Table of Contents

Ordination analysis 1
PCoA & NMDS (distance-based unconstrained ordination) 1
Example 1 - PCoA on the matrix of distances between European cities 1
Example 2: NMDS on the Morse code confusion matrix 4
Example 3: NMDS of river valley dataset 5

Ordination analysis

PCoA & NMDS (distance-based unconstrained ordination)

Theory R functions **Examples** Exercise 🔒

Example 1 - PCoA on the matrix of distances between European cities

We use data from the variable `eurodist`, which is available in R (you don't need to install any library, just type `euro disc`). This distance matrix contains real geographical distances among big European cities (driving distance, in km). We will use this matrix to calculate PCoA and draw the PCoA ordination diagram, and also a screeplot of eigenvalues for individual PCoA axes.

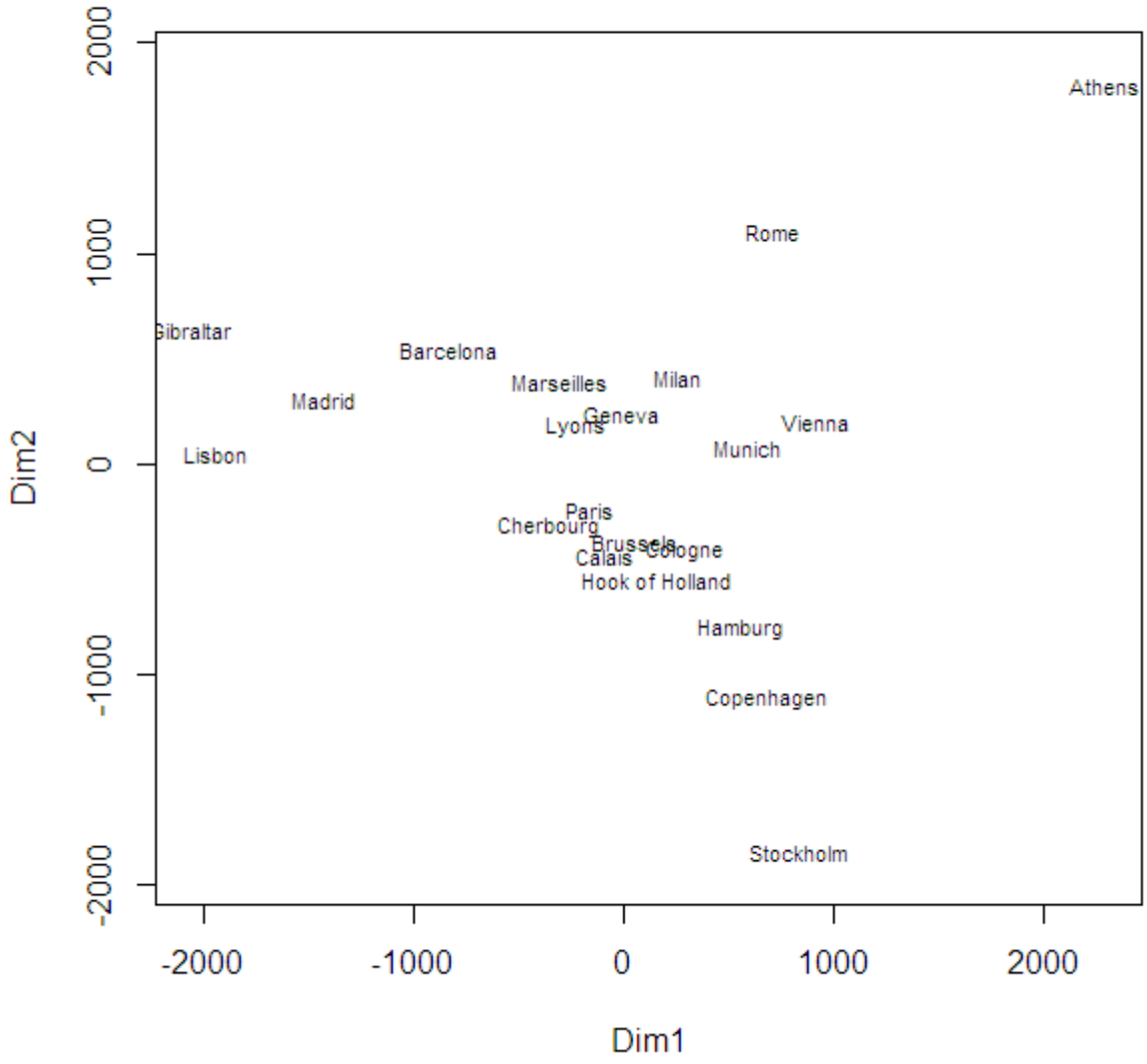
To calculate PCoA, use the base R function `cmdscale` (note that `vegan` contains the function `wcmdscale`, which in default setting is doing the same):

```
pcoa <- cmdscale (eurodist, eig = TRUE)
```

