Assignment: Temporal Diversity

Matt Gibson; Z620: Quantitative Biodiversity, Indiana University

15 February, 2017

OVERVIEW

In this Assignment, we extend our understanding of diversity from the spatial dimension to the temporal dimension.

After completing this exercise you will know how to:

1. wrangle a large dataset to visualize and analyze time series data
2. test hypotheses from experiments with temporal data
3. quantify temporal β-diversity and stability

Directions:

1. Change "Student Name" on line 3 (above) with your name.
2. Complete as much of the exercise 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 exercise.
4. Be sure to **answer the questions** in this exercise document; they also correspond to the Handout. 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 with the Assignment, **Knit** the text and code into a html file.
7. After Knitting, please submit the completed Assignment by creating a **pull request** via GitHub. Your pull request should include this file *temporal_assignment.Rmd* and the html output of `Knitr` (*temporal_assignment.html*).

1) R SETUP

Typically, the first thing you will do in either an R script or an RMarkdown file is setup your environment. This includes things such as setting the working directory and loading any packages that you will need.

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 "*/Week5-Temporal*" folder, and
4. load any packages you need to complete the assignment.

```
rm(list=ls())
getwd()
setwd("c:/Users/matth/Documents/bin/QB2017_Gibson/Week5-Temporal")

package.list <- c('vegan', 'tidyr', 'dplyr', 'codyn', 'ggplot2',
                  'MullerPlot', 'RColorBrewer', 'reshape2', 'lubridate',
```

```
                  'TTR', 'xtable', 'multcomp', 'pander', 'png', 'grid', 'tseries', 'nlme',
                  'forecast', 'lsmeans','vegan', 'ade4', 'viridis', 'gplots', 'BiodiversityR', 'indicsp
library(BiodiversityR)
library(indicspecies)
library(MullerPlot)
for (package in package.list){
  if (!require(package, character.only=T, quietly=T)){
    install.packages(package, repos="http://cran.us.r-project.org")
    library(package, character.only=T)
  }
}
```

2) LOADING DATA

Load dataset

In the R code chunk below, do the following:

1. load the `portal` dataset from in the "*/Week5/data*" folder, and
2. explore the structure of the dataset.

```
portal <- read.table("data/combined.csv", sep=",", header=T)

str(portal)
```

```
## 'data.frame':    34786 obs. of  13 variables:
##  $ record_id      : int  1 72 224 266 349 363 435 506 588 661 ...
##  $ month          : int  7 8 9 10 11 11 12 1 2 3 ...
##  $ day            : int  16 19 13 16 12 12 10 8 18 11 ...
##  $ year           : int  1977 1977 1977 1977 1977 1977 1977 1978 1978 1978 ...
##  $ plot_id        : int  2 2 2 2 2 2 2 2 2 2 ...
##  $ species_id     : Factor w/ 48 levels "AB","AH","AS",..: 16 16 16 16 16 16 16 16 16 16 ...
##  $ sex            : Factor w/ 3 levels "","F","M": 3 3 1 1 1 1 1 1 3 1 ...
##  $ hindfoot_length: int  32 31 NA NA NA NA NA NA NA NA ...
##  $ weight         : int  NA NA NA NA NA NA NA NA 218 NA ...
##  $ genus          : Factor w/ 26 levels "Ammodramus","Ammospermophilus",..: 13 13 13 13 13 13 13 13
##  $ species        : Factor w/ 40 levels "albigula","audubonii",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ taxa           : Factor w/ 4 levels "Bird","Rabbit",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ plot_type      : Factor w/ 5 levels "Control","Long-term Krat Exclosure",..: 1 1 1 1 1 1 1 1 1 1
```

```
dplyr::count(portal, plot_id)
```

```
## # A tibble: 24 × 2
##    plot_id     n
##      <int> <int>
## 1        1  1989
## 2        2  2191
## 3        3  1808
## 4        4  1960
## 5        5  1150
## 6        6  1562
```

```
## 7        7   729
## 8        8  1881
## 9        9  1929
## 10      10   324
## # ... with 14 more rows
```

```
dplyr::count(portal, species)
```

```
## # A tibble: 40 × 2
##            species     n
##             <fctr> <int>
## 1         albigula  1252
## 2        audubonii    75
## 3          baileyi  2891
## 4        bilineata   303
## 5  brunneicapillus    50
## 6        chlorurus    39
## 7           clarki     1
## 8         eremicus  1299
## 9           flavus  1597
## 10      fulvescens    75
## # ... with 30 more rows
```

Question 1: Describe some of the attributes of the `portal` dataset.

a. How many plots are in `portal`?
b. How many rodent species are there in the `portal` dataset?

Answer 1a: 24 ***Answer 1b***: 40

3) WRANGLING THE PORTAL DATASET

In the R code chunk below, do the following:

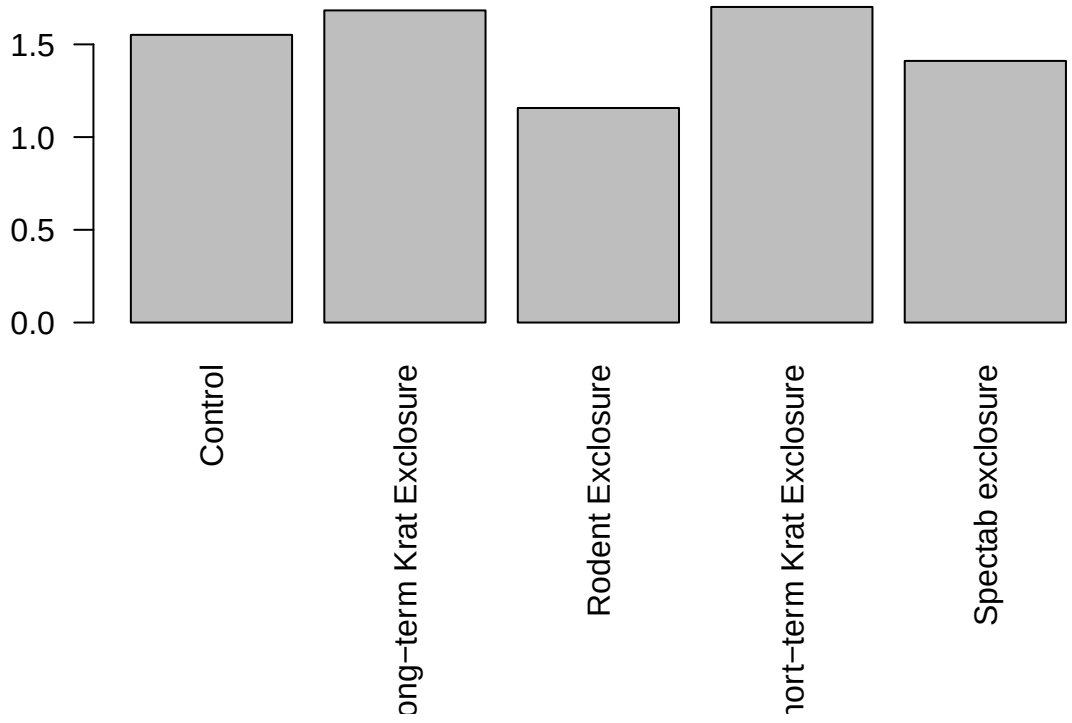
1. Create a site-by-species matrix for any year of your choosing.
2. Create a vector of plot_type for sites in the site-by-species matrix.
3. Analyze alpha diversity (e.g., Shannon/Simpson) across the sites for that year.
4. Create a PCoA ordination of your site-by-species matrix.
5. Using the hypothesis testing tools you learned in the beta-diversity module, test the hypothesis that species abundances across sites vary as a factor of treatment type (i.e., plot_type).

```
portal <- unite(portal, col=date, c(year, month, day), sep = "-", remove=F)
portal <- unite(portal, col = taxon, c(genus, species), sep = "_", remove=F)

time.by.species <- group_by(portal, year, plot_id, plot_type) %>%
  count(taxon) %>% spread(key = taxon, value = n, fill = 0)
time.by.species <- as.data.frame(time.by.species)

matrix <- dplyr::filter(time.by.species, year == 1984)
matrix <- as.data.frame(matrix)
```

```
siteBySpecies <- matrix[,4:50]

############################Calculate shannon diversity using vegan#####################################
print(shannon <- diversity(siteBySpecies, index = "shannon"))
```

```
##  [1] 1.3652619 2.0160552 2.1048747 0.8861713 1.3552517 1.8036086 0.7963116
##  [8] 1.4610630 1.4565878 0.6615632 1.7010410 1.9192314 1.6307278 1.5499932
## [15] 1.9432443 1.2634795 1.3386461 1.6974928 1.3534969 1.6755192 1.3316643
## [22] 1.5395696 1.2852930 1.5797406
```

```
siteShannon <- cbind(as.data.frame(matrix[,3]), as.data.frame(shannon))


a <- aggregate(shannon ~ matrix[, 3], data=siteShannon, FUN=mean)
op <- par(mar = c(10,4,4,2) + 0.1)

barplot(a$shannon, names.arg=a$`matrix[, 3]`, las=2)
```




```
par(op) ## reset

#PCoA
species.db <- vegdist(siteBySpecies, method = "bray", diag = T)
species.pcoa <- cmdscale(species.db, eig=T, k = 3)
```

```r
explainvar1 <- round(species.pcoa$eig[1] / sum(species.pcoa$eig), 3) * 100
explainvar2 <- round(species.pcoa$eig[2] / sum(species.pcoa$eig), 3) * 100
explainvar3 <- round(species.pcoa$eig[3] / sum(species.pcoa$eig), 3) * 100
sum.eig <- sum(explainvar1, explainvar2, explainvar3)

#Variance explained by first axis
explainvar1
```

```
## [1] 56.3
```

```r
#variance explained by second axis
explainvar2
```

```
## [1] 12.3
```

```r
#variance explained by third axis
explainvar3
```

```
## [1] 8.8
```

```r
sum.eig
```

```
## [1] 77.4
```

