Task 5

Task 5

Introduction

In a study on Morse code signals Rothkopf (1957) asked 598 subjects to indicate for each pair of signals (acoustically presented after each other) whether the two signals were equal. Note that each signal consists of a sequence of short beeps (0.05 sec) and long beeps (0.15 sec) separated by a silence (0.05 sec). The data set confusion.Rdata contains a 36 x 36 matrix of confusion rates for all pairs of signals included in the study. The signals in the subsequent rows of the confusing matrix are the letters of the alphabet A, B, C… in alphabetical order and the numbers 1,2,3,4,5,6,7,8,9,0. The value in cell (i,j) of the matrix represents the percentage of subjects who indicated that signal i was equal to signal j, when signal j was presented after signal i.

Methodology

First of all and in order to observe dissimilarities, smacofSym() will be used to conduct a MDS with 2 dimensions.

After

We compute a symmetric matrix of similarities by taking the average of corresponding elements in the confusion matrix above and below the diagonal, next we convert the matrix of similarities into a matrix of dissimilarities, and finally we put all diagonal elements of the dissimilarity matrix equal to 0: …… ## Results ……

Question 1

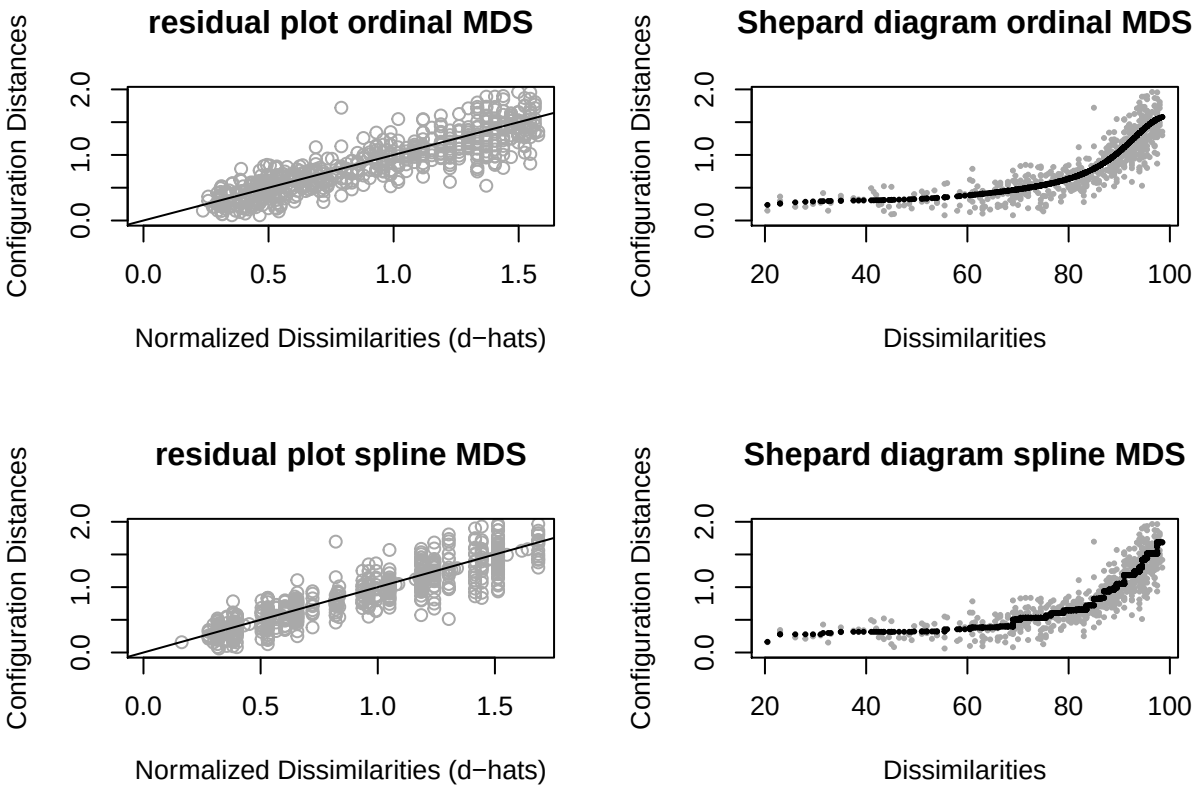
1. Use *smacofSym()* to conduct MDS with 2 dimensions and assuming different measurement levels (i.e., *ratio, interval, mspline, ordinal*) for the observed dissimilarities.

