nMDS

Pablo E. Gutiérrez-Fonseca

9/25/2021

```r
libraries <- c("vegan", "ggplot2", "dplyr")
lapply(libraries, require, character.only = TRUE)
```

```
## [[1]]
## [1] TRUE
##
## [[2]]
## [1] TRUE
##
## [[3]]
## [1] TRUE
```

Paso 1. LLamar a la tabla (como .csv)

```r
Moth_full  <- read.csv("data/fullmatrix.csv")
head(Moth_full)
```

Paso 2. Seleccionar las especies.

```r
moth_sp <- select(Moth_full, M1:A248)
str(moth_sp)
```

```
## 'data.frame':    66 obs. of  233 variables:
##  $ M1  : int  1 0 6 3 2 0 0 0 0 0 ...
##  $ M2  : int  0 0 1 0 0 0 0 0 0 0 ...
##  $ M3  : int  0 0 2 0 2 0 0 0 0 0 ...
##  $ M5  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ M7  : int  2 4 21 5 6 78 11 10 13 8 ...
##  $ M8  : int  0 0 1 0 0 0 0 0 0 0 ...
##  $ M10 : int  0 0 1 0 0 0 0 0 0 1 ...
##  $ M13 : int  1 1 0 1 0 0 0 0 0 0 ...
##  $ M15 : int  0 2 0 0 0 0 0 0 0 0 ...
##  $ M17 : int  1 0 0 1 0 0 0 0 0 0 ...
##  $ M19 : int  0 0 2 0 1 0 0 0 0 0 ...
##  $ M25 : int  0 0 1 0 0 0 0 1 0 1 ...
##  $ M26 : int  1 1 2 9 3 14 5 2 5 2 ...
##  $ M29 : int  0 1 0 0 0 0 0 0 0 0 ...
##  $ M45 : int  0 0 1 0 0 0 0 0 0 3 ...
##  $ M48 : int  1 1 2 0 0 0 1 0 0 0 ...
##  $ M50 : int  0 0 1 0 1 0 0 0 0 1 ...
```

```
## $ M52 : int  0 2 1 1 0 1 0 0 0 0 ...
## $ M53 : int  1 0 3 0 1 0 0 0 0 0 ...
## $ M54 : int  0 0 1 0 0 0 0 0 0 0 ...
## $ M55 : int  7 1 2 0 0 0 0 0 1 2 ...
## $ M66 : int  0 0 0 1 0 0 0 0 0 0 ...
## $ M70 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ M74 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ M77 : int  0 0 0 2 0 0 0 0 0 0 ...
## $ M79 : int  0 0 1 0 0 4 1 0 0 1 ...
## $ M80 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ M81 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ M85 : int  0 1 0 0 0 0 0 0 1 0 ...
## $ M86 : int  0 1 1 1 0 0 1 0 0 0 ...
## $ M87 : int  2 3 2 0 0 4 1 0 1 2 ...
## $ M88 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ M89 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ M94 : int  0 1 3 0 1 0 4 2 10 1 ...
## $ M95 : int  0 0 1 0 0 0 0 0 0 1 ...
## $ M96 : int  1 0 1 0 0 0 0 1 1 0 ...
## $ M100: int  0 0 1 0 0 0 0 0 0 0 ...
## $ M104: int  0 0 0 0 0 0 0 0 0 0 ...
## $ M105: int  0 0 0 0 0 0 0 0 0 0 ...
## $ M106: int  0 0 1 0 0 0 0 0 0 0 ...
## $ M109: int  0 0 0 0 0 0 0 0 0 0 ...
## $ M110: int  0 0 0 0 0 0 0 0 0 0 ...
## $ A1  : int  1 0 0 0 0 0 0 0 0 0 ...
## $ A3  : int  4 0 0 0 0 0 0 0 1 2 ...
## $ A4  : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A5  : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A6  : int  1 0 0 0 0 0 0 0 0 0 ...
## $ A7  : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A8  : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A9  : int  0 0 0 0 0 0 0 0 1 0 ...
## $ A10 : int  1 0 0 0 0 0 0 0 0 0 ...
## $ A11 : int  1 1 0 1 1 0 0 0 1 0 ...
## $ A12 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A14 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A15 : int  1 1 0 0 0 0 1 0 1 1 ...
## $ A16 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A17 : int  0 0 0 0 1 1 1 2 0 0 ...
## $ A19 : int  0 2 0 1 1 2 1 1 0 0 ...
## $ A21 : int  0 1 0 1 0 5 0 0 0 1 ...
## $ A22 : int  0 0 0 0 0 3 0 2 1 0 ...
## $ A24 : int  1 0 0 0 0 0 0 0 0 0 ...
## $ A25 : int  1 0 1 4 4 0 0 0 0 0 ...
## $ A27 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A28 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A29 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A30 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A31 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A32 : int  1 0 0 0 0 0 0 2 0 0 ...
## $ A33 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A34 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A36 : int  0 1 1 0 0 0 0 0 0 0 ...
```

```
## $ A37 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A38 : int  1 0 0 0 0 0 0 0 0 1 ...
## $ A39 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A40 : int  0 0 0 0 0 2 0 2 0 ...
## $ A41 : int  0 0 1 0 1 0 0 0 2 0 ...
## $ A42 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A43 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A44 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A45 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A46 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A47 : int  0 0 0 0 0 0 1 0 0 0 ...
## $ A48 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A49 : int  0 0 0 0 0 0 0 0 0 1 ...
## $ A50 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A51 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A52 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A53 : int  0 0 1 0 0 0 0 0 0 0 ...
## $ A54 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A55 : int  0 0 0 0 0 0 0 0 0 1 ...
## $ A56 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A58 : int  0 0 0 0 0 0 0 0 1 0 ...
## $ A59 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A60 : int  0 0 1 0 0 1 0 1 0 0 ...
## $ A61 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A65 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A66 : int  0 1 0 0 0 0 0 0 0 0 ...
## $ A67 : int  0 0 0 0 0 0 0 0 0 0 ...
## $ A68 : int  0 0 0 0 0 0 0 0 0 0 ...
##  [list output truncated]
```

```
ncol(moth_sp)
```

```
## [1] 233
```

```
nrow(moth_sp)
```

```
## [1] 66
```

