Assignment: Spatial Diversity

Matt Gibson; Z620: Quantitative Biodiversity, Indiana University

08 February, 2017

OVERVIEW

This assignment will emphasize primary concepts and patterns associated with spatial diversity, while using R as a Geographic Information Systems (GIS) environment. Complete the assignment by refering to examples in the handout.

After completing this assignment you will be able to:
1. Begin using R as a geographical information systems (GIS) environment.
2. Identify primary concepts and patterns of spatial diversity.
3. Examine effects of geographic distance on community similarity.
4. Generate simulated spatial data.

Directions:

1. Change "Student Name" on line 3 (above) with your name.
2. Complete as much of the assignment as possible during class; what you do not complete in class will need to be done on your own outside of class.
3. Use the handout as a guide; it contains a more complete description of data sets along with the proper scripting needed to carry out the assignment.
4. Be sure to **answer the questions** in this assignment document. Space for your answer is provided in this document and indicated by the ">" character. If you need a second paragraph be sure to start the first line with ">".
5. Before you leave the classroom, **push** this file to your GitHub repo.
6. When you are done wit the assignment, **Knit** the text and code into an html file.
7. After Knitting, please submit the completed assignment by creating a **pull request** via GitHub. Your pull request should include this file *spatial_assignment.Rmd* and the html output of `Knitr` (*spatial_assignment.html*).

1) R SETUP

In the R code chunk below, provide the code to:

1. Clear your R environment
2. Print your current working directory,
3. Set your working directory to your "*/Week4-Spatial*" folder, and

2) LOADING R PACKAGES

In the R code chunk below, do the following:

1. Install and/or load the following packages: `vegan`, `sp`, `gstat`, `raster`, `RgoogleMaps`, `maptools`, `rgdal`, `simba`, `gplots`, `rgeos`

```r
rm(list=ls())
getwd()
setwd("c:/Users/matth/Documents/bin/QB2017_Gibson/Week4-Spatial")


package.list <- c('vegan', 'sp', 'gstat','raster', 'RgoogleMaps', 'maptools', 'rgdal',
                  'simba', 'gplots', 'rgeos', 'rgdal')
for (p in package.list){
  library(p, character.only = T)
  #install.packages(p)
}
```

Question 1: What are the packages `simba`, `sp`, and `rgdal` used for?

> ***Answer 1***: Simba is for calculating similarity measures, sp is for managing spatial data, rgdal is a geospatial abstraction library.

3) LOADING DATA

In the R code chunk below, use the example in the handout to do the following:

1. Load the Site-by-Species matrix for the Indiana ponds datasets: BrownCoData/SiteBySpecies.csv
2. Load the Environmental data matrix: BrownCoData/20130801_PondDataMod.csv
3. Assign the operational taxonomic units (OTUs) to a variable 'otu.names'
4. Remove the first column (i.e., site names) from the OTU matrix.

```r
Ponds <- read.table("BrownCoData/20130801_PondDataMod.csv", sep=",", header=T)
OTUs <- read.csv("BrownCoData/SiteBySpecies.csv", sep=",", header=T)
otu.names <- names(OTUs)
OTUs <- as.data.frame(OTUs[-1])
d <- dim(OTUs)
s <- seq(2, d[1])

S.obs <- function(x = ""){
  rowSums(x > 0) * 1
}
richness <- S.obs(OTUs[1,])

for (i in s) {
  richness <- cbind(richness, S.obs(OTUs[i, ]))
}

max(richness)
```

Question 2a: How many sites and OTUs are in the SiteBySpecies matrix?

> ***Answer 2a***: There are 51 sites and 16383 OTUs

Question 2b: What is the greatest species richness found among sites?