Note that I set up the argument `eig = TRUE` - in this way, the `cmdscale` function returns also the eigenvalues for individual axes (in the default setting this argument is set to FALSE and the function returns only the data frame of sample scores on individual PCoA axes).

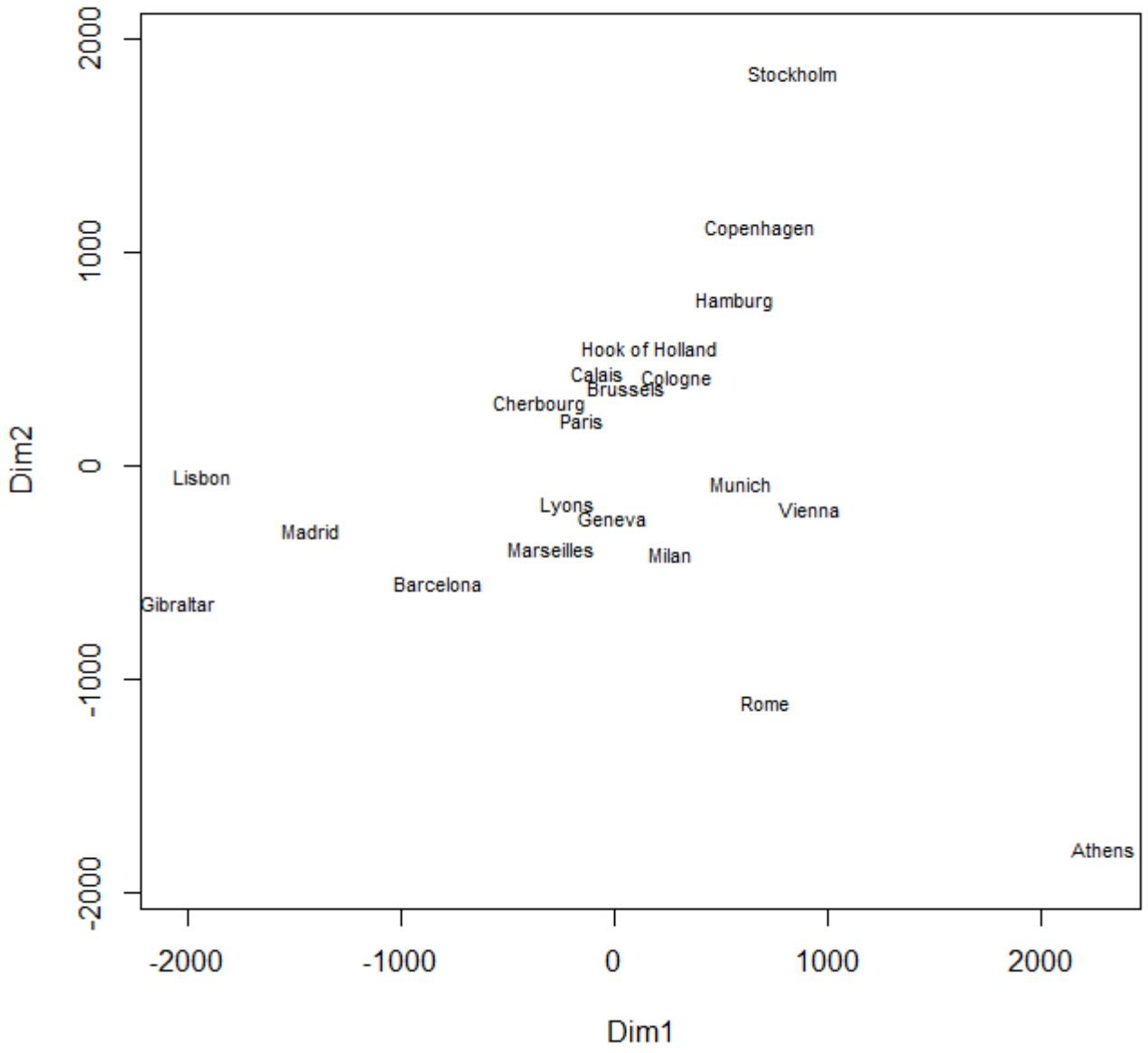
Now we can draw the ordination diagram of the cities:

```
library (vegan)
ordiplot (pcoa, display = 'sites', type = 'text')
```