1.1 Getting information from dissimilarity matrix. Here ratio, interval, mspline and ordinal values are obtained.

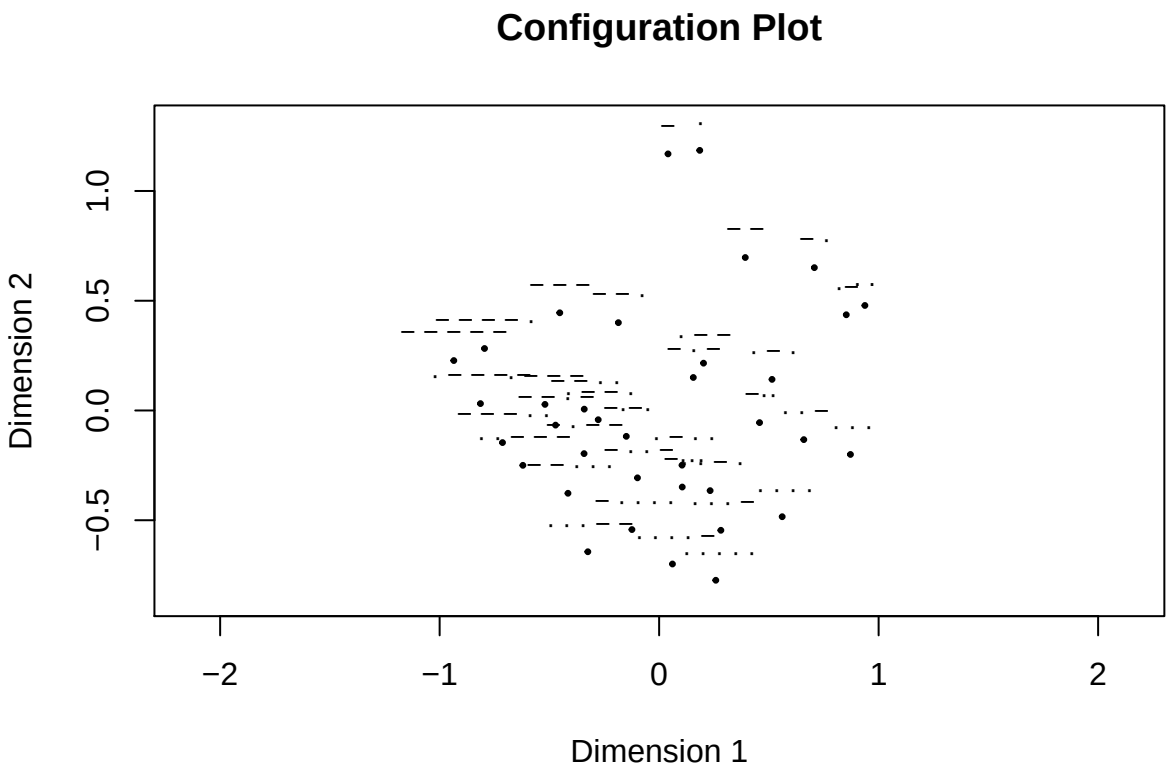
```
#ratio
m1 <- smacofSym(delta=dissim, ndim=2, type="ratio", init="torgerson")
#interval
m2 <- smacofSym(delta=dissim, ndim=2, type="interval", init="torgerson")
#mspline
m3 <- smacofSym(delta=dissim, ndim=2, type="mspline", spline.degree =4,
                spline.intKnots = 4, init="torgerson")
#ordinal
m4 <- smacofSym(delta=dissim, ndim=2, type="ordinal", init="torgerson")
```

Table 1: Stress-1 table values

m1	m2	m3	m4
0.3	0.262	0.205	0.191




```
plot(m3,plot.type="conf")
```



Question 2

2. Evaluate the goodness of fit of solutions with different measurement levels using *stress-1*, and by computing stress norms with the functions *randomstress()* and *permutation()*. Discuss which solution you would select. Investigate the stability of the selected solution using the *Jackknife*.

