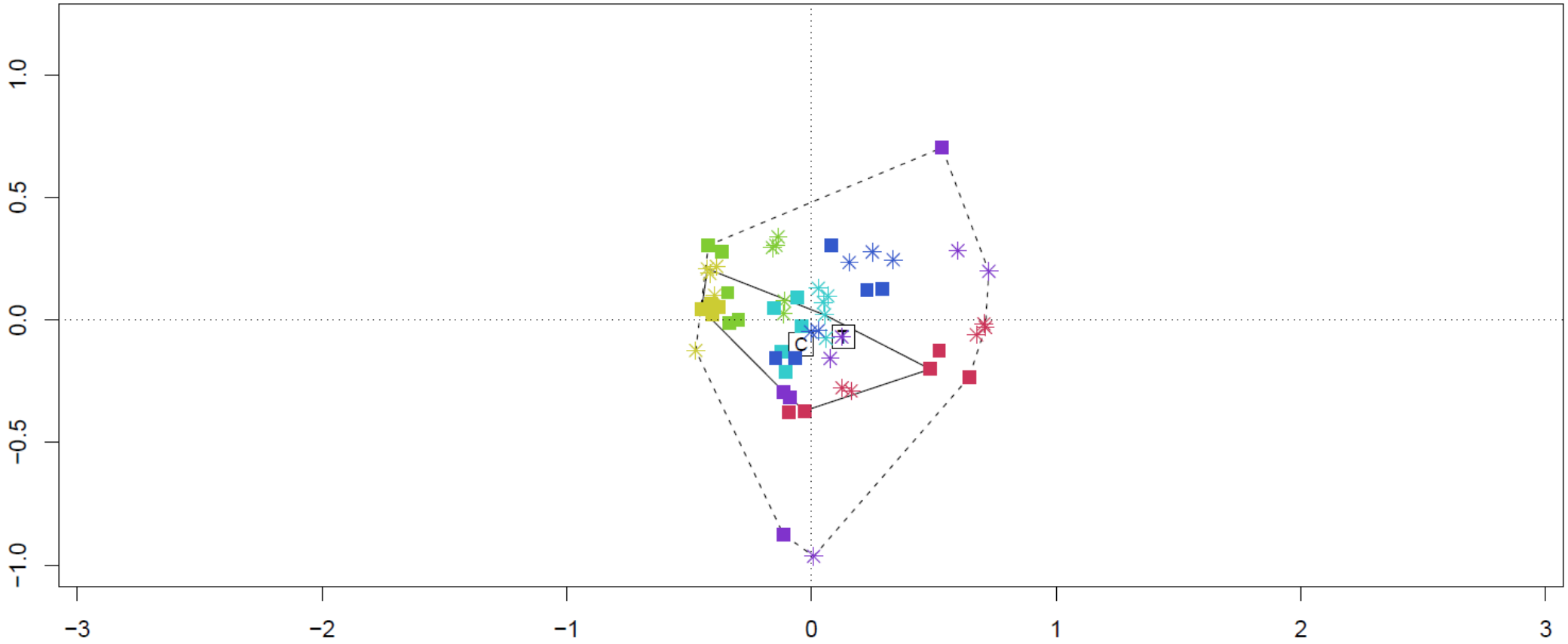
