Table 9.1 from Legendre and Legendre (1998)

<table>
<thead>
<tr>
<th>Method</th>
<th>Distance</th>
<th>Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>PCA</td>
<td>Euclidean</td>
<td>Quantitative, linear relationships assumed, beware of double-zeroes</td>
</tr>
<tr>
<td>PCoA</td>
<td>Any</td>
<td>Quantitative, qualitative or mixed</td>
</tr>
<tr>
<td>NMDS</td>
<td>Any</td>
<td>Quantitative, qualitative or mixed</td>
</tr>
<tr>
<td>CA</td>
<td>( \chi^2 )</td>
<td>Non-negative, quantitative or binary</td>
</tr>
<tr>
<td>FA</td>
<td>Euclidean</td>
<td>Quantitative, linear relationships assumed, beware of double-zeroes</td>
</tr>
</tbody>
</table>

Non Metric Multidimensional Scaling

- Most robust unconstrained ordination for community data
- Distance based
- Similar to metric multidimensional scaling (PCoA)
  - rank distances used instead (thus non metric)
  - Use of ranks aligns it well with ANOSIM
- Usually called NMDS, sometimes called MDS
- Iterative, not eigen based

**NMDS**

- **Goal of analysis** – place samples in \( k \) dimensional space to minimize differences between rank similarities in the original distance matrix and rank euclidean similarities in ordination space.

**NMDS**

- As with PCoA:
  - Need to specify number of axes
  - "quality" of representation improves with higher \( k \), perfect representation when \( k \)-number of variables
  - The goal is to simplify, so typically want 2 or 3 axes
  - No eigen vectors and values, direct links to variables is lost (need to use weighted average scores)
NMDS

- **Stress** – measure of the lack of fit between rank order dissimilarities and rank order euclidean distance in ordination space.
- Analogous to cophenetic correlation or other similar measures
- Unbound but expressed as a scaled percentage.
- 0-100%, higher = worse fit
- Generally, <15% is good
- Stress always reduced by adding a dimension, but the goal is to use as few dimensions as possible.

- Various measures of stress.
- Simple correlation between original distance matrix and distance in ordination space.
- Recall that NMDS is non-metric and tried to match rank similarities. Other measures of stress involve residuals from monotonic regressions.
- Note that some functions report this as a percentage and some as a proportion.
- Function `stressplot()` can be used with `nmds` results to visualize.

NMDS

- Species information lost (distance based)
- No % variation accounted for by axes, stress is an analog and stress can be partitioned by axis
- No strict order of importance to axes
- Points and axes can be fully rotated and scaled, all that matters is their relative position in k-dimensional space
- Must specify number of axes ahead of time
  - Stress is reduced as more axes are used. More axes = less information contained on each.
  - 1st dimension of a 2D NMDS is not the same thing as the 1st dimension of a 6D NMDS
  - This is very different from PCoA
- Iterative process, consecutive runs may differ
- Computationally intensive

Example NMDS from the literature. Axes are unlabeled, without units and no % variance accounted for.
NMDS

- Starts with initial configuration (solution), moves points each iteration to reduce stress (seeks to converge on a best solution)
- Because it is iterative, NMDS is
  - Computationally intensive
  - Susceptible to getting stuck in local optima
  - Typically use multiple runs with different starting points

isoMDS Code

- Plain NMDS
  - Code (MASS package):
    ```r
    library(MASS)
    distance <- vegdist(community, method="bray")
    nmds <- isoMDS(distance, k=2)
    ordiplot(nmds)
    ```
  - Options
    - K – number of axes
    - Tol – convergence tolerance
    - Maxit – maximum number of iterations
    - Can specify a starting configuration, otherwise will perform a PCA to obtain a starting position
  - Output
    - Stress at each iteration
    - Final stress
    - Point scores for samples in k dimensions
    - Can be used with missing data
    - Stressplot function does not work

Convergence on a solution

- isoMDS converges on a configuration that minimizes stress, stops when additional iterations do not improve stress

```
> nmds <- isoMDS(distance, k=2, tol=0.001)
  initial  value 14.250589
  iter   5 value 9.263973
  iter  10 value 7.920977
  iter  15 value 7.734398
  iter  15 value 7.726740
  final  value 7.701990
  converged

> nmds <- isoMDS(distance, k=2, tol=0.01)
  initial  value 14.250589
  iter   5 value 8.434460
  iter   5 value 8.385305
  final  value 7.732970
  converged

> nmds <- isoMDS(distance, k=2, tol=0.1)
  initial  value 14.250589
  final  value 9.489541
  converged
```

- Larger tolerance = convergence reached earlier with fewer iterations.
- Tradeoff in computing time.