Paso 3. Vamos hacer el nMDS

```
set.seed(1)
```

```
moth.mds <- metaMDS(moth_sp, distance = "bray", k = 2, trymax=100)  #using all the defaults
```

```
## Square root transformation
## Wisconsin double standardization
## Run 0 stress 0.2509353
## Run 1 stress 0.2595334
## Run 2 stress 0.2523396
## Run 3 stress 0.282036
## Run 4 stress 0.2509339
## ... New best solution
```

```
## ... Procrustes: rmse 0.0009701113   max resid 0.006667077
## ... Similar to previous best
## Run 5 stress 0.250961
## ... Procrustes: rmse 0.003294134   max resid 0.01957031
## Run 6 stress 0.2566342
## Run 7 stress 0.2510057
## ... Procrustes: rmse 0.007469064   max resid 0.04719079
## Run 8 stress 0.2530351
## Run 9 stress 0.251111
## ... Procrustes: rmse 0.01288445   max resid 0.05411702
## Run 10 stress 0.2563251
## Run 11 stress 0.2509585
## ... Procrustes: rmse 0.006726555   max resid 0.04626972
## Run 12 stress 0.2522903
## Run 13 stress 0.2509695
## ... Procrustes: rmse 0.003379292   max resid 0.02477938
## Run 14 stress 0.2523395
## Run 15 stress 0.286507
## Run 16 stress 0.2572426
## Run 17 stress 0.2566301
## Run 18 stress 0.2563038
## Run 19 stress 0.2579864
## Run 20 stress 0.2810398
## *** Solution reached
```

```
moth.mds
```

```
##
## Call:
## metaMDS(comm = moth_sp, distance = "bray", k = 2, trymax = 100)
##
## global Multidimensional Scaling using monoMDS
##
## Data:      wisconsin(sqrt(moth_sp))
## Distance: bray
##
## Dimensions: 2
## Stress:      0.2509339
## Stress type 1, weak ties
## Two convergent solutions found after 20 tries
## Scaling: centring, PC rotation, halfchange scaling
## Species: expanded scores based on 'wisconsin(sqrt(moth_sp))'
```

Paso 4. Vamos hacer el grafico.

```
plot(moth.mds, type="t")
```
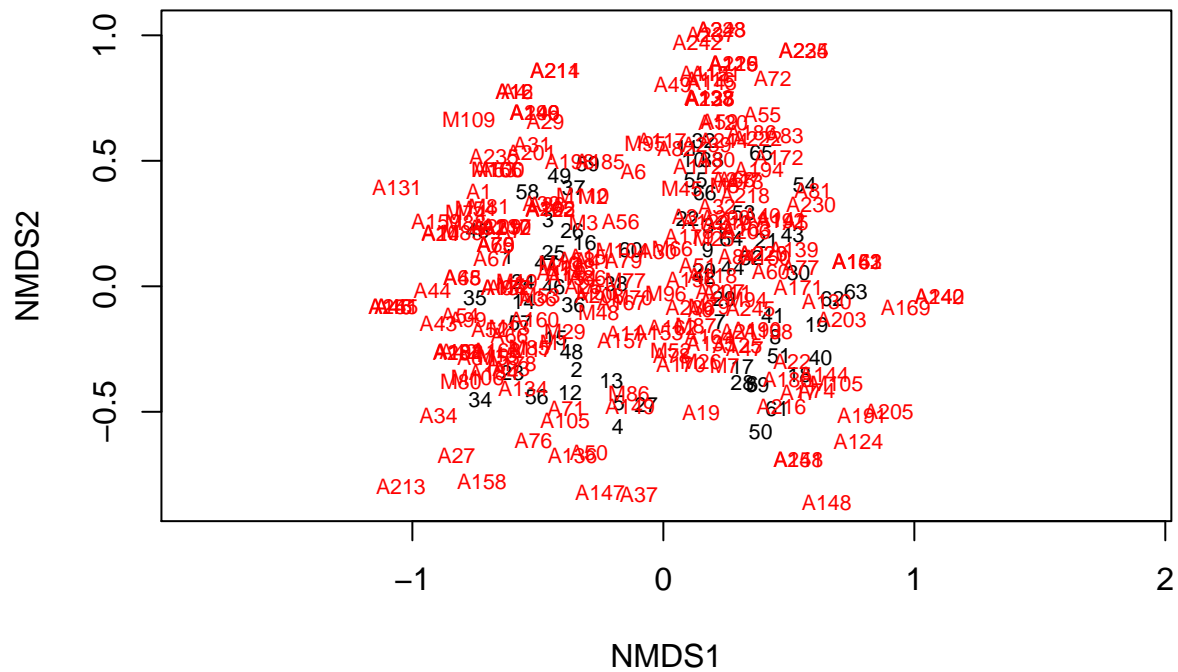
Paso 4. Vamos a poner separarlo por Habitat, Site y Periodo

```
Habitat <- select(Moth_full, Habitat)
Site <- select(Moth_full, Site)
Period <- select(Moth_full, Period)
```