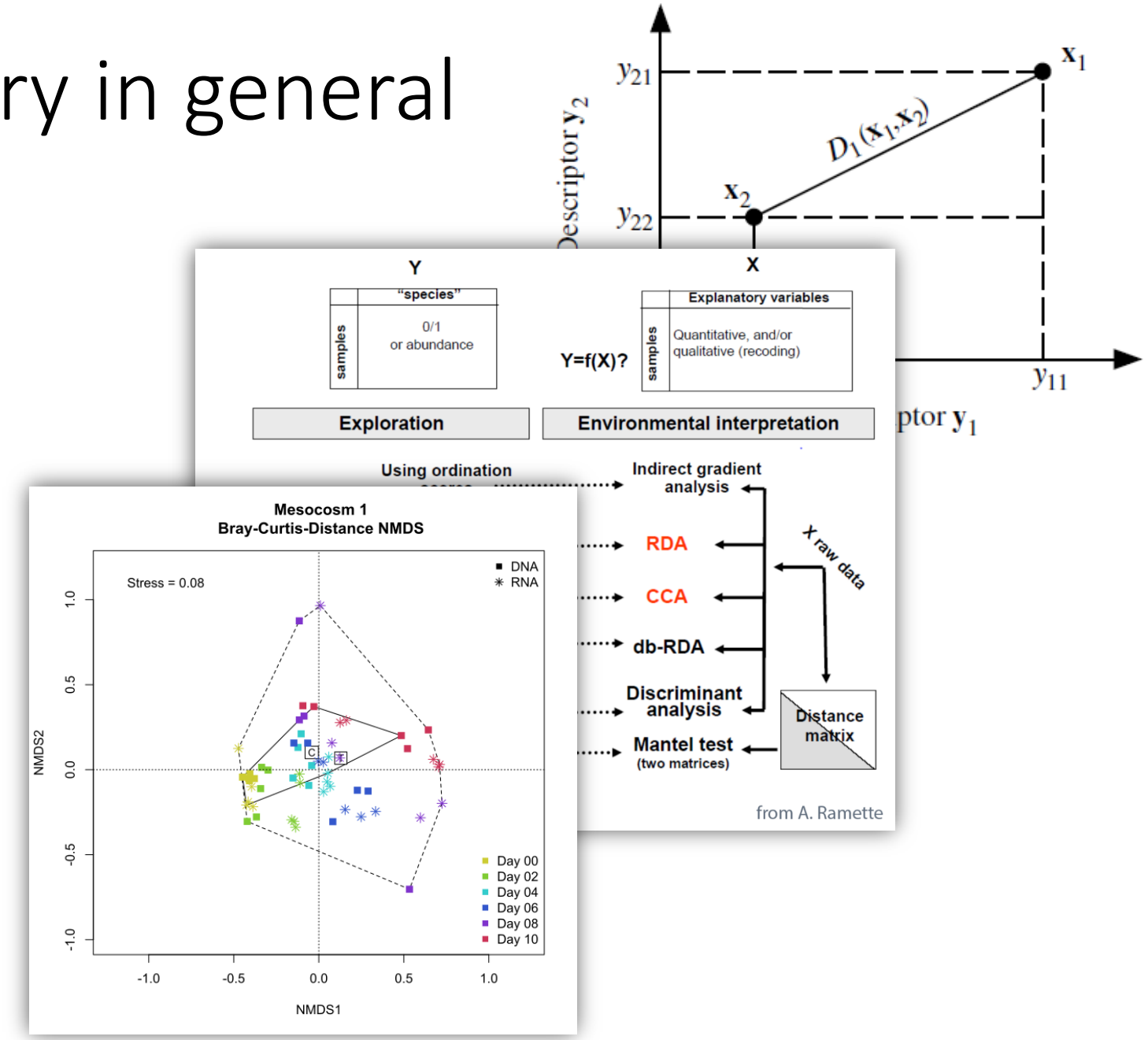
Distance measures