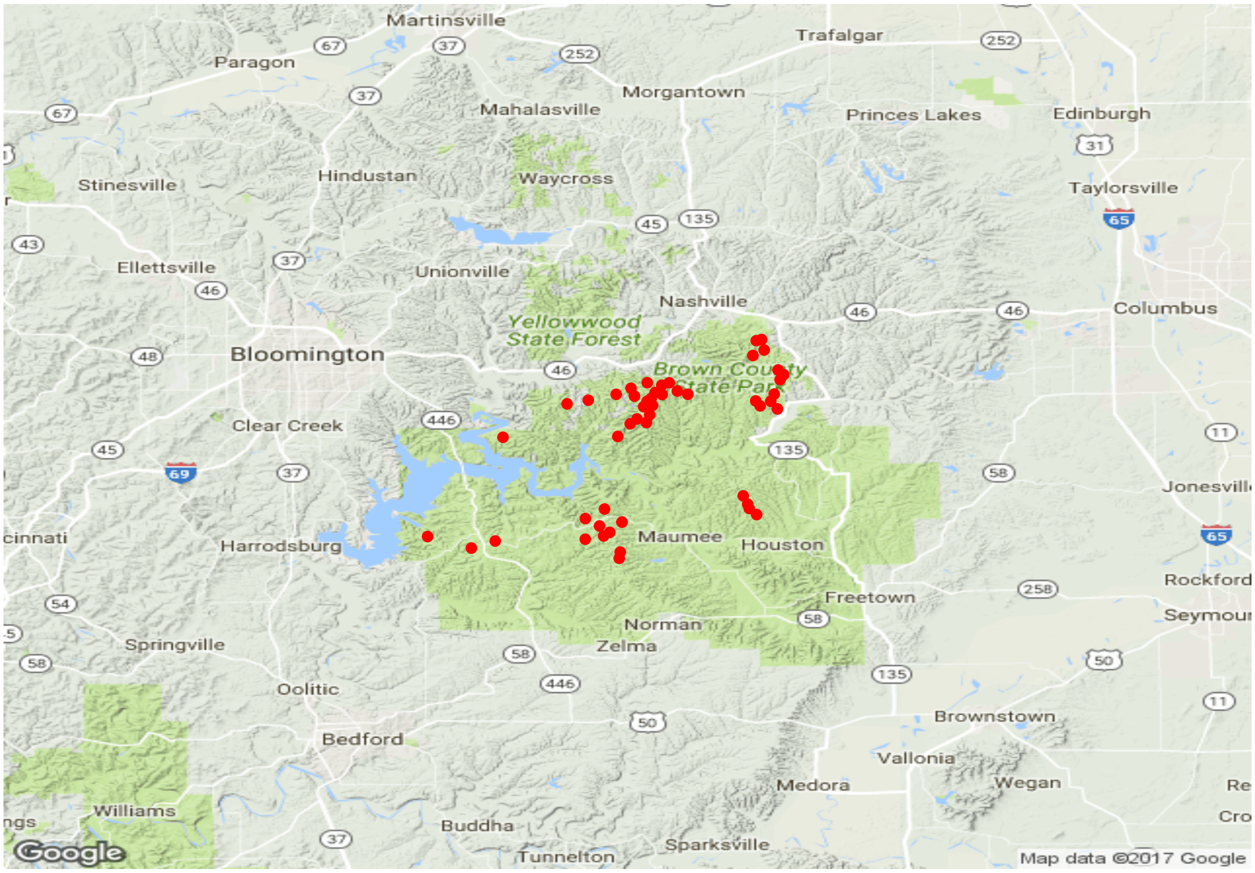
> ***Answer 2b***: 3659 at site 47

4) GENERATE MAPS

In the R code chunk below, do the following:

1. Using the example in the handout, visualize the spatial distribution of our samples with a basic map in RStudio using the `GetMap` function in the package `RgoogleMaps`. This map will be centered on Brown County, Indiana (39.1 latitude, -86.3 longitude).

```r
lats <- as.numeric(Ponds[, 3])
lons <- as.numeric(Ponds[, 4])

newmap <- GetMap(center = c(39.1, -86.3), zoom = 10,
                 destfile = "PondsMap.png", maptype = "terrain")

PlotOnStaticMap(newmap, zoom = 10, cex = 2, col = 'blue')
PlotOnStaticMap(newmap, lats, lons, cex = 1, pch = 20, col = 'red', add = T)
```



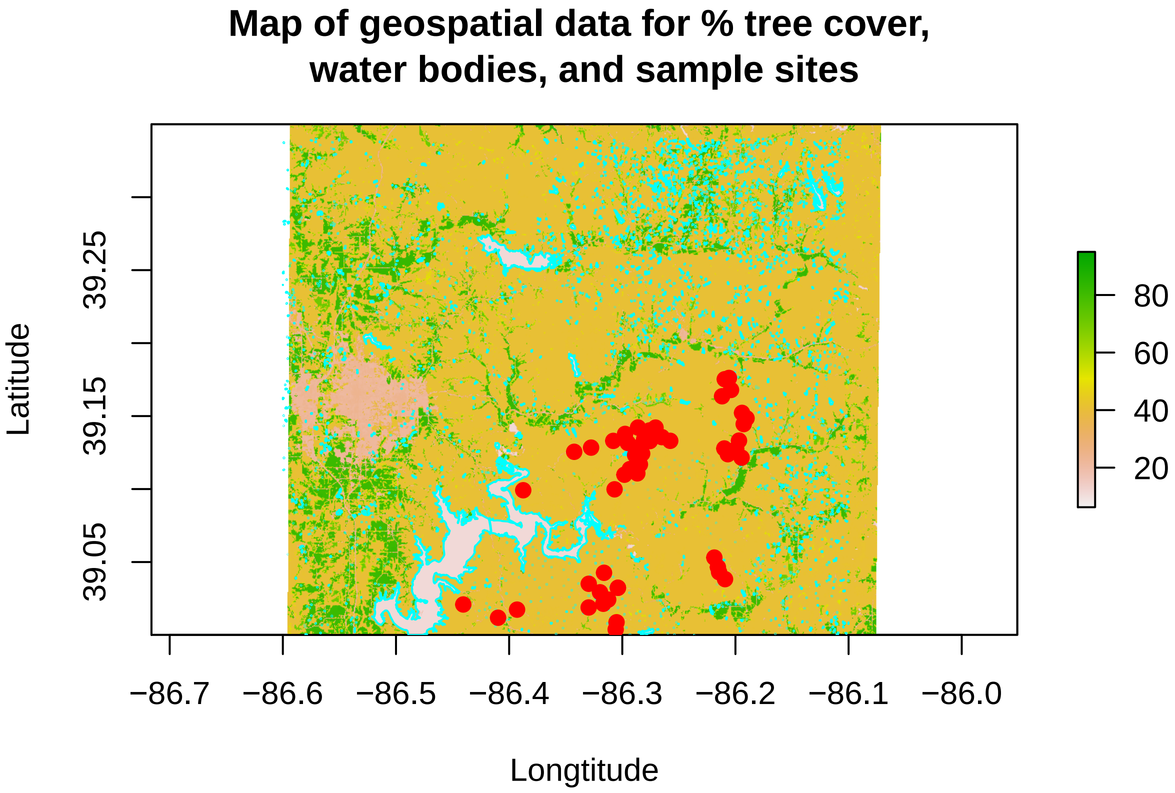
Question 3: Briefly describe the geographical layout of our sites.

