sites", pch = 19, col = "blue", select = metadata$habitat == "Mixedgrass")

#add confidence ellipses around habitat types

ordiellipse(comm.mntd.traits.mds, metadata$habitat, conf = 0.95, label = TRUE)



It looks like fescue and mixed grass habitats are quite distinct regardless of how we quantify their biodiversity - they contain different species, phylogenetically distinct taxa, and the traits of species in the two habitats are distinct.

## Testing for multivariate differences among groups

We can quantify the relationship between dissimilarity measures and different explanatory variables using the permutational MANOVA (a.k.a. ANOVA) framework in the adonis function in **vegan**. This method allows ANOVA-like tests of the variance in beta diversity explained by categorical or continuous variables.

Let's statistically quantify the degree to which habitat can explain taxonomic, phylogenetic, and trait dissimilarity among grasslands.

#taxonomic (Bray-Curtis) dissimilarity explained

adonis(comm.bc.dist ~ habitat, data = metadata)

#phylogenetic dissimilarity explained

adonis(comm.mntd.dist ~ habitat, data = metadata)

#trait dissimilarity explained

adonis(comm.mntd.traits.dist ~ habitat, data = metadata)

These results support the pattern we can see visually in the ordination diagrams. These habitats are distinct in terms of their taxonomic, phylogenetic, and functional trait diversity.