Distance/Simmilariry in general

- Measure of relatedness between samples/sites
- Distance matrices used vor a variety of multivariate ordination techniques (NMDS, PCoA, Cluster-analysis)



The double-zero-problem

Abundance Table

Arctic	Hawaii	Deep Sea	Antarctica	Down town Oldenburg
1	0	0	0	0

Shared absence is no maesure for similar environments

- Distribution Barriers
- Replacement Species

-> in environmental ecology: asymetric measures

symetric distances: euclidean, Mahalanobis, mean character, coefficient of racial likeness

Presence/Absence Measures

- Comparing communities based on the absence/presence of species in a sample
- Species-count is unimportant

	Sample_1540_DNA	Sample_1540_RNA
Bacteria;Actinobacteria	12	8
Bacteria;Bacteroidetes	1415	1326
Bacteria;Chlamydiae	0	0
Bacteria;Cyanobacteria	9	0
Bacteria;Fibrobacteres	0	0
Bacteria;Firmicutes	0	0
Bacteria;Gracilibacteria	0	0
Bacteria;Lentisphaerae	0	0
Bacteria;Marinimicrobia (SAR406 clade)	0	0
Bacteria;Peregrinibacteria	0	0

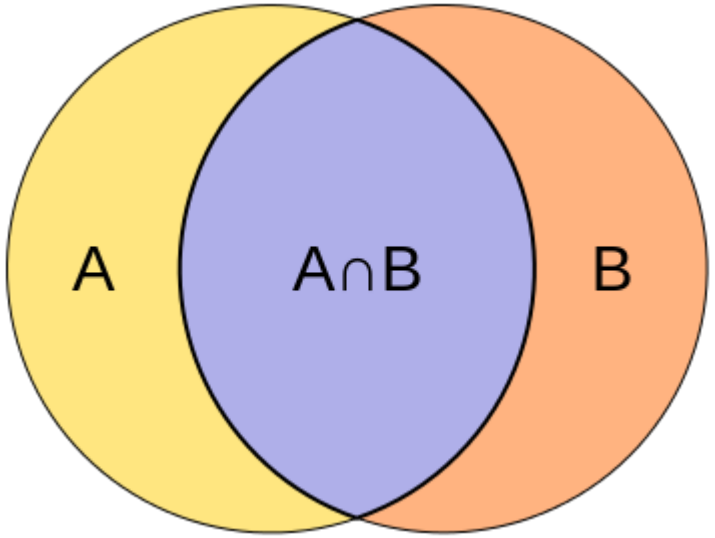
	Sample_1540_DNA	Sample_1540_RNA
Bacteria;Actinobacteria	1	1
Bacteria;Bacteroidetes	1	1
Bacteria;Chlamydiae	0	0
Bacteria;Cyanobacteria	1	0
Bacteria;Fibrobacteres	0	0
Bacteria;Firmicutes	0	0
Bacteria;Gracilibacteria	0	0
Bacteria;Lentisphaerae	0	0
Bacteria;Marinimicrobia (SAR406 clade)	0	0
Bacteria;Peregrinibacteria	0	0

for a comprehensive list of available distance measures: *?vegdist*

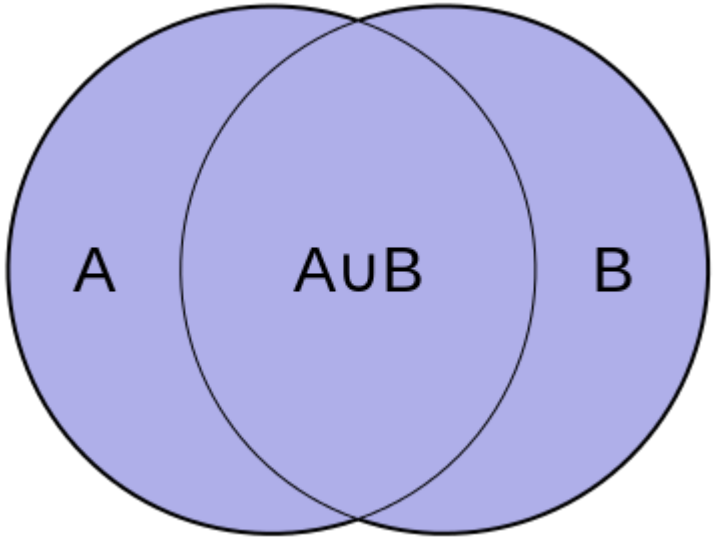
Jaccard-Distance

- Simple coefficient to measure the similarity between two (or more) communities
- Ranges between 0 and 1:
  - 1 = equal samples
  - 0 = no similarities between samples

R-code: `vegan::vegdist(x, method=„jaccard“)`



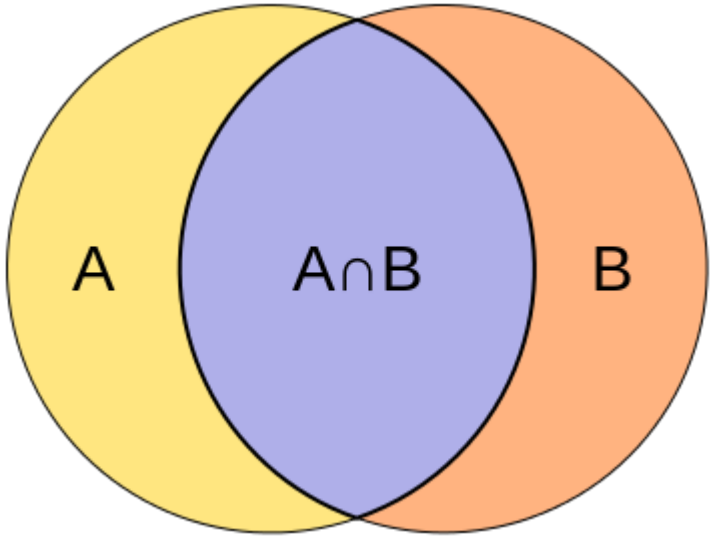
$$1 - J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