> ***Answer 3***: The sites are all located within Brown County State Park, with most of the sites aggregated towards the North and Northwest side of the park. The sites are not distributed very evenly.

In the R code chunk below, do the following:

1. Using the example in the handout, build a map by combining lat-long data from our ponds with land cover data and data on the locations and shapes of surrounding water bodies.

```
# 1. Import TreeCover.tif as a raster file.
Tree.Cover <- raster("TreeCover/TreeCover.tif")
# 2. Plot the % tree cover data
plot(Tree.Cover, xlab = "Longtitude", ylab = 'Latitude', main = 'Map of geospatial data for % tree cove
# 3. Import water bodies as a shapefile.
Water.Bodies <- readShapeSpatial("water/water.shp")
# 4. Plot the water bodies around our study area, i.e., Monroe County.
plot(Water.Bodies, border='cyan', axes = T, add = T)
# 5. Convert lat-long data for ponds to georeferenced points.
Refuge.Ponds <- SpatialPoints(cbind(lons, lats))
# 6. Plot the refuge pond locations
plot(Refuge.Ponds, line='r', col='red', pch=20, cex=1.5, add=T)
```



Question 4a: What are datums and projections?

> ***Answer 4a***: A datum is a model of Earth's shape (of which there are many) and a projection is the way which we take coordinates on a sphere and "project" them onto the 2D map.

5) UNDERSTANDING SPATIAL AUTOCORRELATION

Question 5: In your own words, explain the concept of spatial autocorrelation.

> ***Answer 5***: Spatial autocorrelation is the degree to which sites that are close to one another in space have similar trait values. If variables group by space, then we say there is positive autocorrelation. If varaibles are not grouped by space and are overdispersed, we call this negative autocorrelation.

6) EXAMINING DISTANCE-DECAY

Question 6: In your own words, explain what a distance decay pattern is and what it reveals.

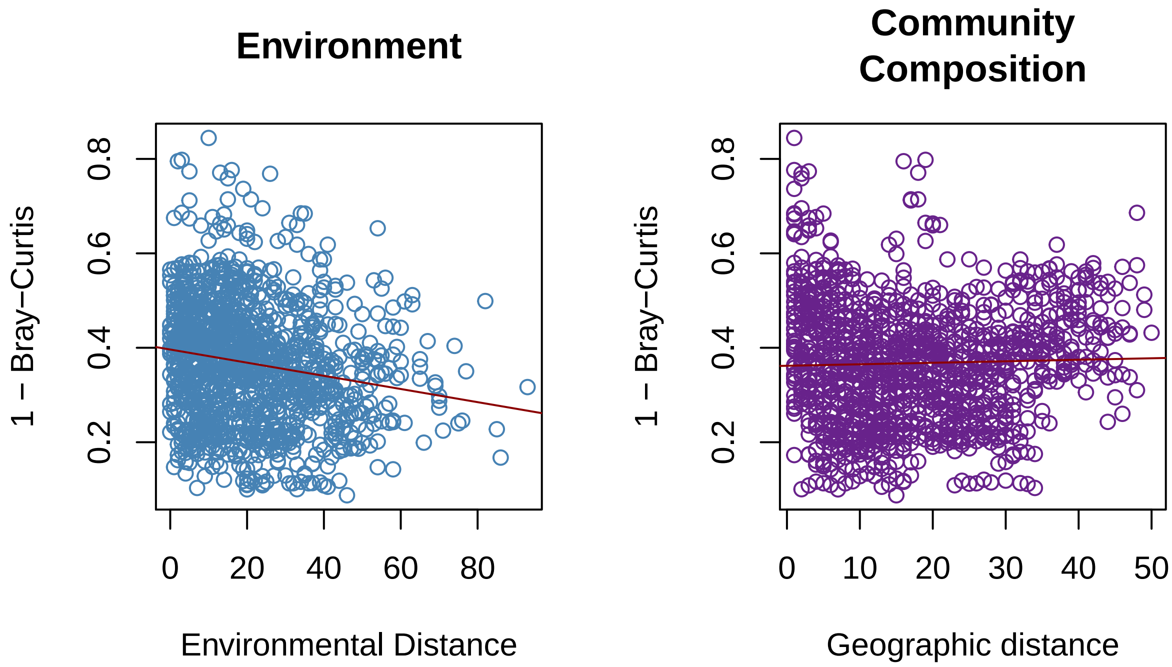
> ***Answer 6***: Distance decay patterns (or graphs) describe how species composition similarity (or dissimilarity) between sites is related to the geographic (or environmental) distance between sites. If this pattern shows a negative relationship, that would be evidence for spatial autocorrelation (sites closer together are more similar to one another). When we use environmental distance instead of geographic distance, a negative relationship would suggest that sites with similar environments have similar species composition.