You can see that the distances between cities make intuitive sense (Athens are far from Stockholm, for example), and it almost looks like the map of Europe, except that Athens are at north and Stockholm at south. Let's flip the y-axis (the second axis of PCoA) and draw the ordination diagram again. The sample scores in PCoA ordination are in the object `pcoa`, in the component `points`[1] (use `str (pcoa)` if you wish to see the structure of `pcoa` object).

```
pcoa[,2] <- -pcoa[,2]
ordiplot (pcoa, display = 'sites', type = 't')
```

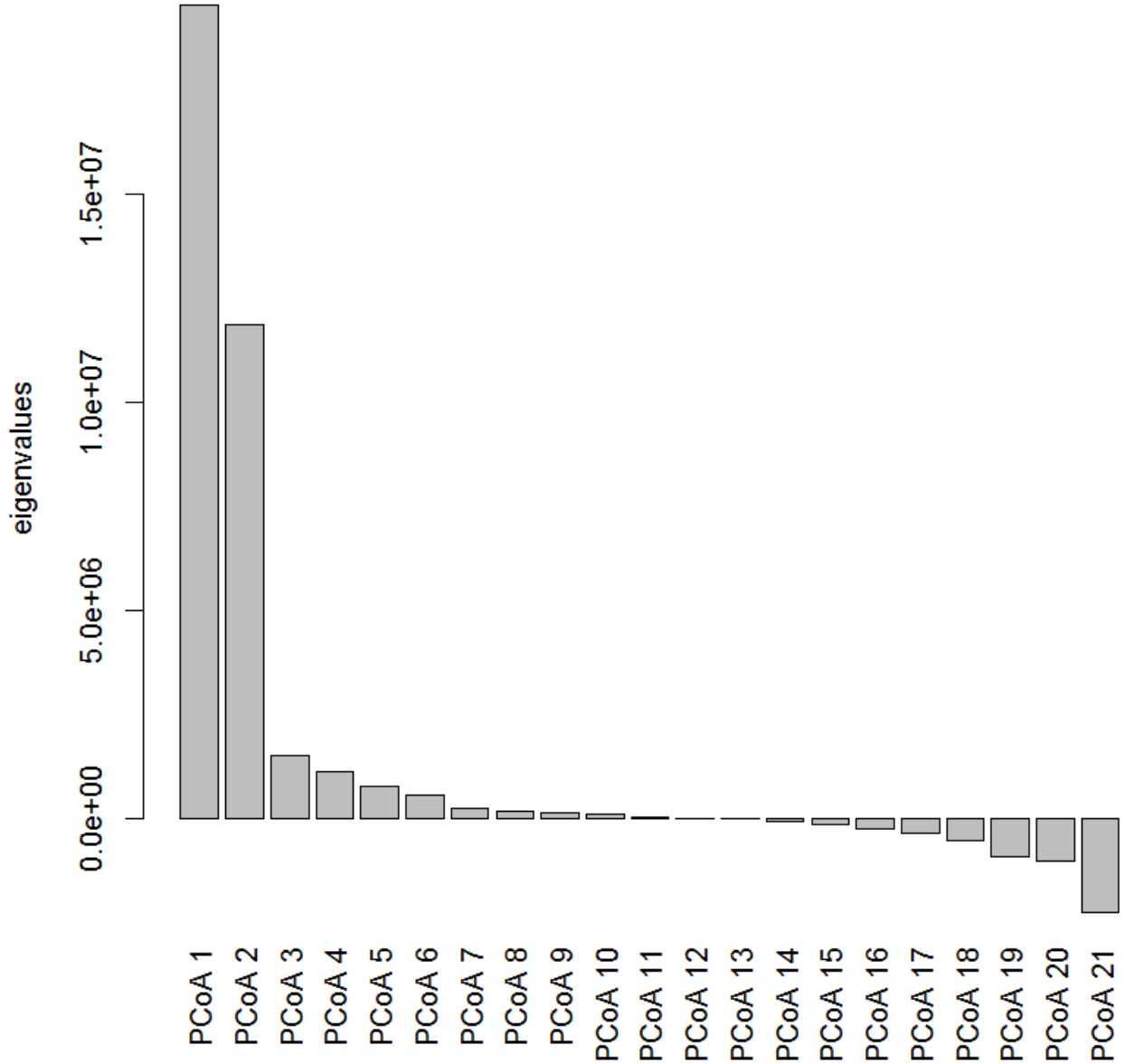


Now the distribution of cities make better geographical sense!

Finally, let's draw the screeplot with eigenvalues for individual axes; these eigenvalues are stored in the component `$eig` of the object `pcoa`:

```
barplot (pcoa$eig, names = paste ('PCoA', 1:21), las = 3, ylab =
'eigenvalues')
```

Note that `names` argument adds the names to tickmarks on horizontal axis, `las` argument influences rotation of labels on both x and y axis (see `?par` for explanation) and `ylab` adds the label to the



Example 2: NMDS on the Morse code confusion matrix

Use data about confusion of different Morse codes, originating from Rothkopf's experiment with Morse codes. This is a classical data set, used by Shepard (1962)[2] to demonstrate the use of NMDS analysis.