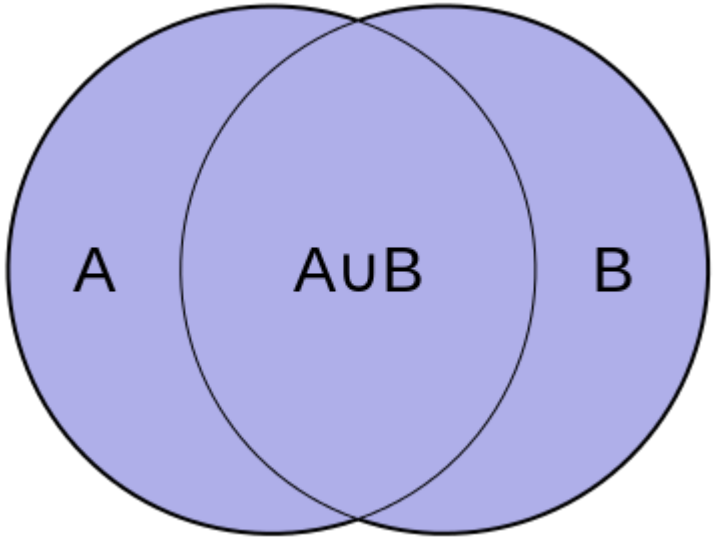


Jaccard-Distance

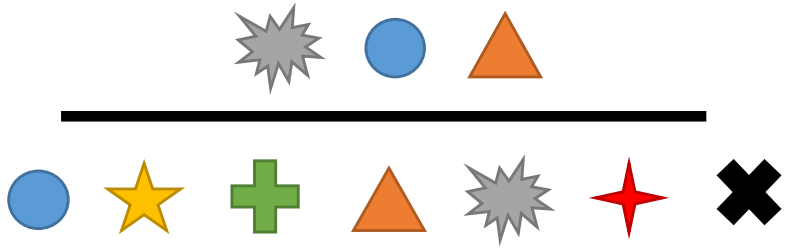
A

B





$$1 - J(A,B) = \frac{|A \cap B|}{|A \cup B|} =$$



Bray-Curtis-(Dis)Simmilarity

- Compares Sites in terms of minnimum abundance of species
- Widely used to compare raw (log transformed) species Data

$$D_{14}(\mathbf{x}_1, \mathbf{x}_2) = \frac{\sum_{j=1}^{p} |y_{1j} - y_{2j}|}{\sum_{j=1}^{p} (y_{1j} + y_{2j})}$$

Quadrats	Species				
	$\mathbf{y}_1$	$\mathbf{y}_2$	$\mathbf{y}_3$	$\mathbf{y}_4$	$\mathbf{y}_5$
$\mathbf{x}_1$	2	5	2	5	3
$\mathbf{x}_2$	3	5	2	4	3
$\mathbf{x}_3$	9	1	1	1	1

$$D_{14}(\mathbf{x}_1, \mathbf{x}_2) = \frac{1+0+0+1+0}{17+17} = 0.059$$

$$D_{14}(\mathbf{x}_1, \mathbf{x}_3) = \frac{7+4+1+4+2}{17+13} = 0.600$$

$$D_{14}(\mathbf{x}_2, \mathbf{x}_3) = \frac{6+4+1+3+2}{17+13} = 0.533$$

If the sum of species greatly varies among sites, the Bray-Curtis index may become negative!
-> sqrt-transformation

```
R-code: vegan::vegdist(x, method=„bray“)
```

Sörensen, Sorensen, Sørensen or Dice index

Simply put: **Sörensen index = Bray-Crurtis** distance on **Presence/Absence** data

Difference to the Jaccard-Index:

- more robust against outliers
- more sensitive in heterogeneous Datasets

R-code: `ecodist::distance(x, method=„sorensen“)`

Euclidean Distance

- metric distance between two points (or vectors) in geometrical „euclidean space“
- symmetrical distance
- Range: zero to positive inf.