In the R code chunk below, do the following:

1. Generate the distance decay relationship for bacterial communities of our refuge ponds and for some of the environmental variables that were measured. Note: You will need to use some of the data transformations within the *semivariogram* section of the handout.

```
# 1) Calculate Bray-Curtis similarity between plots using the `vegdist()` function
comm.dist <- 1- vegdist(OTUs)
# 2) Assign UTM lattitude and longitude data to 'lats' and 'lons' variables
xy <- data.frame(env = Ponds$TDS, pond.name = Ponds$Sample_ID, lats = Ponds$lat, lons = Ponds$long)
coordinates(xy) <- ~lats+lons
proj4string(xy) <- CRS("+proj=longlat +datum=NAD83")
UTM <- spTransform(xy, CRS("+proj=utm +zone=51 +ellps=WGS84"))
UTM <- as.data.frame(UTM)
xy$lats_utm <- UTM[,2]
xy$lons_utm <- UTM[,3]
lats <- as.numeric(xy$lats_utm)
lons <- as.numeric(xy$lons_utm)
# 3) Calculate geographic distance between plots and assign to the variable 'coord.dist'
coord.dist <- dist(as.matrix(lats, lons))
# 4) Transform environmental data to numeric type, and assign to variable 'x1'
x1 <- as.numeric(Ponds$"SpC")
# 5) Using the `vegdist()` function in `simba`, calculate the Euclidean distance between the plots for
env.dist <- vegdist(x1, "euclidean")
# 6) Transform all distance matrices into database format using the `liste()` function in `simba`:
comm.dist.ls <- liste(comm.dist, entry="comm")
env.dist.ls <- liste(env.dist, entry = "env")
coord.dist.ls <- liste(coord.dist, entry="dist")
# 7) Create a data frame containing similarity of the environment and similarity of community.
df <- data.frame(coord.dist.ls, env.dist.ls[,3], comm.dist.ls[,3])

# 8) Attach the columns labels 'env' and 'struc' to the dataframe you just made.
names(df)[4:5] <- c("env", "struc")
attach(df)
# 9) After setting the plot parameters, plot the distance-decay relationships, with regression lines in
par(mfrow=c(1, 2), pty="s")
```

```r
plot(env, struc, xlab="Environmental Distance", ylab="1 - Bray-Curtis",
     main = "Environment", col='SteelBlue')

OLS <- lm(struc ~ env)
OLS
abline(OLS, col="red4")

plot(dist, struc, xlab="Geographic distance", ylab="1 - Bray-Curtis",
     main = "Community\nComposition", col="darkorchid4")

OLS <- lm(struc ~ dist)
OLS
abline(OLS, col="red4")
```



```r
# 10) Use `simba` to calculates the difference in slope or intercept of two regression lines
diffslope(env, struc, dist, struc)
```

Question 7: What can you conclude about community similarity with regards to environmental distance and geographic distance?

> ***Answer 7***: It seems that Bray Curtis similarity is more influenced by environment than by geographic distance. Higher environmental distance (i.e. more different environments) leads to reduced Bray Curtis similarity. Based on permutation tests, the slopes of the environmental curve and the geographic distance curve are significantly different (P = 0.001) leading me to conclude that, in this dataset, environment may be more important in shaping species distributions.

7) EXAMINING SPECIES SPATIAL ABUNDANCE DISTRIBUTIONS

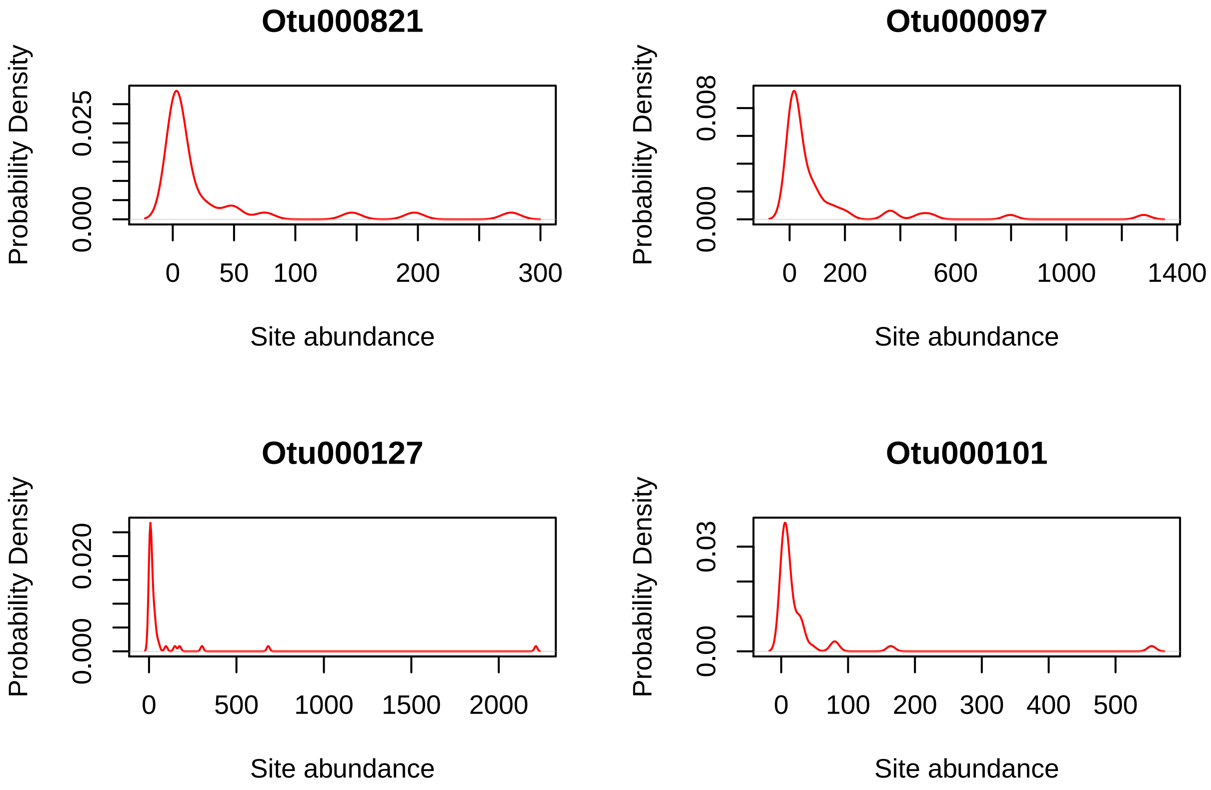
Question 8: In your own words, explain the species spatial abundance distribution and what it reveals.