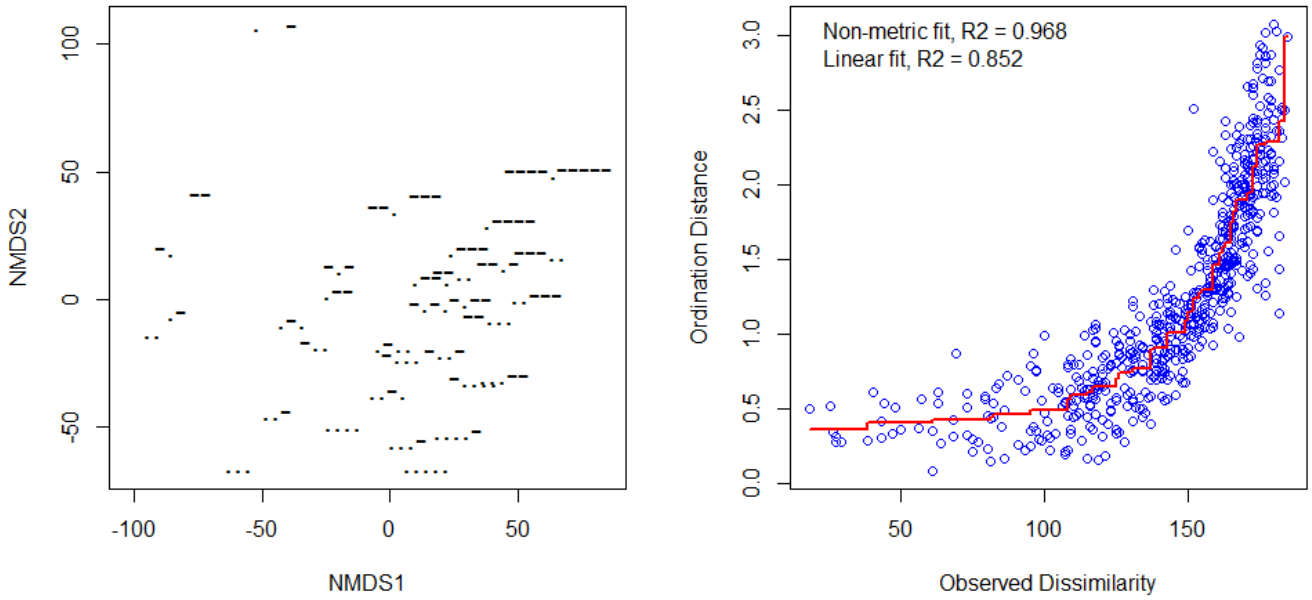
```
library (vegan)
morse.dist <- read.delim
('https://raw.githubusercontent.com/zdealveindy/anadat-r/master/data/morsecodes-dist.txt', row.names = 1, head = T)
names (morse.dist) <- rownames (morse.dist)
NMDS <- metaMDS (morse.dist)
NMDS
```

The stress values are in the following output:

```
Call:
metaMDS(comm = morse.dist)

global Multidimensional Scaling using monoMDS

Data:     morse.dist
Distance: user supplied

Dimensions: 2
Stress:     0.1800255
Stress type 1, weak ties
Two convergent solutions found after 1 tries
Scaling: centring, PC rotation
Species: scores missing
```

The ordination diagram and Shepard diagram could be drawn in the following way:

```
par (mfrow = c(1,2))
ordiplot (NMDS, cex = 1.5, type = 't')
stressplot (NMDS)
```



From left to right there is a gradient of increasing code length, from bottom up increases the proportion of long beeps within the code.

Example 3: NMDS of river valley dataset

```
vltava.spe <- read.delim
('https://raw.githubusercontent.com/zdealveindy/anadat-r/master/data/vltava-spe.txt', row.names = 1)
```

```
NMDS <- metaMDS (vltava.spe)
```

```
Square root transformation
Wisconsin double standardization
Run 0 stress 0.2022791
Run 1 stress 0.2193042
Run 2 stress 0.2130607
Run 3 stress 0.208742
Run 4 stress 0.2022791
... procrustes: rmse 9.278716e-06  max resid 3.31574e-05
*** Solution reached
```