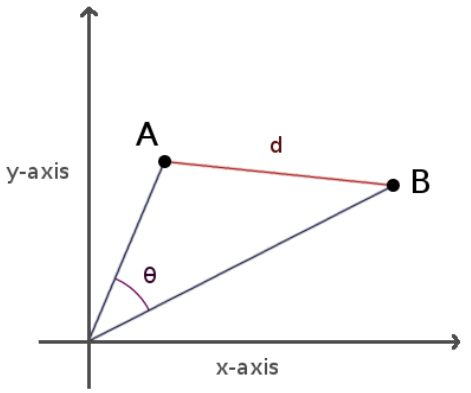
$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2+(q_2-p_2)^2+\cdots+(q_n-p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}.$$

R-code: `stats::dist(x, method=„euclidean“)`

- Hellinger transformation for ecological data (used in PCA, PCoA, RDA, CCA) $\sqrt{\frac{species\ count\ x}{site's\ total\ abundance}}$

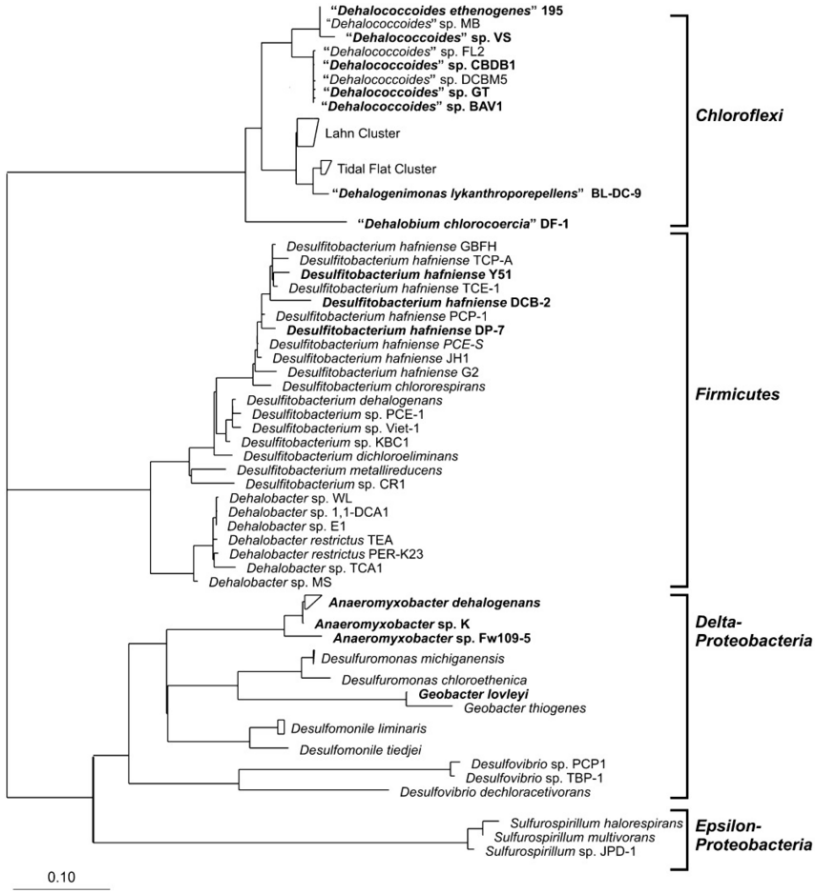
R-code: `vegan::decostand(x,“hel“), stats::dist(x.hel)`



- Chord transformation: euclidean distance normalized to 1

R-code: `vegan::decostand(x,“nor“); stats::dist(x.nor)`

UniFrac (Lozupone & Knight 2005)



- Phylogenetic distance measure (especially for sequence based community assessments)
- requires a rooted tree auf the OTU-table

R-code:

```
system("muscle3.8.31_i86win32.exe -in OTU_table.fasta -out OTU_table_red.afa")

system("muscle3.8.31_i86win32.exe -maketree -in OTU_table_red.afa -out
OTU_table_red.phy -cluster neighborjoining")
```

(http://www.drive5.com/muscle/downloads.htm)

Available unifrac-measures: weighted, unweighted, variance adj. weighted, alpha = 0, alpha = .5

