By Ashley Shade, for use by EDAMAME workshop students at Michigan State University in August 2014

Tutorial Day 4

Day 4. Analyzing community patterns: Hypothesis testing, identifying gradients, and linking environmental and community data

Software: R

1. Hypothesis testing

Welcome back!
Open R studio and change directory to the analysis folder that you used yesterday and load the vegan package. Use the provided scripts as a guide and reminder as to how to perform the following tasks:

a. Re-read in the UniFrac resemblance matrices. Tell R that these are resemblance matrices using **as.dist()**.
b. Read in the Bray-Curtis and Sørenson resemblance matrices. Tell R that these are resemblance matrices.

We will go through the hypothesis tests with the Bray-Curtis resemblance, and then you can go then you will work through the remaining Sørenson, weighted UniFrac, and unweighted UniFrac on your own. Make a table of results and compare across tests and across resemblance metrics.

A. PERMANOVA (permuted analysis of variance)
The vegan function **adonis()** performs permutated analysis of variance to test for differences (in centroid and/or dispersion) between treatment groups. The algorithm acts on the resemblance matrix, and links it to Treatment groups through the map file. We set it up exactly like the ANOVA, and the output is an ANOVA table, which we call a.table:

```
> ad=adonis(braycurtis.d~Treatment, data=map, permutations=999)
> a.table=ad$aov.tab

Terms added sequentially (first to last)

          Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
Treatment  5    2.8596 0.57192  3.2621 0.18898  0.001 ***
Residuals 70   12.2726 0.17532         0.81102
Total     75   15.1322                 1.00000
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Inspect a.table. Is the global effect of Treatment significant?

B. Permuted multivariate analysis of beta-dispersion (PERMDISP)

The vegan function "**betadisper()"** performs PERMDISP. This test is different from the others in that it specifically tests for differences in the spread (dispersion, variability) among groups. Therefore, if you use this test in combination with one of the other three, you will be able to tease apart whether groups of communities are different because they have different centroids or different spreads. For example, if PERMANOVA yields a significant difference, but PERMDISP does not, you can safely say that the distinction between groups can be attributed to differences in their centroid. Ordinations are often a good way to visually support and summarize these findings.

```
> b=betadisper(braycurtis.d, group=map[,"Treatment"], type="median")
> b
> b.perm=permutest(b, group=map[,"Treatment"], type="median", permutations=999,
pairwise=TRUE)
> b.perm

Permutation test for homogeneity of multivariate dispersions

No. of permutations: 999

**** STRATA ****
Permutations are unstratified

**** SAMPLES ****
Permutation type: free
Mirrored permutations for Samples?: No

Response: Distances
          Df Sum Sq  Mean Sq      F N.Perm Pr(>F)
Groups     5 0.6927 0.138549 2.6926    999  0.031 *
Residuals 70 3.6018 0.051455
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Pairwise comparisons:
(Observed p-value below diagonal, permuted p-value above diagonal)
          Eggs       LGG        Lp        Lr       MRS   PBS
Eggs           0.1810000 0.0040000 0.0070000 0.0990000 0.332
LGG  0.1654204           0.1540000 0.0370000 0.4450000 0.653
Lp   0.0011242 0.1452157           0.3120000 0.5860000 0.061
Lr   0.0019757 0.0308457 0.3013399           0.2380000 0.016
MRS  0.0921442 0.4453272 0.6122409 0.2151416           0.239
PBS  0.3154924 0.6460088 0.0678813 0.0134611 0.2474733
```

Fill in the chart as you go:

	Sørenson	Bray-Curtis	Weighted UniFrac	Unweighted UniFrac
PERMANOVA	R^2 = , p =	R^2 = **0.19 , p = 0.001**	R^2 = , p =	R^2 = , p =
PERMDISP	Global p = List significant pairwise differences (based on permuted p < 0.10):	Global **p = 0.31** List significant pairwise differences (based on permuted p < 0.10): **Eggs v. Lp** **Eggs v. Lr** **Eggs v. MRS** **LGG v. Lr**	Global p = List significant pairwise differences (based on permuted p < 0.10):	Global p = List significant pairwise differences (based on permuted p < 0.10):

		Lp v. PBS **Lr v. PBS**		

2. How reproducible are replicates?

In answering this question, we will also introduce how to use loops in R. Loops are a useful way to prevent yourself from doing tedious, repetitive calculations. Here, we write a loop to examine the reproducibility across replicates. Details for each step in the loop are provided in the R script. In the end, we plot the results to find that there is quite a bit of variability between some of the replicate samples.