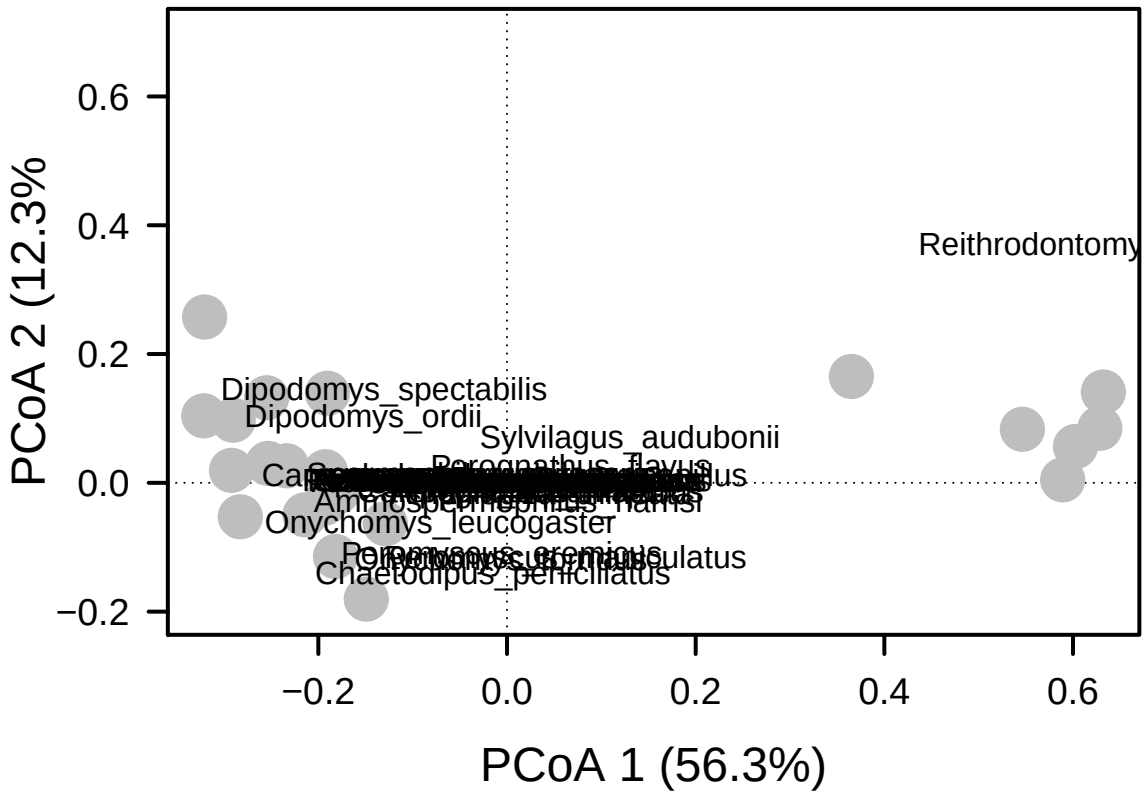
```r
par(mar = c(5, 5, 1, 2) + 0.1)

plot(species.pcoa$points[ ,1], species.pcoa$points[ ,2], ylim = c(-0.2, 0.7),
     xlab = paste("PCoA 1 (", explainvar1, "%)", sep = ""),
     ylab = paste("PCoA 2 (", explainvar2, "%", sep = ""),
     pch = 16, cex = 2.0, type = "n", cex.lab = 1.5, cex.axis = 1.2, axes = F)

axis(side = 1, labels = T, lwd.ticks = 2, cex.axis = 1.2, las = 1)
axis(side = 2, labels = T, lwd.ticks = 2, cex.axis = 1.2, las = 1)
abline(h = 0, v = 0, lty = 3)
box(lwd = 2)

points(species.pcoa$points[ ,1], species.pcoa$points[ ,2],
       pch = 19, cex = 3, bg = "gray", col = "gray")
text(species.pcoa$points[ ,1], species.pcoa$points[ ,2],
     labels = row.names(species.pcoa$points))


speciesREL <- siteBySpecies
  for(i in 1:nrow(siteBySpecies)){
    speciesREL[i, ] = siteBySpecies[i, ]/ sum(siteBySpecies[i, ])
  }

species.pcoa <- add.spec.scores(species.pcoa, speciesREL, method = "pcoa.scores")
```

```
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
```

```
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
```

```
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
## Warning in cor(comm[, i], ordiscores[, j]): the standard deviation is zero
```

```r
text(species.pcoa$cproj[ ,1], species.pcoa$cproj[ ,2],
     labels = row.names(species.pcoa$cproj), col = "black")
```




```r
#######HYPOTHESIS TESTING#############################################################################

#quality <- c(rep("HQ", 13), rep("MQ", 5), rep("LQ", 6), rep("MQ", 5))
adonis(siteBySpecies ~ matrix$plot_type, method = "bray", permutations = 999)
```

```
##
## Call:
## adonis(formula = siteBySpecies ~ matrix$plot_type, permutations = 999,      method = "bray")
##
## Permutation: free
## Number of permutations: 999
##
## Terms added sequentially (first to last)
##
##                   Df SumsOfSqs MeanSqs F.Model     R2 Pr(>F)
## matrix$plot_type  4    2.2725 0.56811  3.7564 0.4416  0.002 **
## Residuals        19    2.8735 0.15124         0.5584
## Total            23    5.1460                 1.0000
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#Quality is significant in the PERMANOVA ( p = 0.001)

indval <- multipatt(siteBySpecies, cluster = matrix$plot_type, func = "IndVal.g", control = how(nperm=99
summary(indval)
```

```
##
##  Multilevel pattern analysis
##  ---------------------------
##
##  Association function: IndVal.g
##  Significance level (alpha): 0.05
##
##  Total number of species: 47
##  Selected number of species: 4
##  Number of species associated to 1 group: 0
##  Number of species associated to 2 groups: 1
##  Number of species associated to 3 groups: 3
##  Number of species associated to 4 groups: 0
##
##  List of species associated to each combination:
##
##  Group Long-term Krat Exclosure+Rodent Exclosure  #sps.  1
##                              stat p.value
## Reithrodontomys_megalotis 0.905   0.007 **
##
##  Group Control+Short-term Krat Exclosure+Spectab exclosure  #sps.  3
##                           stat p.value
## Dipodomys_spectabilis  0.958   0.001 ***
## Dipodomys_ordii        0.923   0.009 **
## Spermophilus_spilosoma 0.804   0.043 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
species.rel <- decostand(siteBySpecies, method = "total")
phi <- multipatt(species.rel, cluster=matrix$plot_type, func = "r.g", control = how(nperm=999))
summary(phi)
```

```
##
##  Multilevel pattern analysis
##  ---------------------------
##
##  Association function: r.g
##  Significance level (alpha): 0.05
##
##  Total number of species: 47
##  Selected number of species: 3
##  Number of species associated to 1 group: 0
##  Number of species associated to 2 groups: 1
##  Number of species associated to 3 groups: 2
##  Number of species associated to 4 groups: 0
##
##  List of species associated to each combination:
```

```
##
##  Group Long-term Krat Exclosure+Rodent Exclosure  #sps.  1
##                            stat p.value
## Reithrodontomys_megalotis  0.7   0.025 *
##
##  Group Control+Short-term Krat Exclosure+Spectab exclosure  #sps.  2
##                         stat p.value
## Dipodomys_merriami    0.741   0.004 **
## Dipodomys_spectabilis 0.638   0.041 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Question 2: Describe how different biodiversity estimates vary among sites.

a. Does diversity vary among sites? Does this correspond to treatment type?
b. Is treatment type a significant predictor of site dissimilarity?

> ***Answer 2a***: Yes, diversity varies slightly among sites. There is a weak correspondance with treatment type as seen in the barplot. Long-term Krat exposure and Short-term Krat exposure plots had higher diversity on average. Rodent exclosures had the lowest diversity on average.
> ***Answer 2b***: Based on the PERMANOVA, treatment type is a significant predictor of site similarity (P = 0.001). More details on these effects can be seen in the indVal and Phi tests.

4) TIME SERIES ANALYSIS

In the R code chunk below, do the following:

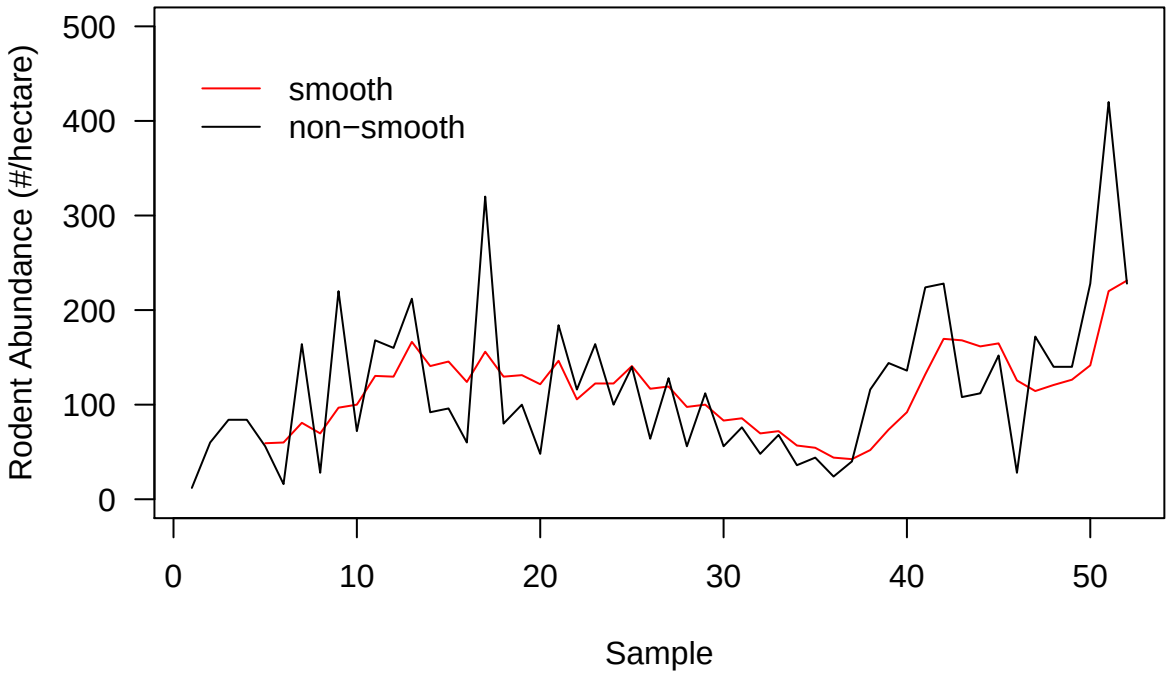
1. Create a time-by-species matrix that includes year, month, and plot_id for a site other than plot_id 2.
2. Examine per-hectare rodent abundance using simple moving average smoothing.
3. Test whether your data meets the assumption of stationarity.
4. If it does not meet this asumption, explore wasy to make your data stationary.
5. Examine and plot time lags using the partial autocorrelation function (PACF) and autocorrelation function (ACR).
6. Use the tools outlined in the Handout to create an ARMA model.