> ***Answer 8***: The SSAD is a distribution describing the abundances of a pariticular species (or OTU) at all sites. It is like a continuous version of a histogram. It reveals what abundance (or range of abundance) a particular species is likely to be found at in the dataset.

In the R code chunk below, do the following:

1. Define a function that will generate the SSAD for a given OTU.
2. Draw six OTUs at random from the IN ponds dataset and and plot their SSADs as kernel density curves. Use **while loops** and **if** statements to accomplish this.

```r
# 1. Define an SSAD function
ssad <- function(x){
  ad <- c(2,2)
  ad <- OTUs[, otu]
  ad = as.vector(t(x = ad))
  ad = ad[ad > 0]
}
# 2. Set plot parameters
par(mfrow=c(2, 2))
# 3. Declare a counter variable
ct <- 0
# 4. Write a while loop to plot the SSADs of six species chosen at random
while (ct < 4){
  otu <- sample(1:length(OTUs), 1)
  ad <- ssad(otu)
  if (length(ad) > 10 & sum(ad > 100)){
    ct <- ct + 1
    plot(density(ad), col = "red", xlab="Site abundance",
         ylab="Probability Density", main = otu.names[otu])
  }
}
```



8) UNDERSTANDING SPATIAL SCALE

Many patterns of biodiversity relate to spatial scale.

Question 9: List, describe, and give examples of the two main aspects of spatial scale

> ***Answer 9***: Extent is the largest distance between any two sites or plots in a study. An example of a relatively small extent would be studying species diversity across sites within Indiana and an example of a larger extent would be studying species diversity across the entire US. Grain is essentially our resolution. It is the smallest unit by which our study is structured. An example of grain is the 5m X 5m quadrats that each of the sites in our project dataset were organized into.

9) CONSTRUCTING THE SPECIES-AREA RELATIONSHIP

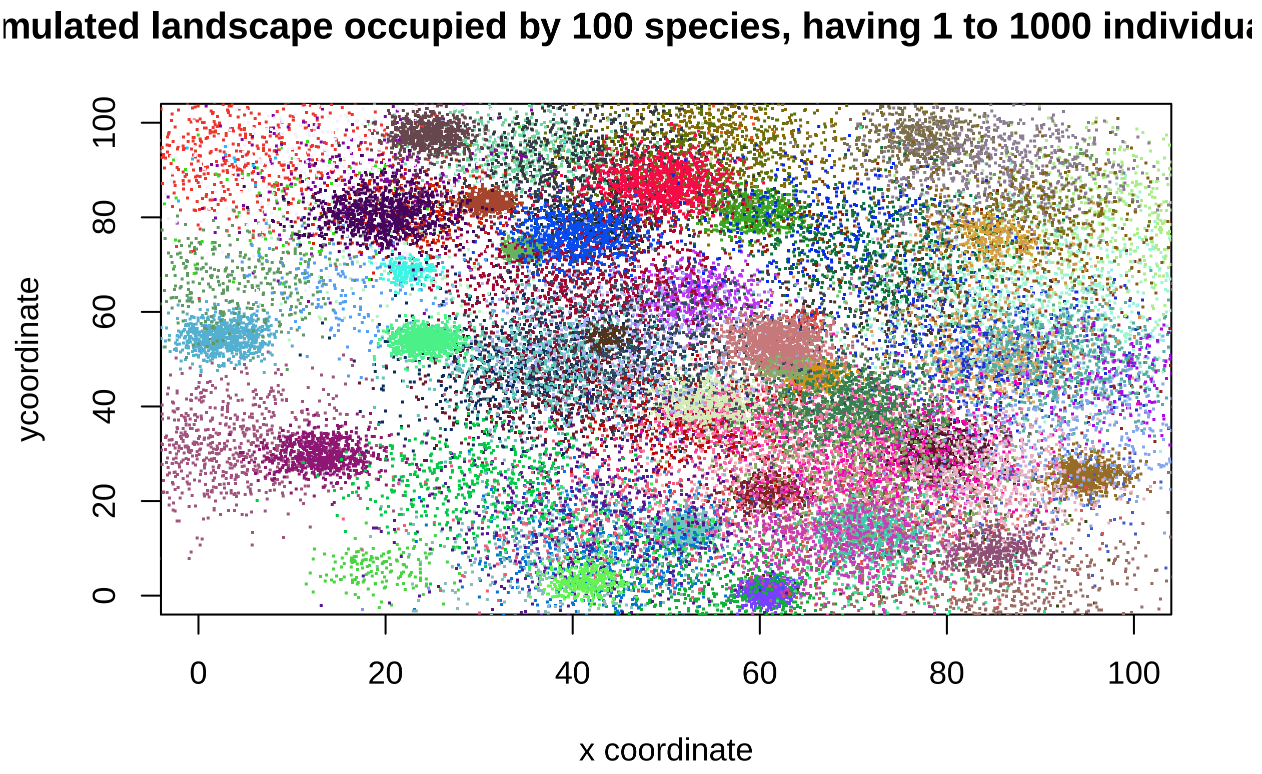
Question 10: In your own words, describe the species-area relationship.