R-package: GUnifrac

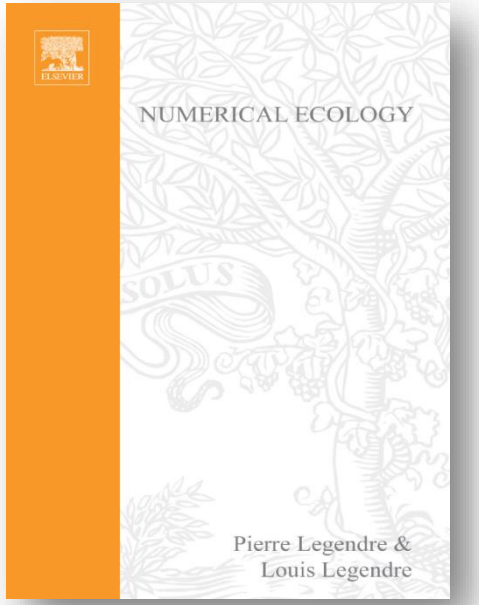
PERMAVOVA (Permutational Multivariate Analysis of Variance Using Distance Matrices)

- **R-code:** `vegan::adonis(distance.table ~ Env1 + Env2, method = „bray“)`
- Permutational MANOVA explains variance withing the dataset using environmental parameters

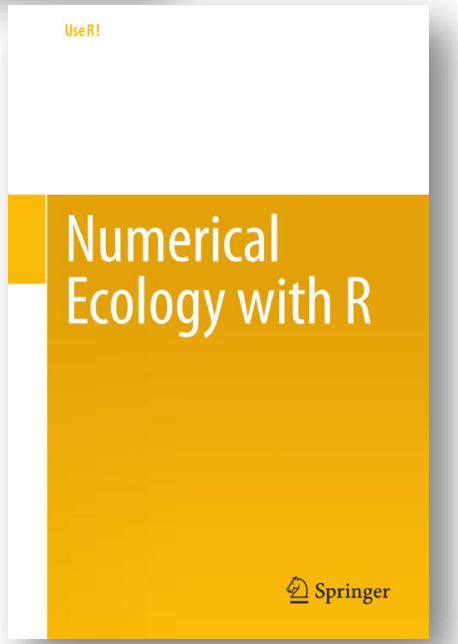
```
> adonis(m1log ~ Meta1$Type + Meta1$Day + Meta1$Nuc, permutations = 1000)
> Call: adonis(formula = m1log ~ Meta1$Type + Meta1$Day + Meta1$Nuc, permutations = 1000)
> Permutation: free
> Number of permutations: 1000
> Terms added sequentially (first to last)
>            Df  SumsOfSqs  MeanSqs   F.Model   R2       Pr(>F)
> Meta1$Type 1  0.2214     0.22137   7.0298    0.06499 0.000999 ***
> Meta1$Day  5  1.1336     0.22672   7.1998    0.33282 0.000999 ***
> Meta1$Nuc  1  0.4451     0.44512   14.1351   0.13068 0.000999 ***
> Residuals 51  1.6060     0.03149             0.47151
> Total     58  3.4061                         1.00000

>  --- Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Further Information

- Legendre & Legendre 1998 - Numerical ecology
  - Mathematical background



- Borcard, Gillet & Legendre 2011 - Numerical Ecology with R
  - Application in R



- GUSTA ME: https://sites.google.com/site/mb3gustame/reference/dissimilarity

Idea… Little Helper Scripts!

```r
########################################## Anderson-Darling Normality Test #################
library(nortest)

cat("Data\tp-value\n")
for(i in 1:12)
{
  x = species[,i]
  j = ad.test(x)
  cat(Species_names[i], "\t", j$p.value, "\n")

}

# Testing for normal distribution of the Data; p>0.05 = normal distributed Data. If data are
# not normally distributed, Pearsons correlation is not an option

########################################## Correlation Test ################################
setwd("C:/Users/icbmadmin/Desktop/R/Transects HE425/Correlations/")

##### Full Cor. Table
sink("Correlation_table.txt") # File in which all the console output will be saved
cat("Group\tParameter\tPearson Correlation\tp-value\n") # Columnames for the file "\t" = tabstop
for(i in 1:12) # Species/whatever from a to b (in species tabele) to be tested against ...
{
  x = Species[,i]

  for(j in 3:9) # ...parameters in columns from x to y (in env-data-table)
    {
    y = ENV[,j]

    cor = cor.test(x = x, y = y, method = "pearson")

    cat(Species_names[i], "\t", ENV_names[j], "\t", cor$estimate,"\t", cor$p.value, "\n")

    }
 }
sink()
```