```
time.by.species_2 <- filter(portal, taxa=="Rodent") %>% group_by(year,month,plot_id) %>% count(taxon)

time.by.species_2$season <- NA
time.by.species_2$season <- time.by.species_2$month %in% c(6:10)

time.by.species_2$season <- ifelse(time.by.species_2$season == T, "rain", "norain")
group_by(time.by.species_2, year, season)
```

```
## Source: local data frame [16,391 x 6]
## Groups: year, season [52]
##
##     year month plot_id                   taxon     n season
##    <int> <int>   <int>                   <chr> <int>  <chr>
## 1   1977     7       1       Dipodomys_merriami     2   rain
## 2   1977     7       1       Perognathus_flavus     1   rain
## 3   1977     7       2 Chaetodipus_penicillatus     1   rain
```

```
## 4   1977     7       2       Dipodomys_merriami     1   rain
## 5   1977     7       2         Neotoma_albigula     1   rain
## 6   1977     7       2      Peromyscus_eremicus     1   rain
## 7   1977     7       3       Dipodomys_merriami     2   rain
## 8   1977     7       3    Dipodomys_spectabilis     1   rain
## 9   1977     7       3         Neotoma_albigula     1   rain
## 10  1977     7       4       Dipodomys_merriami     1   rain
## # ... with 16,381 more rows
```

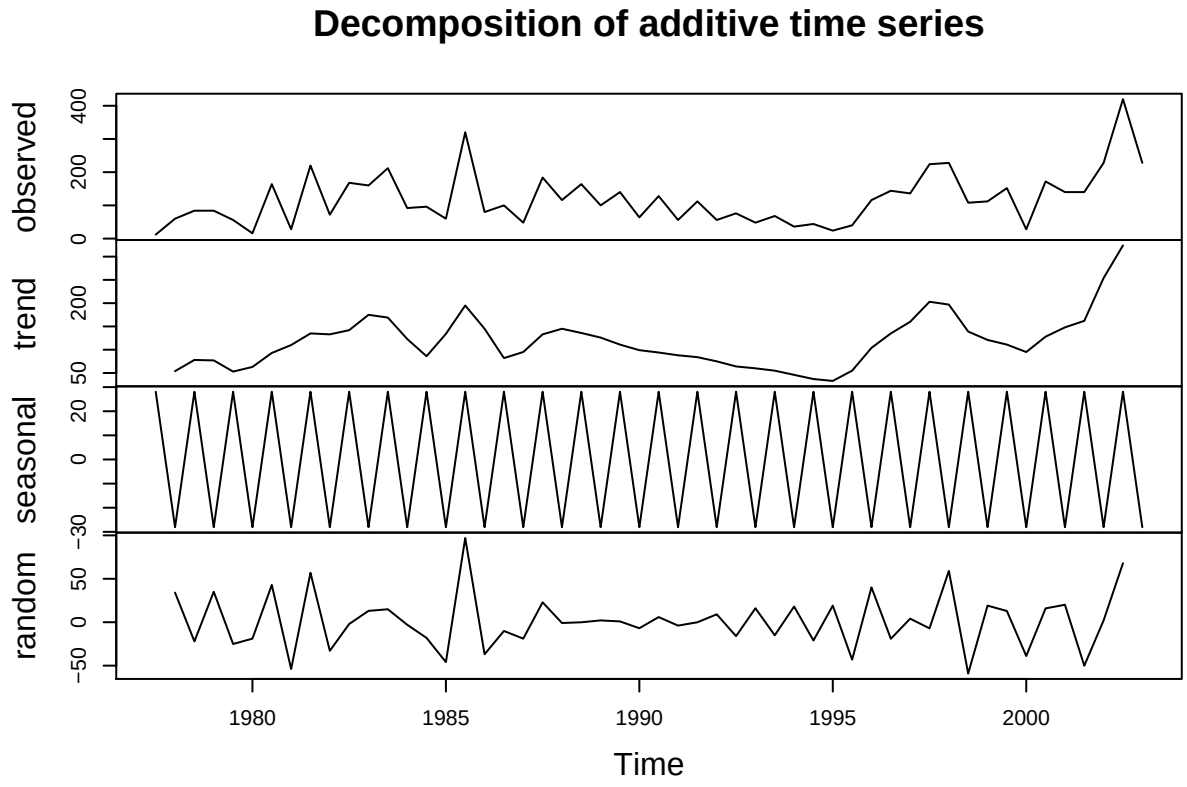
```
#CHANGE TO 6!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
abund <- filter(time.by.species_2, plot_id == 6) %>%
  group_by(year, season) %>%
  count(wt = n)

abund$nn <- abund$nn*4

abund.ts <- ts(abund$nn, frequency = 2, start = c(1977, 2))

#plot.ts(abund.ts, type = "l", ylab = "Rodent Abundance (#/hectare",
#        xlab = "Time (year)", las = 1, ylim = c(0,500))

abund.sm <- SMA(abund$nn, n = 5)
plot(abund.sm, type = "l", col = "red", ylab = "Rodent Abundance (#/hectare)",
     xlab = "Sample", las = 1, ylim = c(0, 500))
lines(abund$nn, col = "black")

legend(0, 475, col = c("red", "black"), lty = c(1,1),
       c("smooth", "non-smooth"), bty= "n", cex = 1)
```




```
#TESTING STATIONARITY#########################################################################
adf.raw <- adf.test(abund.ts, alternative = "stationary")
adf.raw$p.value
```

```
## [1] 0.8389527
```

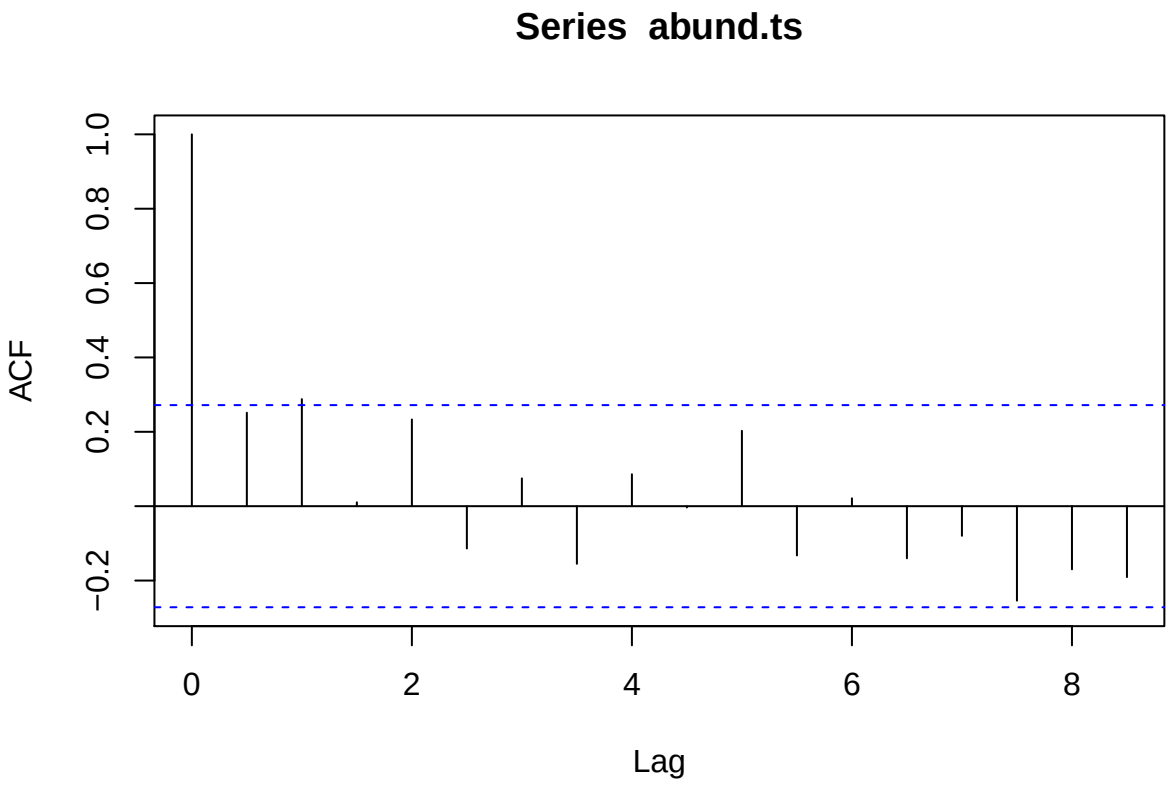
```
abund.comp <- decompose(abund.ts)
plot(abund.comp)
```



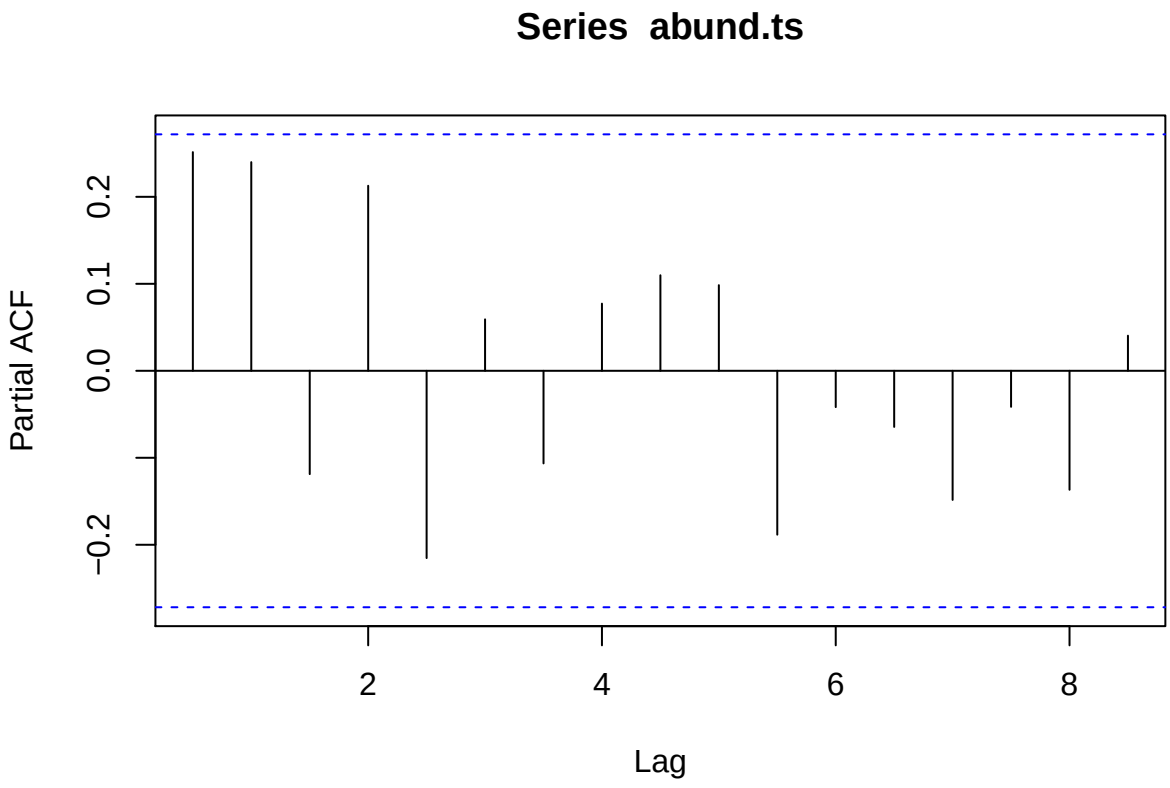