> ***Answer 10***: The species area relationship describes how the size of our sampling area relates to observed species richness. When we increase our sampling area, do we see more new species (a positive slope) or do we see the same species regardless of extent (a slope of 0)?

In the R code chunk below, provide the code to:

1. Simulate the spatial distribution of a community with 100 species, letting each species have between 1 and 1,000 individuals.

```r
# 1. Declare variables to hold simulated community and species information
community <- c()
species <-c()
# 2. Populate the simulated landscape
plot(0, 0, col='white', xlim = c(0, 100), ylim = c(0,100),
     xlab='x coordinate', ylab='ycoordinate',
     main='A simulated landscape occupied by 100 species, having 1 to 1000 individuals each.')

while (length(community) < 100){
  std <- runif(1,1,10)
  ab <- sample(1000, 1)
  x <- rnorm(ab, mean = runif(1, 0, 100), sd = std)
  y <- rnorm(ab, mean = runif(1, 0, 100), sd = std)
  color <- c(rgb(runif(1),runif(1), runif(1)))

  points(x, y, pch=".", col=color)
  species <- list(x, y, color)
  community[[length(community)+1]] <- species
}
```



While consult the handout for assistance, in the R chunk below, provide the code to:

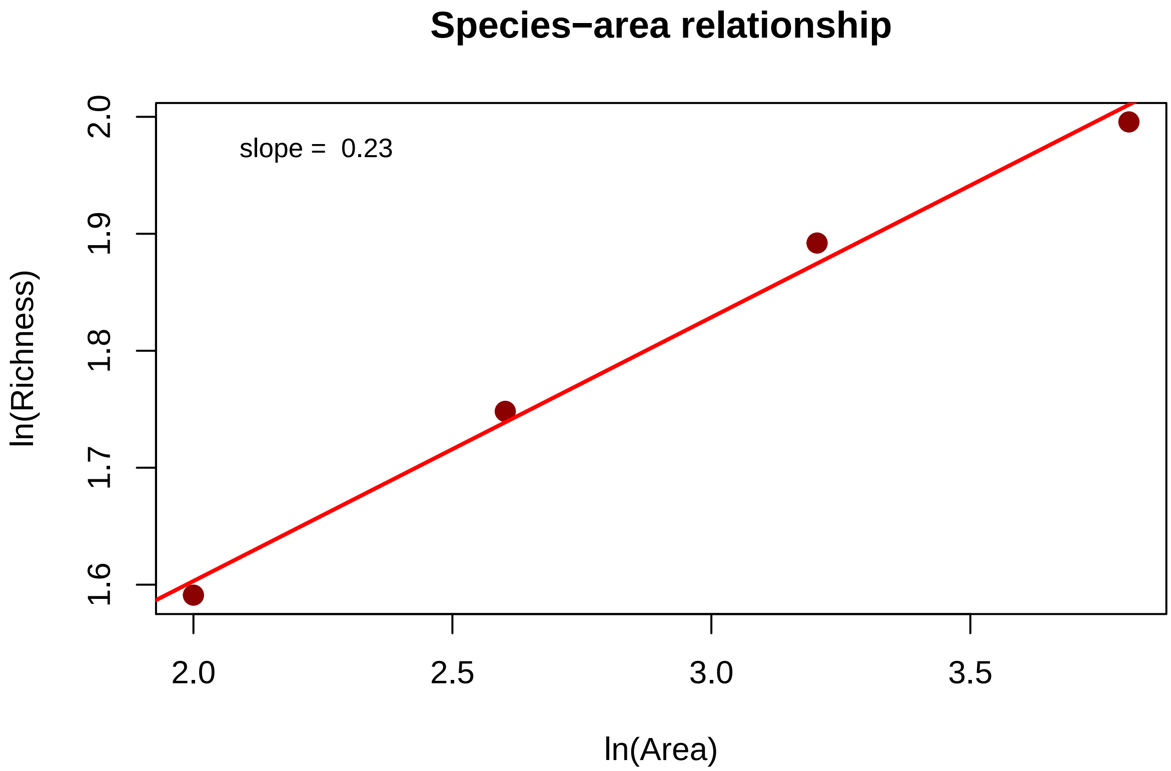
1. Use a nested design to examine the SAR of our simulated community.
2. Plot the SAR and regression line.

```r
# 1. Declare the spatial extent and lists to hold species richness and area data
lim <- 10
S.list <- c()
A.list <- c()
# 2. Construct a 'while' loop and 'for' loop combination to quantify the numbers of species for progres
while (lim <= 100){
  S <- 0

  for (sp in community){
    xs <- sp[[1]]
    ys <- sp[[2]]
    sp.name <- sp[[3]]
    xy.coords <- cbind(xs, ys)
    for (xy in xy.coords){
      if (max(xy) <= lim){
        S <- S + 1
        break
      }
    }
  }
  S.list <- c(S.list, log10(S))
  A.list <- c(A.list, log10(lim^2))
  lim <- lim *2
}
# 3. Be sure to log10-transform the richness and area data
```

