Ordination

Lecture 09.1: Distance Matrices

Lauren Sullivan

Module: Multivariate Models

Readings

Required for class:

- NA

Optional:

- Legendre, P. and Gallagher, E.D. (2001) Ecologically meaningful transformations for ordination of species data. *Oecologia.*
- Strecker, A. L. and Brittain, J. T. (2017) Increased habitat connectivity homogenizes freshwater communities: historical and landscape perspectives. *Journal of Applied Ecology.*

Multivariate Analysis

There are several ways to look at multivariate patterns from a matrix of $\mathbf{Y}$'s.

1. Linear models: MANOVA/regression to test patterns
2. **Ordination: PCA, nMDS, etc to visualize patterns**
3. Permutation tests: PERMANOVA to test patterns

Ordination

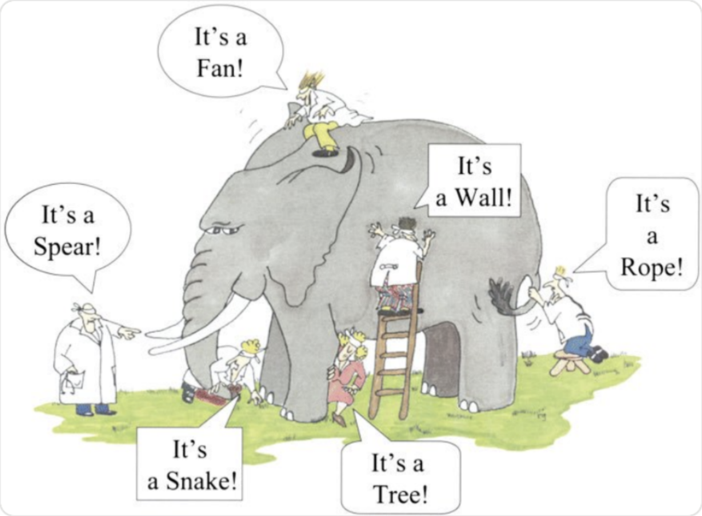
Use ordination techniques when you have a matrix of Y data and you want to explore the multi-dimensional aspects of the **Y**'s.

This type of data exploration is common in:

1. Community ecology (simultaneous response of multiple members within a community)
   - Composition of plants within quadrats
   - Composition of aquatic organisms within a sample
   - Composition of microbes in a sample (using genetic data)
2. Morphometrics
   - Complex shape of a sample (e.g. skull, limb, etc)
3. Chemical/Molecular makeup
   - Composition of metabolites within a tissue sample
   - Composition of proteins in a sample

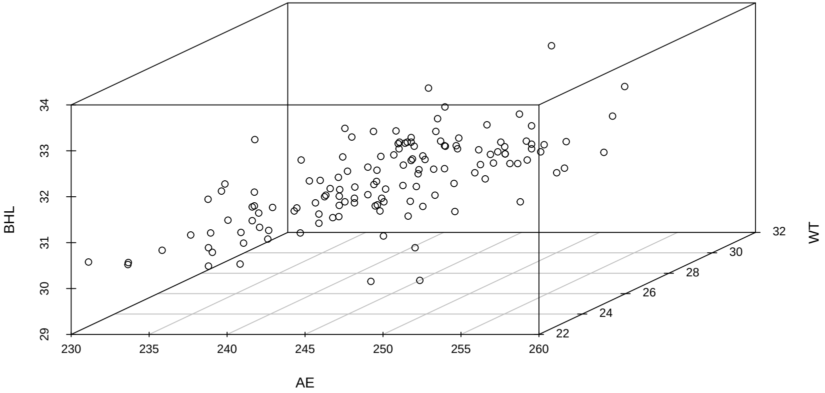
Pattern Description

Trying to describe the whole pattern of the data, not just a piece of the data

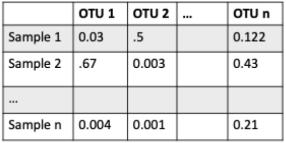


Pattern Description

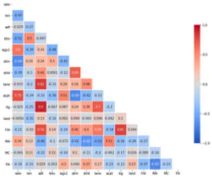
To try to understand the data in mult-dimensional space, we start by describing the "distance" between these data points using a distance matrix.