```
#Differencing
abund.ts.diff <- diff(abund.ts)
adf.diff <- adf.test(abund.ts.diff, alternative = "stationary")
adf.diff$p.value
```

```
## [1] 0.02504877
```

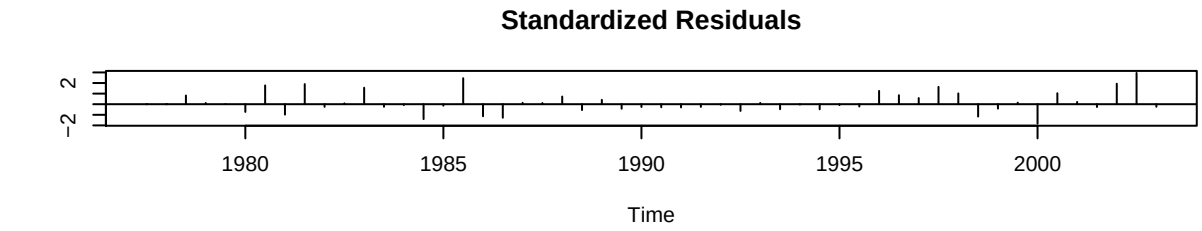
```
acf(abund.ts)
```

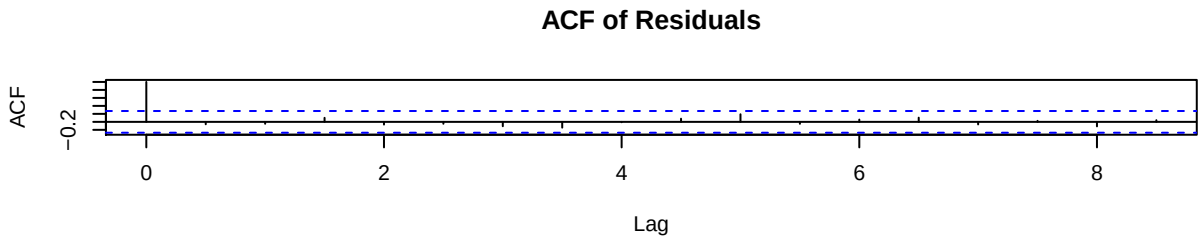


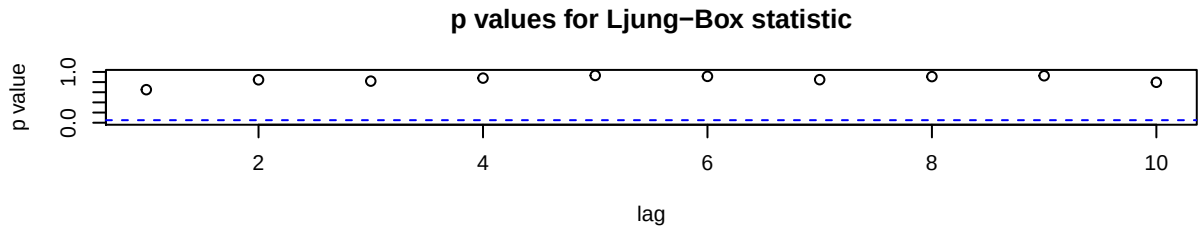

```
pacf(abund.ts)
```



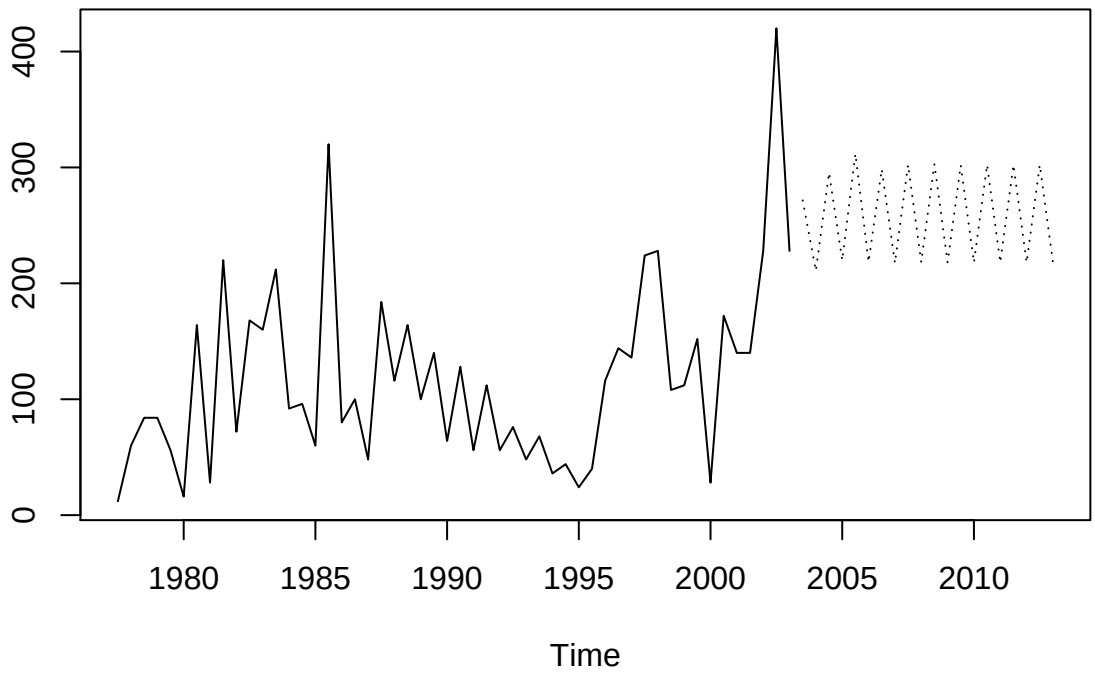

```
abund.arm <- auto.arima(abund.ts)
abund.arm <- arima((abund.ts), c(0,0,1), seasonal = list(order = c(2, 1, 0), period = 2), include.mean =
tsdiag(abund.arm)
```








```
pred.arm <- predict(abund.arm, n.ahead = 20)
ts.plot(abund.ts, pred.arm$pred, lty= c(1,3))
```



Question 3: Describe the results from your time series analysis.

a. Does your data meet the assumption of stationarity? If not, what does this violation imply?
b. What does the ACF function do and how does it relate to the ARMA model? How does this differ from the autocorrelation function (ACF)?
c. What results can you conclude from your full ARMA model along with other methods outlined in the time series setcion of the Handout?

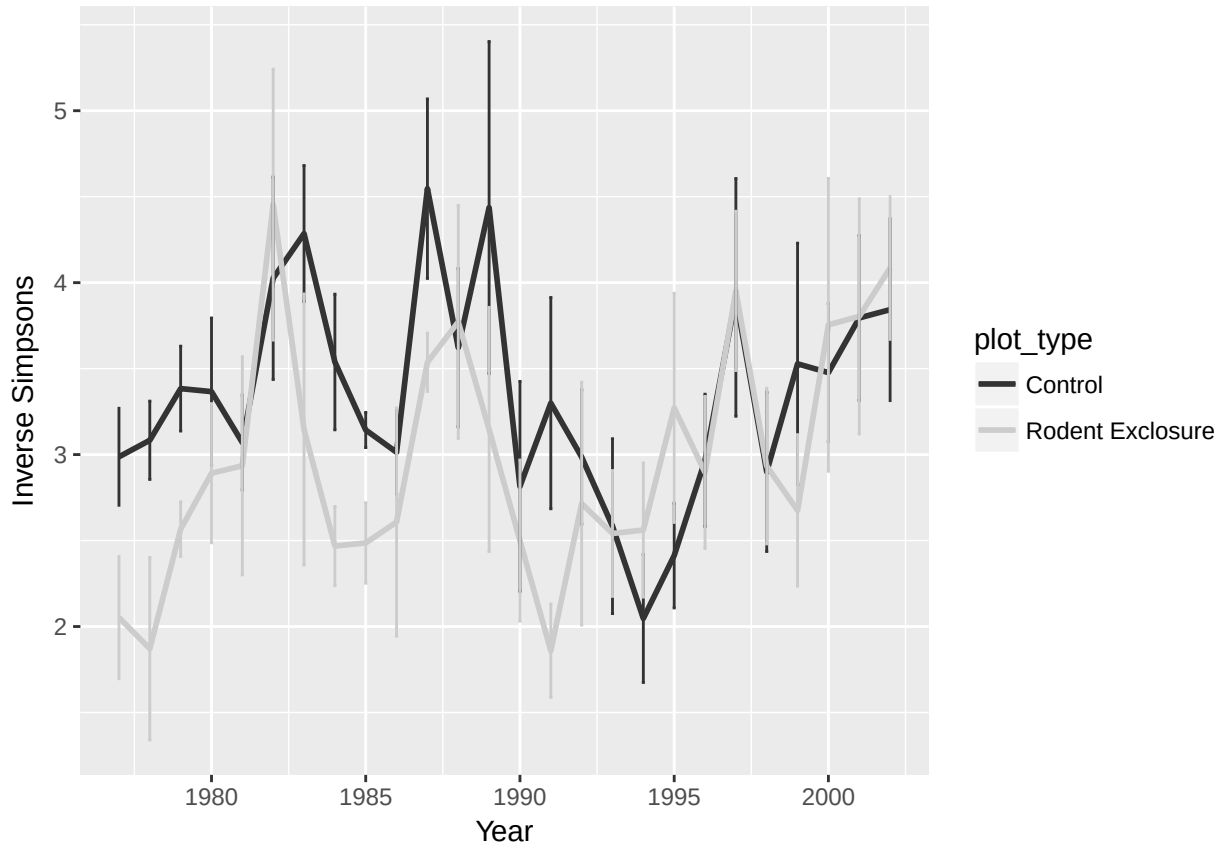
Answer 3a: No, initially it did not. After the differencing correction, though, we can reject the null of non-stationarity (P = 0.0250). Not meeting the assumption of stationarity implies that the mean, variance, or covariance in this series is affected by time. ***Answer 3b***: I do not understand what two functions this question wants us to compare. The ACF function identifes the lags in our time series data. By looking at the correlations of lagged intervals, we can identify specific intervals that might be useful in parameterizing our final ARMA model. ***Answer 3c***: Based on the ARIMA model and predictions, we may expect rodent abundance (#/hectare) to fluctuate between ~225 and ~310 over the 20 years following 2002. I am not sure that I would put much faith in these projections, though. In the 22 years leading up to 2002, variance in rodent abundance from year to year was high. While the variance would ideally be captured in our ARIMA model, the residual random variation in the time series is quite high (see decomposition plots). I am also not entirely sure if I provided the correct parameters to the ARIMA function. I would like to have this discussed in more detail in class.

5) REPEATED MEASURES ANALYSIS OF VARIANCE (RM-ANOVA)

In the R code chunk below, do the following:

1. Create an appropriate data frame for RM-ANOVA (e.g., yearly species abundance values within plots).
2. Calculate the inverse of Simpson's diversity for each year, and plot it as a function of year for the Control and Rodent Exclosure plots.
3. Perform an RM-ANOVA and construct a F-test using the AR(1), compound symmetery, and unstructured covariance structures.