```
> u=unique(map[,"Replicates"])

> meanreps.out=NULL
> for(i in 1:length(u)){
  temp=braycurtis[map[,"Replicates"]==u[i],map[,"Replicates"]==u[i]]
  temp.d=as.dist(temp)

  m=mean(temp.d)
  meanreps.out=c(meanreps.out,m)
}

> names(meanreps.out)=u

> hist(meanreps.out)

> meanreps.out
    Eggs_0      PBS_3      PBS_4      PBS_5      LGG_1      LGG_2      LGG_3
LGG_4
0.15176152 0.15756872 0.07820364 0.53735966 0.66976384 0.15292296 0.15989160
0.23499806
     LGG_5       Lp_1       Lp_2      PBS_1       Lp_3       Lp_4       Lp_5
Lr_1
0.36972513 0.13124274 0.50135501 0.54587689 0.35578784 0.29384437 0.37746806
0.41385985
      Lr_2       Lr_3       Lr_4       Lr_5      MRS_1      MRS_2      PBS_2
MRS_3
0.50522648 0.37166086 0.53929539 0.57955865 0.11304684 0.14866434 0.10840108
0.29307007
     MRS_4      MRS_5
0.55013550 0.30855594
```

3. Linking environmental gradients to community patterns

A. Making a resemblance matrix of time (or space).

I have written a custom R function to create a time/space/environmental matrix to correlate to changes in the communities. It is called **makeTimeDist.f**. The **.f** signifies that this is a function and will require certain arguments to work. Inspect the syntax of the function in the provided R script, but don't worry too much about the mechanics of it now:

```
makeTimeDist.f=function(map_file){
  map=map_file
  temp=as.matrix(map[,"Instar"])
  names(temp)=map[,"SampleID"]
  temp.d=dist(temp, method="manhattan", diag=FALSE)
  head(temp.d)
  temp.out=as.matrix(temp.d)
  colnames(temp.out)=map[,"SampleID"]
  row.names(temp.out)=map[,"SampleID"]
  print(head(temp.out))
  return(temp.d)
}
```

The important part is to be able to use the function. The argument for the function is **map_file**. Thus, you should provide your map file as input. We also need to name the output of the function, here we've named it **time.d** so that it is clear that it is a distance matrix.

```
> time.d=makeTimeDist.f(map)
```

Inspect the head of this time matrix, provided automatically by the function. Do the values make sense with what you know about the instars for each sample?

Now, we will use the non-parametric Mantel test **"mantel()"** to determine if changes in time correlate with community patterns.

```
> mantel(time.d,braycurtis.d, method="pearson", permutations = 999)

Mantel statistic based on Pearson's product-moment correlation

Call:
mantel(xdis = time.d, ydis = braycurtis.d, method = "pearson",      permutations =
999)

Mantel statistic r: 0.248
      Significance: 0.001

Empirical upper confidence limits of r:
   90%    95%  97.5%    99%
0.0549 0.0708 0.0826 0.1062

Based on 999 permutations
```

From this test, we can say with confidence that time provides some explanatory value in describing the differences in communities, as the test statistic is 0.24, permuted $p < 0.001$. We can perform the Mantel test using our other resemblances and compare:

```
> mantel(time.d,sorenson.d, method="pearson", permutations = 999)
> mantel(time.d,weighted_u.d, method="pearson", permutations = 999)
> mantel(time.d,unweighted_u.d, method="pearson", permutations = 999)
```

From these tests, we see that our temporal patterns are more apparent using metrics that account for relative abundances of taxa (weighted UniFrac and Bray-Curtis). This suggests that

the changes over time likely are due to changes in relative abundances of taxa rather than in replacement of membership through time.

B. Correspondence analysis

We can also correlated changes in time with changes along an ordination axis of communities. This is sometimes desirable, as the most variability in communities is often explained by the first and second axes of ordination analyses. By focusing just on the variability along these two axes, we may be able to uncover clear patterns.

Use the **"cca()"** and **"envfit()"** functions to link instar data (a proxy for time) to the communities.

```
> otu.ca=cca(t(otu))
> plot(otu.ca)
> ev.instar=envfit(otu.ca, as.numeric(map[,"Instar"]), perm=1000)
> ev.instar

***VECTORS

         CA1     CA2     r2   Pr(>r)
[1,] 0.70975 0.70445 0.1094 0.006993 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
P values based on 1000 permutations.


> plot(ev.instar)
```

These results again support that there is an influence of time.

4. Testing for time-treatment interactions.

Now that we have evidence that there are differences between treatments (evidenced by PERMANOVA and PERMDISP global differences), and that there are correlations with time, we want to go back and use the **adonis()** function to perform a PERMANOVA to test for a time-treatment interaction.

```
> ad2=adonis(braycurtis.d~Treatment*as.numeric(Instar), data=map, permutations=999)
> a.table2=ad2$aov.tab
> a.table2
Terms added sequentially (first to last)

                           Df SumsOfSqs MeanSqs F.Model      R2 Pr(>F)
Treatment                   5    2.8596 0.57192  3.9656 0.18898  0.001 ***
as.numeric(Instar)          1    1.6457 1.64575 11.4113 0.10876  0.001 ***
Treatment:as.numeric(Instar)  4    1.2525 0.31313  2.1712 0.08277  0.014 *
Residuals                  65    9.3743 0.14422         0.61949
Total                      75   15.1322                 1.00000
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

These results show a significant influence of both time (instar) and treatment.