Stress relations to # iterations and # axes

- Note that isoMDS uses one starting configuration (susceptible to local optima problem)
"% variance" and Species Scores

- No true % variation explained in this analysis.
  - Overall stress is analogous to total percent variation explained
  - Stress for each axis (correlate original distance matrix with pairwise sample similarity on an axis) analogous to variation for an axis

- Species information is lost when forming the distance matrix.
  - Species can be plotted in ordination space by calculating weighted averages.
  - See code and notes for PCoA where we did the same for that analysis.

Multiple Starting Points avoid local optima

- Function `metaMDS` (vegan package) uses multiple starting points:
  - Multiple random starting points (default)
  - Performs other ordination first to get a starting configuration

If you want a good 2D representation, k=2 is better than k=10 even though the stress will be higher.

"% variance" and Species Scores

- No true % variation explained in this analysis.
  - Overall stress is analogous to total percent variation explained
  - Stress for each axis (correlate original distance matrix with pairwise sample similarity on an axis) analogous to variation for an axis

- Species information is lost when forming the distance matrix.
  - Species can be plotted in ordination space by calculating weighted averages.
  - See code and notes for PCoA where we did the same for that analysis.
### NMDS Code
- **metaMDS** (vegan package)
  - uses multiple starting positions and Procrustes to track convergence
- **Code**
  - `metaMDS(community,k=4,distance="bray")`
- **Options**
  - Plot – plot Procrustes errors along the way
  - Trymax – maximum number of starting points
  - `K` – number of axes
  - Distance – distance measure to use on raw data
  - Autotransform – use some automatic transformations if the analysis thinks they are necessary
  - Expand – get species scores as weighted averages
  - Noshare – proportion of samples with shared 0 to initiate a correction.
  - Various other options to center/rotate scores (see metaMDS for details)
- **Output**
  - Stress values
  - Sample scores in ordination space
  - Species weighted average scores

### Principal Components Analysis (PCA)
- **First and most basic eigenvalue based ordination**
- Works with the original dataset, not a distance matrix
- Eigenvalue decomposition of correlation or covariance matrix
  - Relationship among variables and gradients is important
  - Relationship to variables is not lost (no need for wascores)
- Works best with linear relationships among variables, thus, some data are not appropriate (species) and others require transformation

### PCA Functions in R
- **Three functions in R:**
  - `prcomp()` Singular value decomposition of variance/covariance matrix
    - Variance calculated using N-1
  - **princomp()** eigen analysis of correlation or variance/covariance matrix
    - Variance calculated differently (N)
    - No option for scaling data (need to do this before)
    - Option `cor` switches use of correlation or covariance matrix
  - `rda()` vegan package function that will be used for constrained ordinations
    - Similar to `prcomp()`, species and site scores are rescaled according to Legendre, P. and Legendre, L. (1998) Numerical Ecology. 2nd English ed. Elsevier

### Principal Components Analysis (PCA)
Two-dimensional regression
- best fit line through points describes the relationship, explains some proportion of variance.
Principal Components Analysis (PCA)

PCA is similar but in multivariate space. Best fit line in multiple dimensions = component. By definition passes through centroid.

Axis 1
Axis 2

Both vectors pass through the centroid of the data and are rotated to minimize residuals (maximize variance explained).
Relative spatial positioning among points unchanged.
Axis 1 longer = more variance explained.

PCA restrictions
- Relationships among species and gradients are important.
- Thus, variables should be normally distributed and linearly related to gradients.
- Shared zero data is going to be problematic.

Dust-bunny distributions are common in community data.
No transformation is going to fix that---don’t use PCA.

Shared zeroes and Outliers

Shared zero data among species will lead to a false high positive relationship.
A single large outlier can define the linear relationship among two variables.
### Principle Components Analysis (PCA)

- **Code**
  ```r
  prin_comp<-rda(community, scale=TRUE)
  ```
- **Options**
  - Scale = rescale data automatically to have unit variance (~ 1SD per variable)
- **Output**
  - Eigenvalues and eigenvectors
  - Summary of % variance accounted for (“ inertia” = % variance)
    - If scale=TRUE then you have 1 unit of inertia per variable
    - If scale=FALSE then variance contributed is not equal across variables
  - Scaling component
  - Species scores
  - Sample scores

### Assignment

- PCA and NMDS of the same community dataset as we used last week (spaeth.csv)
- Drop rare species (<2 occurrences)
- Log transform
- Use Bray-Curtis index for NMDS
- Compare the two ordination outputs