```
time.by.species <- group_by(portal, year, plot_id,
                            plot_type) %>% count(taxon) %>% spread(key = taxon, value = n, fill = 0)

inv_s <- diversity(as.data.frame(time.by.species)[,-c(1:3)], "inv")


rich.all <- data.frame(time.by.species[,1:3,], inv_s)
names(rich.all)[4] <- "inverse_S"

rich.treat <- rich.all[which(rich.all$plot_type == "Control" | rich.all$plot_type == "Rodent Exclosure"),]



rich.treat.plot <- group_by(rich.treat, plot_type, year) %>%
  summarise(
    mean = mean(inverse_S),
    sd = sd(inverse_S),
    n = n(),
    sem = sd/sqrt(n))

rich.plot <- ggplot(rich.treat.plot, aes(x = year, y = mean, color = plot_type)) +
  geom_line(size = 1, show.legend = T) +
  geom_errorbar(aes(ymin = mean - sem, ymax = mean + sem), width = .1) +
  xlim(1977, 2002)+
  xlab("Year")+
  ylab("Inverse Simpsons")+
  scale_color_grey()
plot(rich.plot)
```




```
#Autoregressive process order 1
rich.rm <- lme(inverse_S ~ plot_type * year, random=  ~ 1 | plot_id,
               correlation = corAR1(form = ~1 | plot_id),
               data=rich.treat)

summary(rich.rm)
```

```
## Linear mixed-effects model fit by REML
##  Data: rich.treat
##        AIC      BIC    logLik
##   1153.217 1180.322 -569.6087
##
## Random effects:
##  Formula: ~1 | plot_id
##         (Intercept) Residual
## StdDev:   0.6469478 1.250206
##
## Correlation Structure: AR(1)
##  Formula: ~1 | plot_id
##  Parameter estimate(s):
##       Phi
## 0.4250398
## Fixed effects: inverse_S ~ plot_type * year
##                                   Value Std.Error  DF    t-value p-value
## (Intercept)                     1.36695  33.32814 343  0.0410150  0.9673
## plot_typeRodent Exclosure     -73.37575  52.01883  12 -1.4105613  0.1838
```

```
## year                              0.00100   0.01675 343  0.0595739  0.9525
## plot_typeRodent Exclosure:year    0.03670   0.02614 343  1.4039132  0.1612
##  Correlation:
##                                (Intr) plt_RE year
## plot_typeRodent Exclosure      -0.641
## year                           -1.000  0.641
## plot_typeRodent Exclosure:year  0.641 -1.000 -0.641
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -1.93162654 -0.70434296 -0.08293334  0.45856267  5.06139168
##
## Number of Observations: 359
## Number of Groups: 14
```

```
anova(rich.rm)
```

```
##                numDF denDF   F-value p-value
## (Intercept)        1   343 256.22009  <.0001
## plot_type          1    12   0.72274  0.4119
## year               1   343   1.56060  0.2124
## plot_type:year     1   343   1.97097  0.1612
```

```
set.caption("RMANOVA for Portal")
pander(anova(rich.rm))
```

Table 1: RMANOVA for Portal

	numDF	denDF	F-value	p-value
(Intercept)	1	343	256.2	0
plot_type	1	12	0.7227	0.4119
year	1	343	1.561	0.2124
plot_type:year	1	343	1.971	0.1612

```
lsmeans(rich.rm, ~plot_type)
```

```
## NOTE: Results may be misleading due to involvement in interactions

##  plot_type          lsmean        SE df lower.CL upper.CL
##  Control          3.352478 0.2644674 13 2.781131 3.923825
##  Rodent Exclosure 3.002943 0.3066216 12 2.334872 3.671014
##
## Confidence level used: 0.95
```

```
#Compounds symmetry
rich.rm <- lme(inverse_S ~ plot_type * year, random=  ~ 1 | plot_id,
               correlation = corCompSymm(form = ~1 | plot_id),
               data=rich.treat)

summary(rich.rm)
```

```
## Linear mixed-effects model fit by REML
##  Data: rich.treat
##        AIC      BIC    logLik
##   1215.368 1242.473 -600.6842
##
## Random effects:
##  Formula: ~1 | plot_id
##         (Intercept) Residual
## StdDev:   0.7104222 1.214275
##
## Correlation Structure: Compound symmetry
##  Formula: ~1 | plot_id
##  Parameter estimate(s):
##          Rho
## 6.672013e-18
## Fixed effects: inverse_S ~ plot_type * year
##                                   Value Std.Error  DF    t-value p-value
## (Intercept)                    13.26069  22.33564 343  0.5937008  0.5531
## plot_typeRodent Exclosure     -68.60423  34.97126  12 -1.9617316  0.0734
## year                           -0.00498   0.01123 343 -0.4438008  0.6575
## plot_typeRodent Exclosure:year   0.03431   0.01758 343  1.9518712  0.0518
##  Correlation:
##                                (Intr) plt_RE year
## plot_typeRodent Exclosure      -0.639
## year                           -1.000  0.639
## plot_typeRodent Exclosure:year  0.639 -1.000 -0.639
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -2.0346517 -0.6823806 -0.1001395  0.4790022  5.1406695
##
## Number of Observations: 359
## Number of Groups: 14
```

```
anova(rich.rm)
```

```
##                numDF denDF   F-value p-value
## (Intercept)        1   343 255.02615  <.0001
## plot_type          1    12   0.73483  0.4081
## year               1   343   1.08893  0.2974
## plot_type:year     1   343   3.80980  0.0518
```

```
set.caption("RMANOVA for Portal")
pander(anova(rich.rm))
```

Table 2: RMANOVA for Portal

	numDF	denDF	F-value	p-value
(Intercept)	1	343	255	0
plot_type	1	12	0.7348	0.4081
year	1	343	1.089	0.2974
plot_type:year	1	343	3.81	0.05177

```
lsmeans(rich.rm, ~plot_type)
```

```
## NOTE: Results may be misleading due to involvement in interactions

##  plot_type          lsmean        SE df lower.CL upper.CL
##  Control          3.348317 0.2649103 13 2.776013 3.920621
##  Rodent Exclosure 2.997377 0.3064250 12 2.329735 3.665020
##
## Confidence level used: 0.95
```

```
#Unstructured
rich.rm <- lme(inverse_S ~ plot_type * year, random=  ~ 1 | plot_id,
               data=rich.treat)

summary(rich.rm)
```

```
## Linear mixed-effects model fit by REML
##  Data: rich.treat
##        AIC      BIC    logLik
##   1213.368 1236.601 -600.6842
##
## Random effects:
##  Formula: ~1 | plot_id
##         (Intercept) Residual
## StdDev:   0.7104222 1.214275
##
## Fixed effects: inverse_S ~ plot_type * year
##                                    Value Std.Error  DF    t-value p-value
## (Intercept)                     13.26069  22.33564 343  0.5937008  0.5531
## plot_typeRodent Exclosure      -68.60423  34.97126  12 -1.9617316  0.0734
## year                            -0.00498   0.01123 343 -0.4438008  0.6575
## plot_typeRodent Exclosure:year   0.03431   0.01758 343  1.9518712  0.0518
##  Correlation:
##                                (Intr) plt_RE year
## plot_typeRodent Exclosure      -0.639
## year                           -1.000  0.639
## plot_typeRodent Exclosure:year  0.639 -1.000 -0.639
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -2.0346517 -0.6823806 -0.1001395  0.4790022  5.1406695
##
## Number of Observations: 359
## Number of Groups: 14
```

```
anova(rich.rm)
```

```
##                numDF denDF   F-value p-value
## (Intercept)        1   343 255.02615  <.0001
## plot_type          1    12   0.73483  0.4081
## year               1   343   1.08893  0.2974
## plot_type:year     1   343   3.80980  0.0518
```

```
set.caption("RMANOVA for Portal")
pander(anova(rich.rm))
```

Table 3: RMANOVA for Portal

	numDF	denDF	F-value	p-value
(Intercept)	1	343	255	0
plot_type	1	12	0.7348	0.4081
year	1	343	1.089	0.2974
plot_type:year	1	343	3.81	0.05177