In the R code chunk below, provide the code to:

1. Plot the richness and area data as a scatter plot.
2. Calculate and plot the regression line
3. Add a legend for the z-value (i.e., slope of the SAR)

```r
results <- lm(S.list ~ A.list)
plot(A.list, S.list, col="dark red", pch=20, cex=2,
     main="Species-area relationship",
     xlab='ln(Area)', ylab='ln(Richness)')

abline(results, col="red", lwd=2)

int <- round(results[[1]][[1]], 2)
z <- round(results[[1]][[2]], 2)
legend(x=2, y=2, paste(c('slope = ', z), collapse = " "), cex=0.8,
       box.lty=0)
```



Question 10a: Describe how richness relates to area in our simulated data by interpreting the slope of the SAR.

> ***Answer 10a***: Species richness increases with area (slope = 0.18). We observe more species as we increase the extent. (note that this slope varies from run to run because it is based on a random sample…so it may be slightly different than what I reported)

Question 10b: What does the y-intercept of the SAR represent?

> ***Answer 10b***: The y-intercept epresents log(c) or the number of species that would be expected to be in 1 square unit. Our `A list` begins at only 10. When A=1, log(A)=0 and we're left with the constant log(c).

SYNTHESIS

Load the dataset you are using for your project. Plot and discuss either the geogrpahic Distance-Decay relationship, the SSADs for at least four species, or any variant of the SAR (e.g., random accumulation of plots or areas, accumulation of contiguous plots or areas, nested design).

```r
setwd("c:/Users/matth/Documents/bin/QB2017_Gibson/project")
myData <- read.table("speciesdata_clean.csv", sep=",", header=T, row.names = 1)
envData <- read.table("environmentaldata(1).csv", sep=",", header=T, row.names = 1)

#Remove plant measurements
```

```r
envData_reduced <- envData[, 1:23]
#Remove categorical variables
envData_reduced <- subset(envData_reduced, select=-c(Mangement.type, Grazing.intensity))
#Remove country and year
envData_reduced <- subset(envData_reduced, select=-c(Country, Survey.year))


#Remove unneeded data
myData <- myData[, 4:ncol(myData)]

#Begin DISTANCE DECAY CODE

xy <- data.frame(site.name = row.names(envData), lats = envData$Latitude, lons = envData$Longitude)
coordinates(xy) <- ~lats+lons

comm.dist <- 1 - vegdist(myData)

proj4string(xy) <- CRS("+proj=longlat +datum=NAD83")
UTM <- spTransform(xy, CRS("+proj=utm + zone=51 ellps=WGS84"))
UTM <- as.data.frame(UTM)
xy$lats_utm <- UTM[,2]
xy$lons_utm <- UTM[,3]
lats <- as.numeric(xy$lats_utm)
lons <- as.numeric(xy$lons_utm)
# 3) Calculate geographic distance between plots and assign to the variable 'coord.dist'
coord.dist <- dist(as.matrix(lats, lons))
# 4) Transform environmental data to numeric type, and assign to variable 'x1'

x1 <- as.numeric(envData$Topsoil.pH)

#x1 <- as.numeric(envData$vegetation.height)

# 5) Using the `vegdist()` function in `simba`, calculate the Euclidean distance between the plots for
env.dist <- vegdist(x1, "euclidean")
# 6) Transform all distance matrices into database format using the `liste()` function in `simba`:
comm.dist.ls <- liste(comm.dist, entry="comm")
env.dist.ls <- liste(env.dist, entry = "env")
coord.dist.ls <- liste(coord.dist, entry="dist")
# 7) Create a data frame containing similarity of the environment and similarity of community.
df <- data.frame(coord.dist.ls, env.dist.ls[,3], comm.dist.ls[,3])

# 8) Attach the columns labels 'env' and 'struc' to the dataframe you just made.
names(df)[4:5] <- c("env", "struc")
attach(df)
```

```
## The following objects are masked from df (pos = 3):
##
##     dist, env, NBX, NBY, struc
```

```r
# 9) After setting the plot parameters, plot the distance-decay relationships, with regression lines in
par(mfrow=c(1, 2), pty="s")
plot(env, struc, xlab="Environmental Distance", ylab="1 - Bray-Curtis",
     main = "Environment (Topsoil pH)", col='SteelBlue')
```

```r
OLS <- lm(struc ~ env)
OLS
```

```
##
## Call:
## lm(formula = struc ~ env)
##
## Coefficients:
## (Intercept)          env
##     0.36118     -0.06438
```

```r
summary(OLS)
```

```
##
## Call:
## lm(formula = struc ~ env)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.34026 -0.08756 -0.00741  0.08155  0.50459
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.361181   0.001945  185.65   <2e-16 ***
## env         -0.064379   0.002816  -22.86   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1234 on 11626 degrees of freedom
## Multiple R-squared:  0.04303,    Adjusted R-squared:  0.04295
## F-statistic: 522.8 on 1 and 11626 DF,  p-value: < 2.2e-16
```