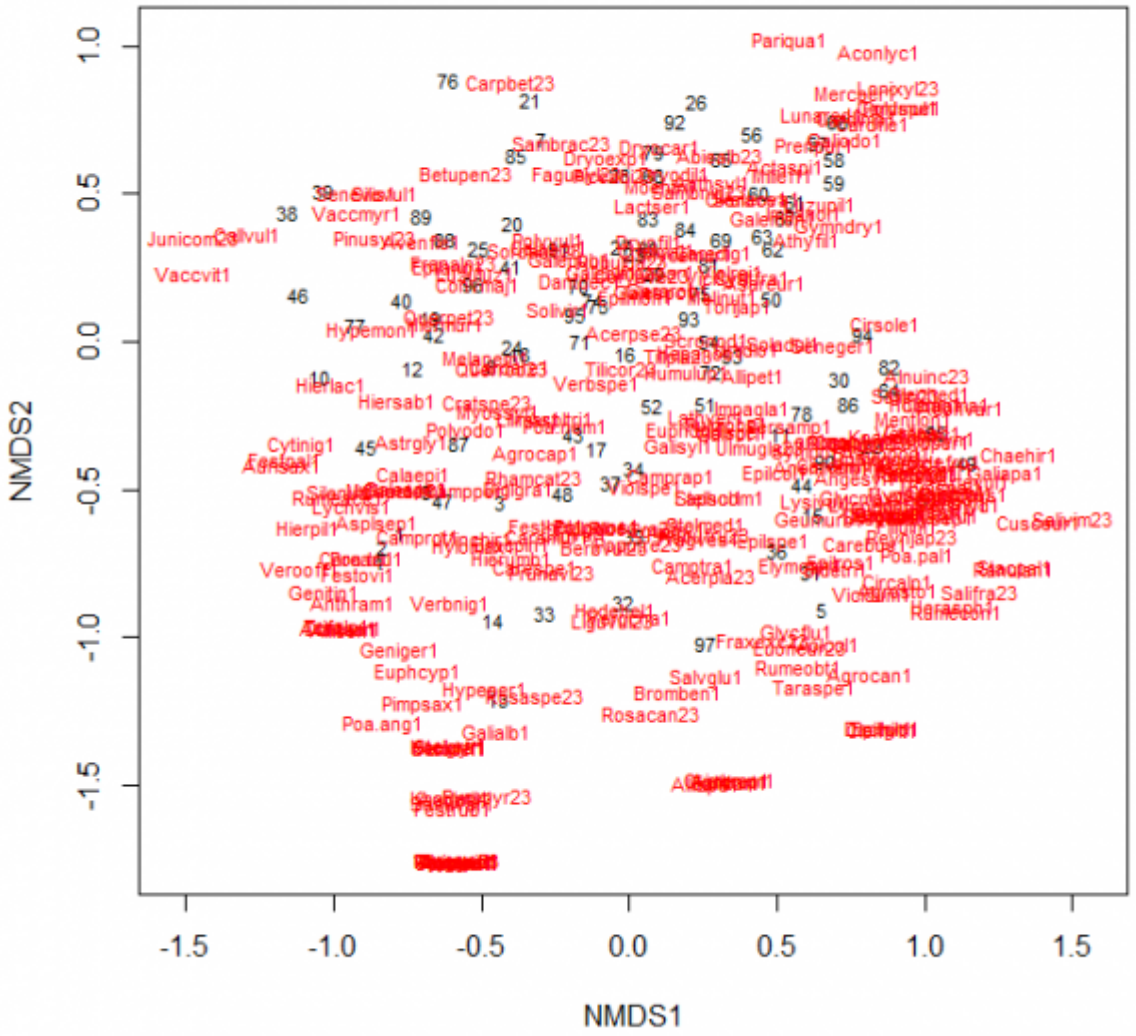
```
NMDS
```

```
Call:
metaMDS(comm = vltava.spe)

global Multidimensional Scaling using monoMDS

Data:     wisconsin(sqrt(vltava.spe))
Distance: bray

Dimensions: 2
Stress:     0.2022791
Stress type 1, weak ties
Two convergent solutions found after 4 tries
Scaling: centring, PC rotation, halfchange scaling
Species: expanded scores based on ‘wisconsin(sqrt(vltava.spe))’
```