```
lsmeans(rich.rm, ~plot_type)
```

```
## NOTE: Results may be misleading due to involvement in interactions
```

```
##  plot_type          lsmean        SE df lower.CL upper.CL
##  Control          3.348317 0.2649103 13 2.776013 3.920621
##  Rodent Exclosure 2.997377 0.3064250 12 2.329735 3.665020
##
## Confidence level used: 0.95
```

Question 4: Describe the results from your RM-ANOVA.

a. In your own words describe what a RM-ANOVA test is doing
b. Is there a noticeable trend in the inverse of Simpson's diversity over time?
c. What does the result of your F-test tell you?
d. Of the three RM-ANOVA models with different covariance structures, which one is best? How does this affect the interpretation of your data?

Answer 4a: RM-ANOVA accounts for non-independence of repeated observations. Measurements from a single individual sampled multiple times over the course of the experiment may vary, but they are not independent. RM-ANOVA accounts for repeated measures by including the term time*treatment.*
Answer 4b*: There is no noticeable trend, but the inverse of Simpson's diversity does change considerably over the course of the study. Overall, the control treatment appears to have higher diversity on average than the rodent exclosure treatment.* **Answer 4c***: The F-test shows no significant effect of* `plot_type`*,* `year`*, or their interaction on the inverse of Simpson's Diversity. This may indicate that, despite their apparent difference in the plot, the rodent exclosure treatment did not significantly reduce diversity.*
4d*: The RM-ANOVA model using the autoregressive process order of 1 covariance structure appears to be the best model. It has the lowest AIC (1153.2), the lowest BIC (1180.3), and the highest log-liklihood (-569.6087). This means that the the observed values for the inverse of Simpson's Diversity were likely most influenced by the observations at the time unit before.

6) TEMPORAL BETA DIVERSITY

Turnover

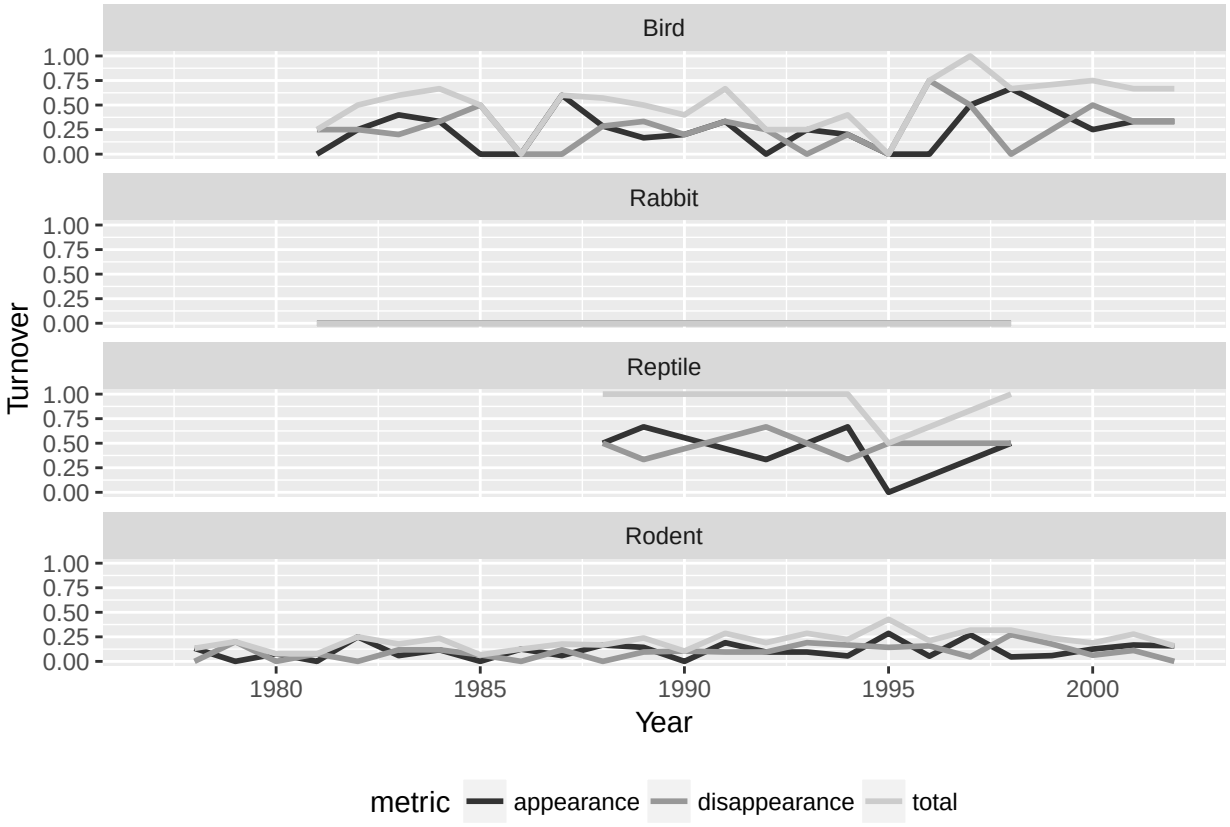
In the R code chunk below, do the following:

1. Calculate species abundances for each taxonomic group (the `taxa` column).
2. Calculate total turnover and turnover due to the gain/loss of species for each group.
3. Visualize turnover within each group

```
portal.species.abunds <- group_by(portal, year, taxa) %>% count(taxon)

portal.total <- turnover(df = portal.species.abunds,
                         time.var = "year",
                         species.var = "taxon",
                         abundance.var = "n",
                         replicate.var = "taxa",
                         metric = "total")

portal.appearance <- turnover(df = portal.species.abunds,
                         time.var = "year",
                         species.var = "taxon",
                         abundance.var = "n",
                         replicate.var = "taxa",
                         metric = "appearance")

portal.disappearance <- turnover(df = portal.species.abunds,
                         time.var = "year",
                         species.var = "taxon",
                         abundance.var = "n",
                         replicate.var = "taxa",
                         metric = "disappearance")

portal.turnover <- full_join(portal.total, portal.disappearance) %>%
  full_join(portal.appearance)
```

```
## Joining, by = c("year", "taxa")
## Joining, by = c("year", "taxa")
```

```
portal.turnover <- gather(portal.turnover, key = metric, value = turnover,
                          total, appearance, disappearance)

turn.plot <- ggplot(
  portal.turnover, aes(x=year, y=turnover, color=metric)) +
  geom_line(size = 1, show.legend = T) +
  facet_wrap(~taxa, ncol = 1)+
  xlim(1977,2002)+
  xlab("Year")+
  ylab("Turnover")+
  theme(legend.position="bottom")+
  scale_color_grey()
plot(turn.plot)
```



Question 5:

a. How does temporal turnover relate to spatial turnover?
b. Which taxonomic group appears to be the most variable? Which group appears to be the least variable?

> ***Answer 5a***: The temporal turnover that we detect could be the result of spatial turnover occuring across the sampling area. In other words, the effects of spatial and temporal turnover are not entirely independent.

> ***Answer 5b***: Birds appear to be the most variable in their turnover rates, but reptiles have the highest sustained total turnover rates of all the taxa (though there appears to be limited data). Rabbits have a sustained total turnover rate of 0 and rodents maintain a low, but variable, turnover rate across the years sampled.

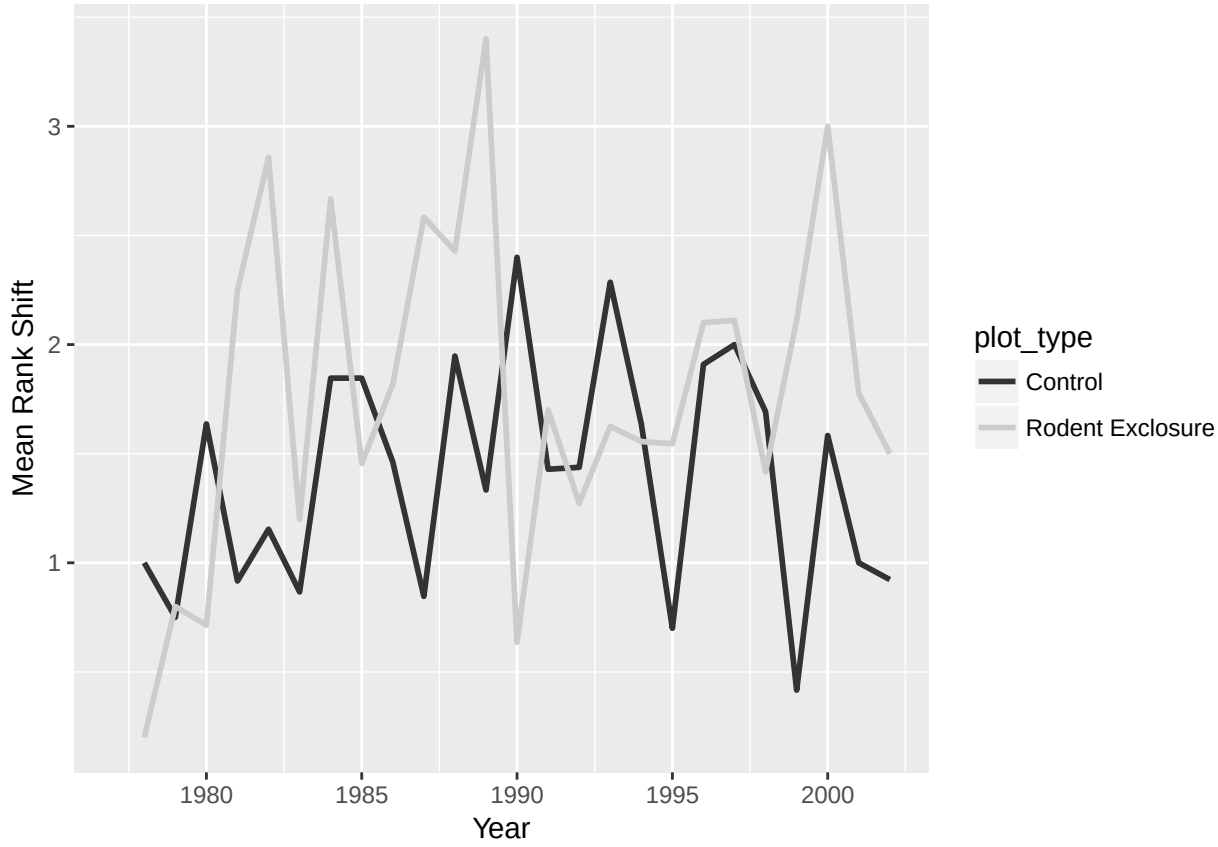
Mean Rank Shift

In the code chunk below, do the following:

1. Choose two plot_types or two plot_ids and compare the mean rank shift (MRS) between them.
2. Plot MRS for each through time.