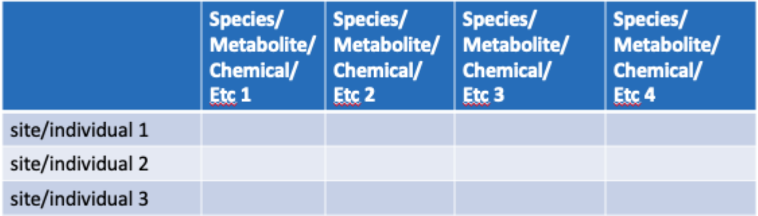
Data -> Distance -> Statistics



	OTU 1	OTU 2	...	OTU n
Sample 1	0.03	.5		0.122
Sample 2	.67	0.003		0.43
...				
Sample n	0.004	0.001		0.21



General Data Structure

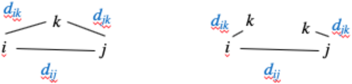


	Species/ Metabolite/ Chemical/ Etc 1	Species/ Metabolite/ Chemical/ Etc 2	Species/ Metabolite/ Chemical/ Etc 3	Species/ Metabolite/ Chemical/ Etc 4
site/individual 1				
site/individual 2				
site/individual 3				

To be able to translate this type of data into any sort of analysis, we need to figure out a way to relate each observation (row) to each other. So we use **Distance Matrices.**

Properites of Distance Measures

1. Minimum distance = 0.
   - This occurs when two observations have exactly the same composition
   - $Y_{ij} = Y_{ik}$. (i = species composition, k, j = sites)
2. If $Y_{ij} \neq Y_{ik}$, then $d_{jk} > 0$.
3. The distance between two sites is always symmetric.
   - $d_{ij} = d_{ji}$
4. Triangle inequality
   - If you have three sites, i, j, k, then $d_{ij} + d_{jk} \geq d_{ik}$

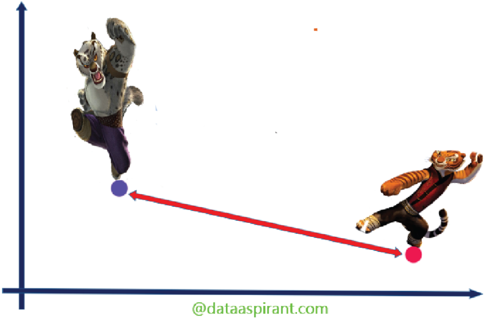


Metric measures satisfy all 4 criteria, semimetric measures (e.g. Bray Curtis) violate #4.

Types of Distance Measures

1. Euclidean Distance - as the crow flies

$$d_{jk} = \sqrt{\sum (y_{ij} - y_{ik})^2}$$

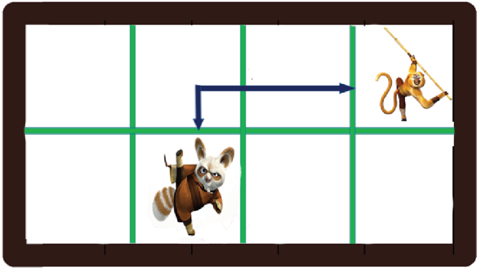


d_{jk} is the distance between samples j and k, and y_{ij} = abundance of species i in sample j.

Types of Distance Measures

2. Manhattan Distance - city block distance

$$d_{jk} = \sum |y_{ij} - y_{ik}|$$



@dataaspirant.com

d_{jk} is the distance between samples j and k, and y_{ij} = abundance of species i in sample j.

Types of Distance Measures

3a. Jaccard Distance - presence/absence (emphasizes rares)

$$d_{jk} = \frac{a+b}{a+b+c}$$

Set A =
Set B =
|A| = 4
|B| = 5
@dataaspirant.com

d_{jk} is the distance between samples j and k, a is the number of species *only* in sample j, b is the number of species *only* in sample k, and c is the number of species in both samples.

Types of Distance Measures

3b. Bray-Curtis Distance - empasizes rare species

$$d_{jk} = \frac{\sum |y_{ij} - y_{ik}|}{\sum (y_{ij} + y_{ik})}$$

@dataaspirant.com

d_{jk} is the distance between samples j and k, and $y_{ij} =$ abundance of species i in sample j.

