• Everyone had slightly different results…rarefy will produce different results each time.

• Richness declined by 30-40%, which species were dropped?

• Bray Curtis dissimilarity dropped from 0.46 to 0.27. Most of the increase in similarity came from controlling abundance of common species, not the loss of rare species.

• Use a qualitative metric to look at changes in species composition

• Mean steinhaus dissimilarity: raw 0.653 vs. 0.506 rarefied

• What if we retain only the 60 most abundant OTU (dropping 82 rare OTU)?

  • Mean Bray Curtis
    - Raw 0.4614
    - Rarefied 0.2649

  • Mean Steinhaus
    - Raw 0.6070
    - Rarefied 0.4774

• So, eliminating almost all of the rare species did not change Bray Curtis at all. Steinhaus similarity increased, and the difference between raw and rarefied decreased.
• Doing 1000 rarefactions of the data as described…results are essentially identical to what is in the paper.

Distance matrix analyses

What can you do with a large triangular similarity matrix?
Recall that each element in the triangular matrix is a similarity measure between two objects (samples in this case).
What if your hypothesis is that two groups of samples are different?

Earlier approaches
• Can be used as descriptive statistics with subjective levels of "similar" or "not similar".
• Next week – cluster analyses and dendrograms. Visual but also largely descriptive.
• Future classes – ordinations (also descriptive!)
• Traditional statistical methods are not appropriate due to a lack of independence among samples.
Three techniques to test for these patterns in a similarity matrix.

- **ANOSIM** – Analysis of similarity
- **MRPP** – Multi response permutation procedure
- **Multivariate Analysis of Variance for Distance Matrices**
- **Analysis of Molecular Variation (AMOVA)**

These all work for any application where you can calculate some distance among units and want to look for differences among groups of units.

- Eg. morphological data, spatial layout data, behavioral data, molecular data

### ANOSIM

- Dissimilarity matrix divided up by grouping variable(s)
- Between groups vs among comparisons of interest
  - Rank all similarities
  - If groups are more similar than random (null), then mean rank similarity within a group should be less than among groups.

#### ANOSIM

- Calculate observed R statistic
  - Ranges from -1 to 1
  - 0 = no pattern
  - 1 = all within group ranks less than among group ranks
  - -1 all within group ranks greater than among groups

\[
R = \frac{(B - W)}{N(N - 1)}
\]

- B – mean between group ranks
- W – mean within group ranks
- N – number of samples

Need variable (factor) grouping your data

- Load the data as you normally would.
- You will need to separate the community data from the variables describing your groups.

```r
# Load data
sample_data <- read.csv("sample2.csv", row.names = 1)
year <- as.factor(sample_data$year) # subset
community <- subset(sample_data, select = c(year, creek))
```

- This code yields four objects
  - Sample_data: everything in the original data file
  - year and creek: variables with year and creek factors
  - Community: community matrix without factors

Test of significance

- How do you test the significance of the R value?
  - “What is the probability of obtaining an R value equal to or greater than the observed?”
  - Permutation test:
    - Randomize the observed data, calculate R, repeat
    - Compare observed R to distribution of randomized R
  - Use caution in how you set up your randomization!

**ANOSIM (vegan package)**

- **R Code:**
  - `community_dist <- vegdist(community_proportion, method = 'bray')`
  - `year_ano <- anosim(community_dist, year, permutations = 5000)`
  - `summary(year_ano)`
  - `plot(year_ano)`
  - `hist(year_ano$perm)`

- **Options**
  - # permutations
  - Strata (block groups for permutations)
  - Requires dissimilarity matrix
  - You can either supply a dissimilarity (triangular) matrix, or community matrix and specify what kind of dissimilarity to produce (method =)
  - Parallel – specify how many cores to use in parallel

- **Output**
  - Observed R, vector of permuted Rs, significance of observed R
ANOSIM

- Can use any similarity measure on virtually any data
- Converts to rank similarity (information loss)
- Two way designs can be tested (see Primer ver. 6)
- Use of ranks means differences in the amount of variability (dispersion) not detected
- Pairwise tests often not built in (but see Primer)
- Often done in conjunction with Non Metric Multidimensional scaling (NMDS) ordination

ANOSIM output

<table>
<thead>
<tr>
<th>Dissimilarity: bray</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANOSIM statistic R: 0.4576</td>
</tr>
<tr>
<td>Significance: 0.001</td>
</tr>
<tr>
<td>Based on 999 permutations</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Between</th>
<th>2007</th>
<th>2008</th>
<th>2009</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Histogram of year_ano$perm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Frequency</td>
</tr>
<tr>
<td>0.00</td>
</tr>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Mean of rank similarities within each group and between all groups.</th>
</tr>
</thead>
<tbody>
<tr>
<td>tapply(year_ano$dis.rank,year_ano$class.vec,mean)</td>
</tr>
</tbody>
</table>

ANOSIM

- From the writer of the procedure in R: “I don’t quite trust this method. Somebody should study its performance carefully. The function returns a lot of information to ease further scrutiny.”