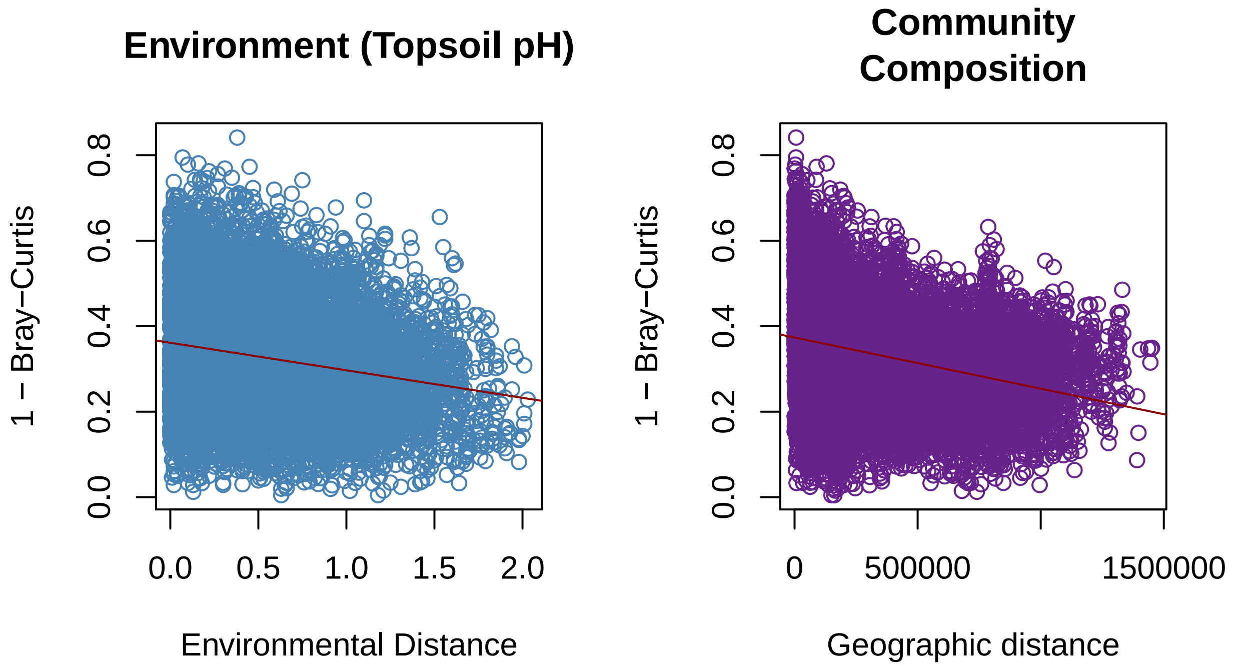
```r
abline(OLS, col="red4")

plot(dist, struc, xlab="Geographic distance", ylab="1 - Bray-Curtis",
     main = "Community\nComposition", col="darkorchid4")

OLS <- lm(struc ~ dist)
summary(OLS)
```

```
##
## Call:
## lm(formula = struc ~ dist)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.35076 -0.08220 -0.00093  0.08069  0.46858
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.735e-01  1.801e-03  207.41   <2e-16 ***
## dist        -1.196e-07  3.503e-09  -34.15   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1202 on 11626 degrees of freedom
## Multiple R-squared:  0.09114,    Adjusted R-squared:  0.09106
## F-statistic:  1166 on 1 and 11626 DF,  p-value: < 2.2e-16
```

```
OLS
```

```
##
## Call:
## lm(formula = struc ~ dist)
##
## Coefficients:
## (Intercept)         dist
##   3.735e-01   -1.196e-07
```

```
abline(OLS, col="red4")
```



```
# 10) Use `simba` to calculates the difference in slope or intercept of two regression lines
diffslope(env, struc, dist, struc)
```

```
##
```

```
## Is difference in slope significant?
## Significance is based on 1000 permutations
##
## Call:
## diffslope(x1 = env, y1 = struc, x2 = dist, y2 = struc)
##
## Difference in Slope: -0.06438
## Significance: 0.001
##
## Empirical upper confidence limits of r:
##      90%      95%    97.5%      99%
## 6.29e-09 7.94e-09 9.20e-09 1.05e-08
```

I looked at the distance decay relationships in our European plant diversity dataset incorporating both the environment (`topsoil pH`) and geographic distance. I chose to use pH in this example because it was heavily correlated with species richness in our initial analysis. Once I conduct a constrained ordination on this dataset, I will likely consider other environmental variables here.

Both plots have a negative slope, indicating that site similarity is related to both the environment (in this case topsoil pH) as well as geographic distance (as expected given the large extent of this dataset). Communities located close to eachother are more similar than those further away and communities that share similar Topsoil pH are more similar. The two slopes are significantly different from one another, but I am not sure it is appropriate to compare the two given that the X-axes of the two plots differ in scale by several orders of magnitude…But if we do, the slope of the environmental distance decay curve is much more negative (-0.06438) compared to the geographic distance decay curve (-1.196e-7) which indicates that Topsoil pH may be more strongly related to Bray-Curtis similarity than geographic distance. I would like to do variance partitioning with this data (which I plan to do) to see how space and environment overall explain species diversity.