```
portal.species.abunds <- group_by(portal, year, plot_type) %>% count(taxon)

portal.abunds.cont.rodent <- filter(portal.species.abunds,
```

```
                                    plot_type == "Control" | plot_type == "Rodent Exclosure")

portal.rankshift <- rank_shift(
  df = as.data.frame(portal.abunds.cont.rodent),
  time.var = "year",
  species.var = "taxon",
  abundance.var = "n",
  replicate.var = "plot_type"
)


portal.rankshift$year <- as.numeric(substr(portal.rankshift$year_pair, 6, 9))

rankshift.plot <- ggplot(portal.rankshift, aes(x = year, y = MRS, color = plot_type))+
  geom_line(size = 1) +
  xlim(1977, 2002)+
  xlab("Year")+
  ylab("Mean Rank Shift")+
  scale_color_grey()

plot(rankshift.plot)
```




```
group_by(portal.rankshift, plot_type) %>%
  summarise(
    mean = mean(MRS),
```

```
    cv = sd(MRS/mean)
  )
```

```
## # A tibble: 2 × 3
##          plot_type     mean        cv
##              <chr>    <dbl>     <dbl>
## 1          Control 1.400675 0.3759483
## 2 Rodent Exclosure 1.788980 0.4361809
```

Question 6:

a. What does a change in the rank shift tell you about the community?
b. Interpret the analysis and figure you just made.

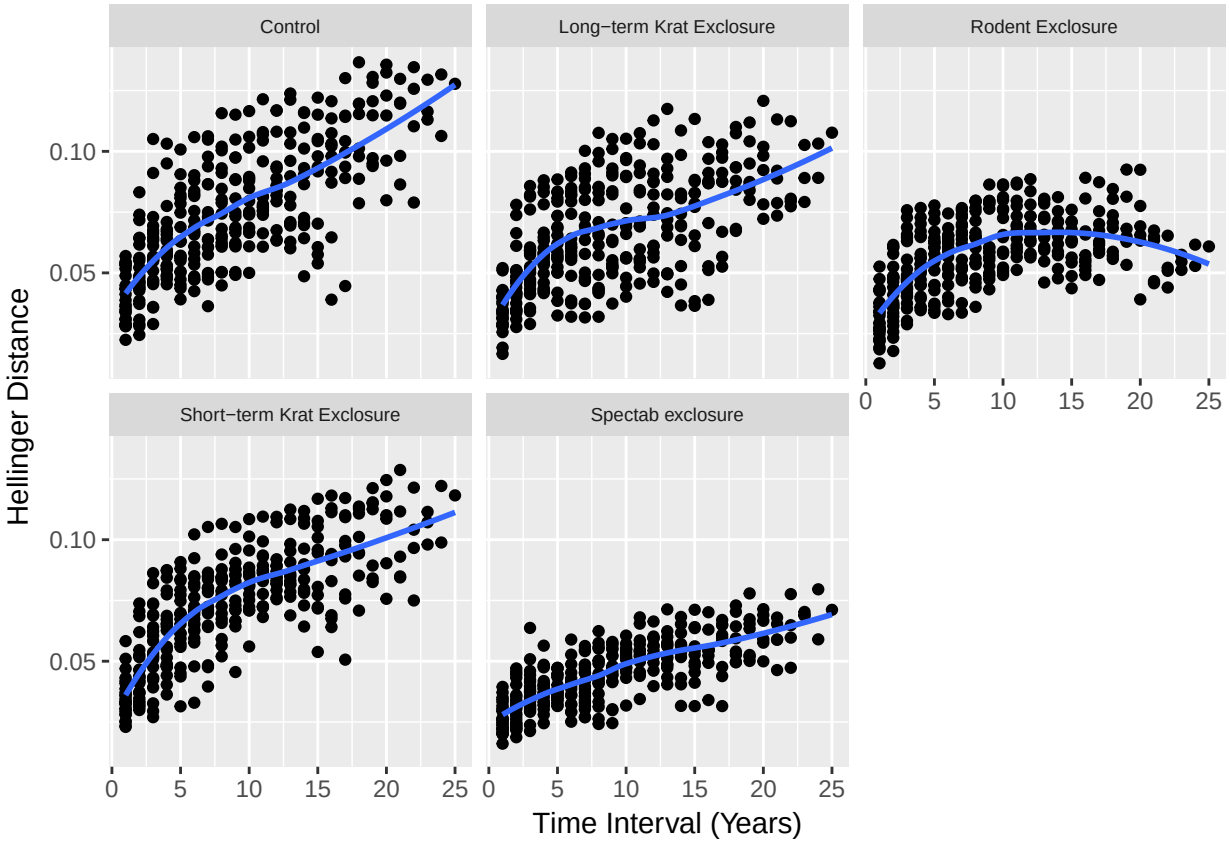
> ***Answer 6a***: A change in rank shift would suggest that there have been changes in species-specfic rank abundances. That certain species have become rarer and others more common. ***Answer 6b***: The mean rank shift for the `Rodent Exclosure` treatment is in most years higher than the `Control` treatment suggesting that the rodent exclosures contribute to changes in species composition over time due to shifts in species abundance. Mean rank shift is also more variable for the rodent exclosures as can be seen by the elevated value for the relative standard deviation (or CV). The rodent exclosures may be contributing to more drastic shifts in species abundance and turnover.

Rate Change Interval

In the R code chunk below, do the following:

1. Calculate the rate change interval using the Hellinger distance.
2. Plot the results.

```
portal.species.abunds$tot.abund <- rep(sum(portal.species.abunds$n),
                                       length(portal.species.abunds$n))


portal.hellinger.transf <- portal.species.abunds %>%
  mutate(hellinger.transf = sqrt(n / tot.abund))

portal.change.int <- rate_change_interval(portal.hellinger.transf,
                                          time.var = "year",
                                          species.var = "taxon",
                                          abundance.var = "hellinger.transf",
                                          replicate.var = "plot_type")

rate.plot <- ggplot(portal.change.int, aes(interval, distance)) +
  geom_point()+
  facet_wrap(~plot_type)+
  theme(strip.text.x = element_text(size = 7))+
  stat_smooth(method = "loess", se = F, size = 1)+
  ylab("Hellinger Distance")+
  xlab("Time Interval (Years)")
rate.plot
```



Question 7:

a. What does it mean to calculate a distance metric across varying time intervals?
b. Interpret the overall results. Develop a hypothesis based on the different responses of each treatment.

> ***Answer 7a***: It means that we are quantifying if and how quickly communities diverge over time.
> ***Answer 7b***: First, the `control` treatment has the quickest rate of community divergence based on the "slope" of the loess line. This relationship is also very linear. The treatments `Long-term Krat Exposure`, `Short-term Krat Exposure`, and `Spectab exclosure` maintain the linear trend yet have overall reduced rates of divergence. The slowest rate of divergence was for the `Spectab exclosure`. The `rodent exclosure` rate change interval plot has a peculiar parabolic shape. I hypothesize that the exclosures, by reducing overall diversity in the plots, make for a more stable community over time. Specifically, given that the `Spectab exclosure` treatment reduced the rate of divergence the most, I would hypothesize that Banner-tailed kangaroo rats have a strong influence on community divergence over time. Why this is the case, I cannot say.

7) STABILITY

In the R code chunk below, do the following:

1. Using total abundance as your focal variable, calculate stability (i.e., 1/CV) and synchrony for each plot type.
2. Test for a biodiversity-stability relationship by regressing community stability on mean richness.
3. Test for a biodiversity-stability relationship by regressing community stability on mean inverse Simpson's diversity.

```
portal.stab <- community_stability(df = as.data.frame(portal.species.abunds),
                                   time.var="year",
                                   abundance.var = "n",
                                   replicate.var = "plot_type")
pander(portal.stab)
```

plot_type	stability
Control	3.044
Long-term Krat Exclosure	1.865
Rodent Exclosure	1.864
Short-term Krat Exclosure	2.462
Spectab exclosure	2.911

```
time.by.species <- group_by(portal, year, plot_id,
                            plot_type) %>% count(taxon) %>% spread(key = taxon, value = n, fill = 0)

richness <- as.data.frame(rowSums(time.by.species[,-c(1:3)] > 0 ))


rich.all_R <- data.frame(time.by.species[,1:3,], richness)
names(rich.all_R)[4] <- "richness"


###SNYCHRONY
portal.loreau <- synchrony(df = as.data.frame(portal.species.abunds),
                           time.var = "year",
                           species.var = "taxon",
                           abundance.var = "n",
                           replicate.var = "plot_type",
                           metric = "Loreau")

names(portal.loreau)[2] <- "loreau"

portal.gross <- synchrony(df = as.data.frame(portal.species.abunds),
                          time.var = "year",
                          species.var = "taxon",
                          abundance.var = "n",
                          replicate.var = "plot_type",
                          metric = "Gross")
names(portal.gross)[2] <- "gross"

pander(full_join(portal.loreau, portal.gross))
```

```
## Joining, by = "plot_type"
```

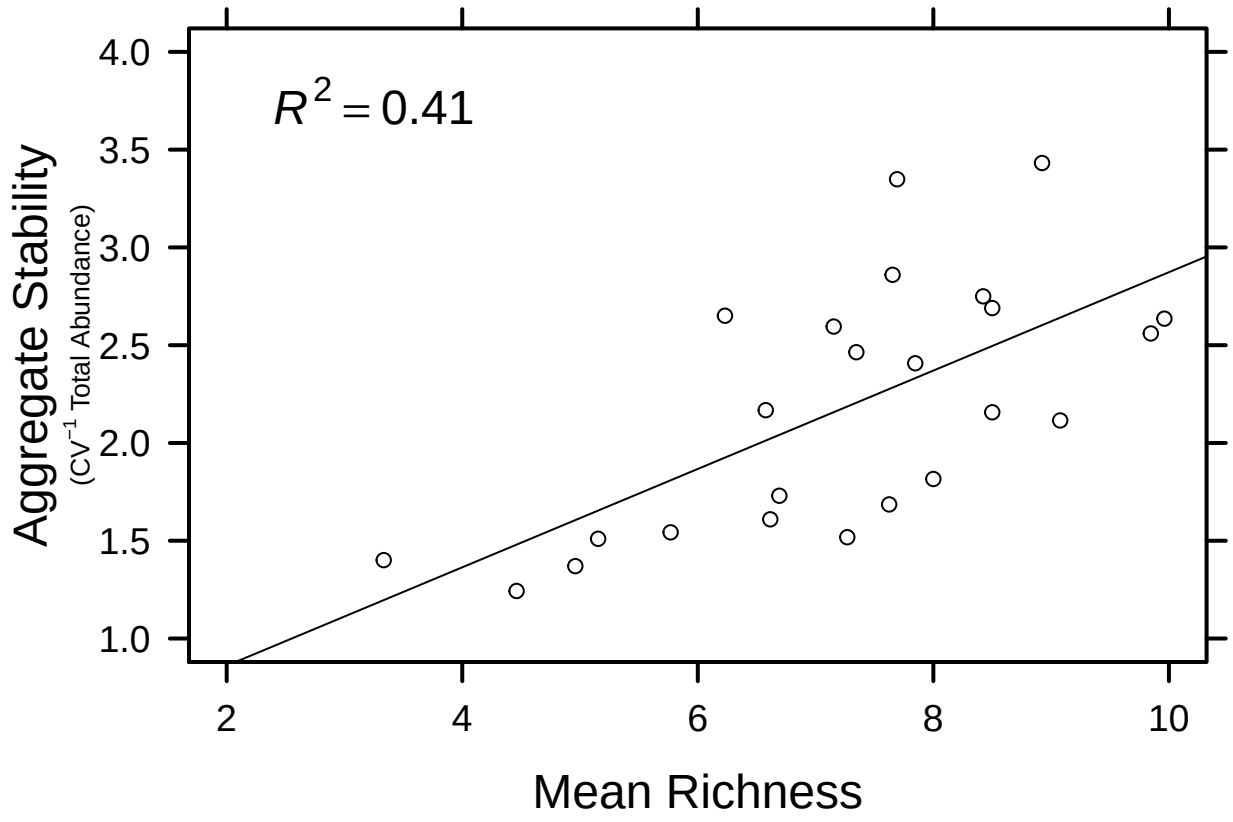
plot_type	loreau	gross
Control	0.1869	0.1263
Long-term Krat Exclosure	0.1578	0.07418
Rodent Exclosure	0.187	0.1423
Short-term Krat Exclosure	0.08773	0.001546

plot_type	loreau	gross
Spectab exclosure	0.1542	0.1393