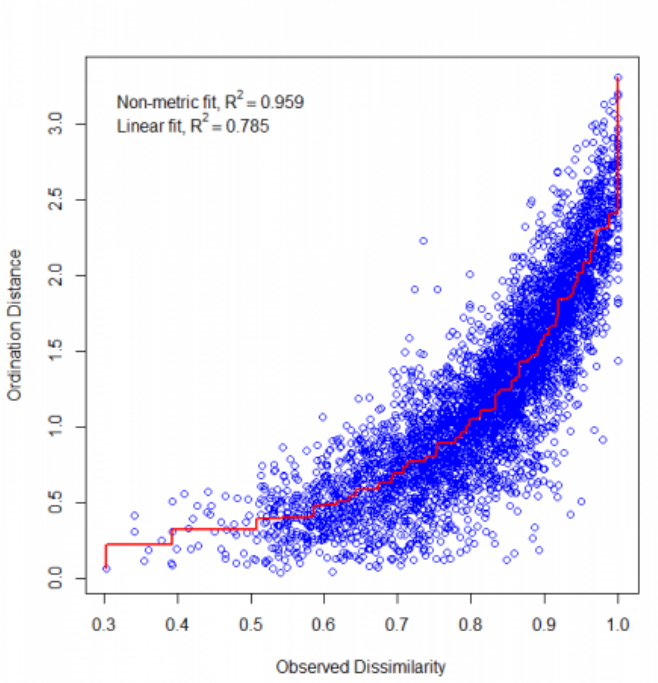
If the default setting of metaMDS function is used, the data are automatically (if necessary) transformed (in this case, combination of wisconsin and sqrt transformation was used). In this case, stress value is 20.2.

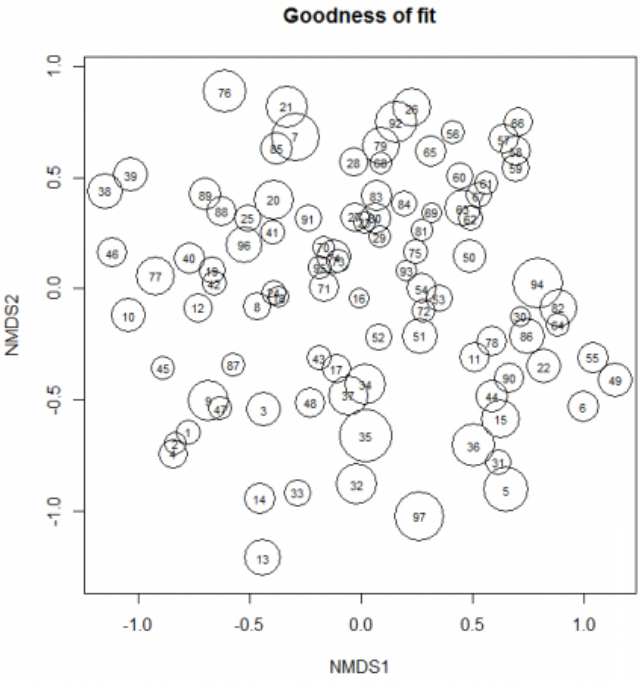
To draw the result, use the function `ordiplot`. In this case, using `type = 't'` will add text labels (default setting adds only points):

```
ordiplot (NMDS, type = 't')
```




```
par (mfrow = c(1,2)) # this function divides plotting window into two columns
stressplot (NMDS)
plot (NMDS, display = 'sites', type = 't', main = 'Goodness of fit') # this function draws NMDS ordination diagram with sites
points (NMDS, display = 'sites', cex = goodness (NMDS)*200) # and this adds the points with size reflecting goodness of fit (bigger = worse fit)
```





1)

If we have calculated `cmdscale` with `eig = FALSE`, the structure of `pcoa` object would be simpler, it would be just a data frame with sample scores; with `eig = TRUE` the object became a list with the score data frame nested inside within the component `$points`.

2)

Shepard, R. N. (1962): The Analysis of Proximities: Multidimensional Scaling with an Unknown Distance Function, I and II. *Psychometrika*, 27: 125-139 and 219-246.

From:
https://www.davidzeleny.net/anadat-r/ - **Analysis of community ecology data in R**

Permanent link:
https://www.davidzeleny.net/anadat-r/doku.php/en:pcoa_nmds_examples?rev=1548499229

Last update: **2019/01/26 18:40**