Types of Distance Measures

4. Canberra - often used in metabolomics

$$d_{jk} = \frac{1}{\#\text{non-zero entries}} \sum \left(\frac{|y_{ij} - y_{ik}|}{y_{ij} + y_{ik}} \right)$$

@dataaspirant.com

d_{jk} is the distance between samples j and k, and $y_{ij} =$ abundance of species i in sample j.

Raw Data Transformation

We have our abundance data of multiple Y variables (e.g. species, metabolites, molecules, etc) per sample. But before we compute most types of distances, we need to standardize the data.

1. Convert abundance to some function of the abundance
   - presence/absence (0/1)
   - log() - common in metabolomics
   - (abundance)$^{1/4}$ - used when you have **very** skewed, patchy data with lots of zeros (e.g. aquatic invertebrates)
2. Convert to site proportion (i.e. divide by site totals)
   - common in community ecology with Bray-Curtis
3. Standardize by species maximum
   - equalizes contributions from rare and abundant species
4. Classic Wisconsin school (WI double standardization)
   - first by site total, then by site max

How to decide on transformation.

There are some things to think about when considering transformations.

1. How much do rare elements “count” vs abundant ones?
   - If you care about rare elements (e.g. species, metabolites, etc), consider presence/absense, dividing by site totals, or choose a measure with absolute values.
2. Does the total abundance matter?
   - If so, then use site max
   - If not, then divide by site totals

Toy Example

The data matrix

```
## # A tibble: 4 x 5
##       a     b     c     d     e
##   <dbl> <dbl> <dbl> <dbl> <dbl>
## 1    50    10     5     0     0
## 2    10     2     1     0     3
## 3    20     4     4     5     4
## 4    50     0     0     5     4
```

library(vegan) and decostand() for standardizing.

```
decostand(toy, "total")
```

```
##           a         b          c          d          e
## 1 0.7692308 0.1538462 0.07692308 0.00000000 0.00000000
## 2 0.6250000 0.1250000 0.06250000 0.00000000 0.18750000
## 3 0.5405405 0.1081081 0.10810811 0.13513514 0.10810811
## 4 0.8474576 0.0000000 0.00000000 0.08474576 0.06779661
```

```
decostand(toy, "log")
```

```
##          a        b        c        d        e
## 1 6.643856 4.321928 3.321928 0.000000 0.000000
## 2 4.321928 2.000000 1.000000 0.000000 2.584963
## 3 5.321928 3.000000 3.000000 3.321928 3.000000
## 4 6.643856 0.000000 0.000000 3.321928 3.000000
```

```
wisconsin(toy)
```

```
##           a         b         c         d         e
## 1 0.3333333 0.3333333 0.3333333 0.0000000 0.0000000
## 2 0.1481481 0.1481481 0.1481481 0.0000000 0.5555556
## 3 0.1111111 0.1111111 0.2222222 0.2777778 0.2777778
## 4 0.3333333 0.0000000 0.0000000 0.3333333 0.3333333
```

library(vegan) and vegdist() for distances.

Euclidean

```
vegdist(toy.t, "eucl")
```

```
##           1         2         3
## 2 0.2387444
## 3 0.2920831 0.1845628
## 4 0.2179070 0.3008810 0.3489082
```

Manhattan

```
vegdist(toy.t, "manhattan")
```

```
##           1         2         3
## 2 0.3750000
## 3 0.5488565 0.3614865
## 4 0.4615385 0.6144068 0.6138342
```

Canberra

```
vegdist(toy.t, "canberra")
```

```
##           1         2         3
## 2 0.3275862
## 3 0.5035491 0.3361651
## 4 0.8096774 0.7239918 0.5358911
```

`library(vegan)` and `vegdist()` for distances.

Jaccard (with presence/absence standardization)

```
vegdist(toy.p, "jaccard")
```

```
##      1    2    3
## 2 0.25
## 3 0.40 0.20
## 4 0.80 0.60 0.40
```

Bray-Curtis

```
vegdist(toy.t, "bray")
```

```
##           1         2         3
## 2 0.1875000
## 3 0.2744283 0.1807432
## 4 0.2307692 0.3072034 0.3069171
```

Bray-Curtis (with spp total standardization)

```
vegdist(toy.t, "bray")
```

```
##           1         2         3
## 2 0.1875000
## 3 0.2744283 0.1807432
## 4 0.2307692 0.3072034 0.3069171
```