Question 3

3. Construct a data set of external variables that describe the signals (e.g. length of the signal, proportion of short beeps in the signal, etc.). Use an *MDS biplot* to project the external variables in the configuration plot of the selected solution, and interpret the results of the analysis.

```
v_signal <- colnames(confusion)
v_signal <- str_replace(v_signal, " ", "")
v_signal <- str_replace(v_signal, "- -", "--")
v_signal <- str_replace(v_signal, "· -", "·-")
v_signal <- str_replace(v_signal, "· ·", "··")
v_signal <- str_replace(v_signal, "- ·", "-·")
alphabet <- c(letters,1:9,0)
another_matrix <- data.frame(v_signal) %>% mutate(alph = alphabet,
         size = nchar(trimws(v_signal)),
         ndots =  str_count(v_signal, "·"),
         nlines = str_count(v_signal, "-"),
         sig_s_length = (str_count(v_signal, "·")*0.05)+(str_count(v_signal,"-")*0.15),
```

```
        t_s_length = (str_count(v_signal, "·")*0.05) +
             (str_count(v_signal,"-")*0.15) +
          ((nchar(trimws(v_signal))-1 )*0.05),
        dots_s_length = (str_count(v_signal,"·")*0.05),
        lines_s_length = (str_count(v_signal,"-")*0.15)) %>% print()
```

```
##    v_signal alph size ndots nlines sig_s_length t_s_length dots_s_length
## 1        ·-    a    2     1      1          0.20       0.25          0.05
## 2      -···    b    4     3      1          0.30       0.45          0.15
## 3      -·-·    c    4     2      2          0.40       0.55          0.10
## 4       -··    d    3     2      1          0.25       0.35          0.10
## 5         ·    e    1     1      0          0.05       0.05          0.05
## 6      ··-·    f    4     3      1          0.30       0.45          0.15
## 7       --·    g    3     1      2          0.35       0.45          0.05
## 8     ··· ·    h    5     4      0          0.20       0.40          0.20
## 9        ..    i    2     0      0          0.00       0.05          0.00
## 10    ·-- -    j    5     1      3          0.50       0.70          0.05
## 11      -·-    k    3     1      2          0.35       0.45          0.05
## 12     ·-··    l    4     3      1          0.30       0.45          0.15
## 13       --    m    2     0      2          0.30       0.35          0.00
## 14       -·    n    2     1      1          0.20       0.25          0.05
## 15      ---    o    3     0      3          0.45       0.55          0.00
## 16     ·--·    p    4     2      2          0.40       0.55          0.10
## 17     --·-    q    4     1      3          0.50       0.65          0.05
## 18      ·-·    r    3     2      1          0.25       0.35          0.10
## 19      ···    s    3     3      0          0.15       0.25          0.15
## 20        -    t    1     0      1          0.15       0.15          0.00
## 21      ··-    u    3     2      1          0.25       0.35          0.10
## 22     ···-    v    4     3      1          0.30       0.45          0.15
## 23      ·--    w    3     1      2          0.35       0.45          0.05
## 24     -··-    x    4     2      2          0.40       0.55          0.10
## 25     -·--    y    4     1      3          0.50       0.65          0.05
## 26     --··    z    4     2      2          0.40       0.55          0.10
## 27  ·-- - -    1    7     1      4          0.65       0.95          0.05
## 28   ··-- -    2    6     2      3          0.55       0.80          0.10
## 29    ···--    3    5     3      2          0.45       0.65          0.15
## 30   ··· ·-    4    6     4      1          0.35       0.60          0.20
## 31  ··· · ·    5    7     5      0          0.25       0.55          0.25
## 32  -·· · ·    6    7     4      1          0.35       0.65          0.20
## 33   --·· ·    7    6     3      2          0.45       0.70          0.15
## 34    ---··    8    5     2      3          0.55       0.75          0.10
## 35   --- -·    9    6     1      4          0.65       0.90          0.05
## 36  --- - -    0    7     0      5          0.75       1.05          0.00
##    lines_s_length
## 1            0.15
## 2            0.15
## 3            0.30
## 4            0.15
## 5            0.00
## 6            0.15
## 7            0.30
## 8            0.00
## 9            0.00
```

```
## 10          0.45
## 11          0.30
## 12          0.15
## 13          0.30
## 14          0.15
## 15          0.45
## 16          0.30
## 17          0.45
## 18          0.15
## 19          0.00
## 20          0.15
## 21          0.15
## 22          0.15
## 23          0.30
## 24          0.30
## 25          0.45
## 26          0.30
## 27          0.60
## 28          0.45
## 29          0.30
## 30          0.15
## 31          0.00
## 32          0.15
## 33          0.30
## 34          0.45
## 35          0.60
## 36          0.75
```

```
#bimorse <- biplotmds(sim, another_matrix)

#plotMDS(dissim)

#fit two-dimensional indscal model assuming different measurement levels
#m1<-smacofIndDiff(dissim, type="ratio", constraint="indscal",ties="primary")
#m2<-smacofIndDiff(dissim, type="interval", constraint="indscal",ties="primary")
#m3<-smacofIndDiff(dissim, type="ordinal", constraint="indscal",ties="primary")
#round(c(m1$stress,m2$stress,m3$stress),3)

#res <- mds(dissim)
#fitbi <- biplotmds(res, morsescales[,2:3])
#plot(fitbi, main = "MDS Biplot", vecscale = 0.5)
#alphabet <- c(letters,1:9,0)

#confusion2 <- confusion
#colnames(confusion2) <- paste(colnames(confusion), "=", c(letters,1:9,0))
#rownames(confusion)

#compute MDS biplot: run multivariate linear regression of
#external variables on configuration

#fitFace <- mds(dissim, type = "ordinal")

#biFace <- biplotmds(fitFace, another_matrix[,-c(1:2)])
```

```
#plot(biFace, main = "Biplot Vector Representation",
#     vecscale = 0.8, xlim = c(-1.5, 1.5),
#     vec.conf = list(col = "brown"), pch = 20, cex = 0.5)
```