```
####FOR RICHNESS

portal.mean.rich.plot <- rich.all_R %>%
  group_by(plot_id) %>%
  summarise(mean.rich = mean(richness))

portal.plot.abunds <- as.data.frame(
  group_by(portal, year, plot_id) %>% count(taxon))
portal.stab.plot <- community_stability(df = portal.plot.abunds,
                                        time.var = "year",
                                        abundance.var = "n",
                                        replicate.var = "plot_id")

portal.div.stab <- portal.mean.rich.plot %>%
  inner_join(portal.stab.plot)
```

```
## Joining, by = "plot_id"
```

```
par(mar = c(5,5,1,1))
plot(portal.div.stab$stability ~ portal.div.stab$mean.rich,
      xlab = "", ylab = "", yaxt = "n", xaxt = "n",
     xlim = c(2,10), ylim = c(1,4))

axis(side = 1, cex.axis = 1.2, lwd.ticks = 2, las = 1)
axis(side = 2, cex.axis = 1.2, lwd.ticks = 2, las = 1)
axis(side = 3, lwd.ticks = 2, las = 1, labels = F)
axis(side = 4, lwd.ticks = 2, las = 1, labels = F)
box(lwd = 2)
mtext("Mean Richness", side = 1, line = 3, cex = 1.5)
mtext("Aggregate Stability", side = 2, line = 3.5, cex = 1.5)
mtext(expression(paste("(CV"^"-1"," Total Abundance)")),
      side = 2, line = 2.25, cex = .8)

div.stab.lm <- lm(portal.div.stab$stability ~ portal.div.stab$mean.rich)
abline(div.stab.lm)
r2 <- bquote(italic(R)^2 == .(format(
  summary(div.stab.lm)$adj.r.square, digits = 3)))

text(3.25, 3.75, cex = 1.5, labels = r2)
```

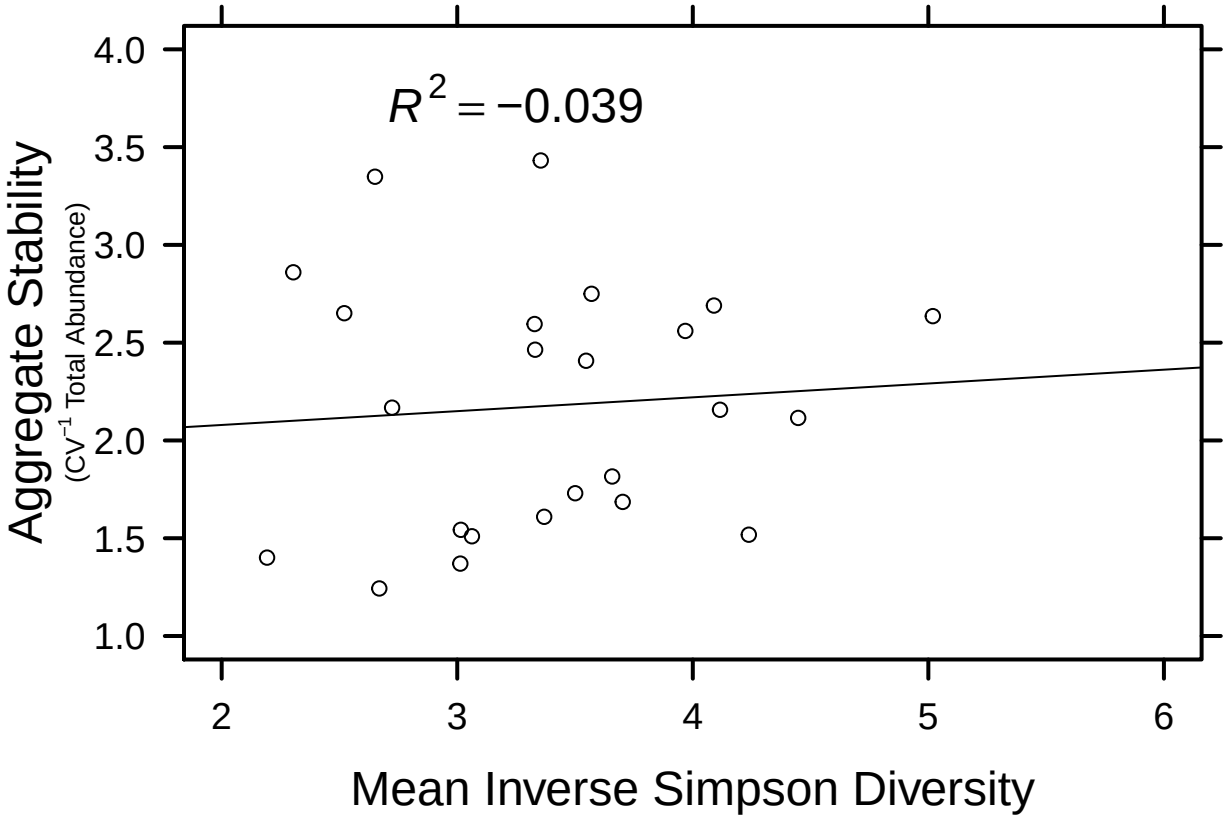



```
#####FOR INV SIMP

portal.mean.rich.plot <- rich.all %>%
  group_by(plot_id) %>%
  summarise(mean.rich = mean(inverse_S))

portal.plot.abunds <- as.data.frame(
  group_by(portal, year, plot_id) %>% count(taxon))
portal.stab.plot <- community_stability(df = portal.plot.abunds,
                                        time.var = "year",
                                        abundance.var = "n",
                                        replicate.var = "plot_id")

portal.div.stab <- portal.mean.rich.plot %>%
  inner_join(portal.stab.plot)
```

```
## Joining, by = "plot_id"
```

```
par(mar = c(5,5,1,1))
plot(portal.div.stab$stability ~ portal.div.stab$mean.rich,
      xlab = "", ylab = "", yaxt = "n", xaxt = "n",
     xlim = c(2,6), ylim = c(1,4))

axis(side = 1, cex.axis = 1.2, lwd.ticks = 2, las = 1)
axis(side = 2, cex.axis = 1.2, lwd.ticks = 2, las = 1)
```

```
axis(side = 3, lwd.ticks = 2, las = 1, labels = F)
axis(side = 4, lwd.ticks = 2, las = 1, labels = F)
box(lwd = 2)
mtext("Mean Inverse Simpson Diversity", side = 1, line = 3, cex = 1.5)
mtext("Aggregate Stability", side = 2, line = 3.5, cex = 1.5)
mtext(expression(paste("(CV"^"-1"," Total Abundance)")),
      side = 2, line = 2.25, cex = .8)

div.stab.lm <- lm(portal.div.stab$stability ~ portal.div.stab$mean.rich)
abline(div.stab.lm)
r2 <- bquote(italic(R)^2 == .(format(
  summary(div.stab.lm)$adj.r.square, digits = 3)))

text(3.25, 3.75, cex = 1.5, labels = r2)
```



Question 8:

a. Which plot type has the highest stability in total abundance? How is stability of total abundance measured with the function you learned? How does this measure of stability relate to the coefficient of variation?

b. In your own words, describe the concept of synchrony

c. Interpret the results from the biodiversity-stability relationships you analyzed.

Answer 8a: The `Control` plot type has the highest stability. Stability is measured as the inverse of the coefficient of variation (CV) where CV = (stdev/mean). The function we learned takes the

species abundance matrix, calculates the mean and standard deviation for abundance across sites of the same type, calculates the CV, and then takes the inverse. ***Answer 8b***: Synchony measures if and how species densities (for the same environment or treatment) are correlated over time. If they are strongly correlated, this may indicate that the species as a whole will respond to changes in the environment together and that they may be susceptible to environmental disturbances. ***Answer 8c***: Regressing stability on mean richness verified the prediction of biodiversity-stability theory. Increased richness was associated with increased stability (R2 = 0.41). Conversely, when regressing stability on inverse of Simpson's diversity, we do not see this relationship. Although the plot looks like it has a slightly positive slope, R2 = -.039. Simpsons diversity takes into account richness AND evenness, so using this metric may provide a better indication of the true biodiversity-stability relationship.

SYNTHESIS

Compare and contrast the core concepts from temporal and spatial diversity (e.g., autocorrelation, scale, variability, etc.). Identify a few of the major challenges associated with studying biodiversity through time and across space.

Answer: ***Autocorrelation in time and space.*** Spatial autocorrelation refers to the degree to which sites that are close to one another in space have similar trait values. Similarly, temporal autocorrelation is the degree to which a variable at one time point is correlated with the same variable at a later (or earlier) time point. ***Scale:*** Having the appropriate scale in spatial studies is critical. We want to have a large enough extent to capture the the dynamics of the entire range but we also want a small snough grain to resolve significant patterns. While not explicitly mentioned, I feel the same could be said for temporal studies of diversity. We want a large extent (over many years) and a small grain (sampling every month as opposed to every year) in order to capture the effects of seasonality and other imporant attributes that may affect diversity across time. ***Variability:*** Variation in diversity across space can broadly be grouped into three categories: environment, space, and environment+space. In the Beta-Diversity assignment, we learned how to partition this variation to understand the relative contributions of each to the observed patterns of diversity. Studying diversity temporally allows us to quantify how diversity varies through time. We can visualize this variation, formally test for effects of other variables while accounting for non-independence through time, build models that attempt to capture temporal variation or seasonal patterns, study variation in species turnover through time, look at species divergence by site through time, as well as measure stability through direct quantification of variation through time (CV, synchrony, variance ratios, biodiversity-stability relationship). A major challenge in studying spatial diversity is understanding the relative contributions of geographic distance and environmental divergence on species distributions. Often times these two factors are confounded. In other words, disentangling isolation-by-environment and isolation-by-distance to arrive at a formal conclusion on which is driving patterns of observed diversity is rather tough even if we can partition the variation. Two major challenges in studying temporal patterns of diversity is 1) accounting for non-independence of observations in longitudinal studies and 2) building acurate forecasting models to describe patterns of diversity into the future. The ARIMA model that I built did not do that great of a job capturing and forecasting the large amount of variance from year to year in rodent abundance. It is likely that the model significantly underestimated the variance.