- Very popular technique. However, there are more powerful alternatives.

Multiple Response Permutation Procedure (MRPP)

- Conceptually similar to ANOSIM but does not use ranks
- Calculated statistic is $\Delta = $ weighted average within group similarity
- Permutations – randomize matrix, recalculate $\Delta$
- Significance of $\Delta$ assessed by distribution of permuted $\Delta$ scores.
- Code
  ```
  community_dist <- vegdist(community_proportion, method='bray')
  year_mrpp <- mrpp(community_dist, year, permutations=5000)
  ```
- Options
  - As in ANOSIM - permutations, strata
  - Weights – three methods for weighting sample size for $\Delta$ calculation

MRPP

- Avoiding ranks means
  - MRPP can detect differences in mean as well as dispersion.
  - Biases in distance metric will be more pronounced than in ANOSIM
- Output
  - Observed $\Delta$
  - Vector of permuted $\Delta$ and expected $\Delta$
  - Significance of observed $\Delta$
  - $A$ - estimate of the proportion of distances explained by the factor

Multivariate Analysis of Variance Using Distance Matrices

- Recently described, generally superior to ANOSIM in all ways, MRPP in most ways
- Robust alternative to MANOVA, sometimes called permutational MANOVA
- Using this technique with one dependent variable and a Euclidean distance matrix should yield same results as a traditional ANOVA
- Also similar to AMOVA

Multivariate Analysis of Variance Using Distance Matrices

- SS$_t$ = sum of squared differences between all observations and the overall centroid.
- SS$_w$ = sum of squared differences between group observations and group centroid.
- SS$_a$ = SS$_t$-SS$_w$
- Pseudo-F = ratio of SS$_a$ to SS$_w$
**Multivariate Analysis of Variance Using Distance Matrices**

- Should be applicable any time ANOSIM or MRPP can be used.
- Should be more robust than ANOSIM or MRPP
- Higher level ANOVA designs, including interactions

**Code:**

```r
permanova <- adonis(community_proportion ~ creek*year, method="bray", permutations=10000)
print(permanova)
```

**Options**

- Permutations
- Distance metric (procedure works with raw community dataset not a triangular matrix)
- Strata
- Model eg. community ~ site * time

**Output**

- Standard ANOVA table with % variance accounted for by each variable (factor) and the residuals (error)
- Observed and permuted pseudo-F
- Species coefficients for each level of each factor

| Df | Sum Sq Mean Sq F value    Pr(>F) |
|----|-------------------------|------------------|
| creek | 5  | 0.4985 0.0997 0.1387 0.13887 0.0304 * |
| year | 2  | 1.6113 0.8057 16.0804 0.44891 9.999e-05 *** |
| creek:year | 10 | 0.5778 0.0578 1.1533 0.16098 0.3073 |

**Number of permutations: 10000**

Terms added sequentially (first to last)

| Df | Sum Sq Mean Sq F value    Pr(>F) |
|----|-------------------------|------------------|
| creek | 5  | 0.4985 0.0997 0.1387 0.13887 0.0304 * |
| year | 2  | 1.6113 0.8057 16.0804 0.44891 9.999e-05 *** |
| creek:year | 10 | 0.5778 0.0578 1.1533 0.16098 0.3073 |
| Residuals | 18 | 0.9018 0.0501 0.25125 |
| Total | 35 | 3.5895 |

**A note about R models**

- There is standard R language for model formulas
- Assume the following variables
  - Y – dependent variable
  - X1, X2 – continuous independent variables
  - F1, F2 – discrete independent variables (factors)
    - Y~X
      - Linear regression
    - Y~F
      - Multiple regression without interaction
    - Y~F+F
      - Multiple regression with interaction
    - Y~F+F*F
      - Two factor ANOVA without interaction
    - Y~F+F+F
      - Two factor ANOVA with interaction

- See help.start() section 11 – Statistical Models in R

---

**AMOVA**

- Very similar to analysis of variance using distance metric (in fact, using a Euclidean distance matrix should yield identical results).
- First developed for analyzing mtDNA haplotypes
- Distance matrix was pairwise steps in a network
- Grouping variable was population

AMOVA

- Two amova functions, both named "amova"
- Package ade4
  - Haplotypic approach – requires 1) matrix of haplotype frequency by population, 2) Euclidian distance among haplotypes in netowrk, and 3) assignment of populations to groups
  - Separate function randtest tests significance through permutation.
- Package pegas
  - More general approach – requires genetic distance matrix and a factor.

Reading

- Sample script and dataset
- Papers
- Text: Chapter 3, information on similarity measures
- For information on formulas in R, review chapter 11 -help.start()