Powering Dissimilarities in Metric MDS

Jan de Leeuw - University of California Los Angeles

Started June 16 2023, Version of July 18, 2023

Abstract

For each integer p there is an $r > 0$ such that for all $s \geq r$ the Hadamard power Δ^s has Gower rank less than or equal to p.

Contents

Note: This is a working paper which will be expanded/updated frequently. All suggestions for improvement are welcome.

1 Introduction

In Full Dimensional Multidimensional Scaling (FMDS) we minimize (metric, Euclidean) stress over all configurations of n points in $n-1$ dimensions. It was shown in De Leeuw (1993) that in FMDS there is a single local minimum, which is consequently global. There is more detail in De Leeuw, Groenen, and Mair (2016) and De Leeuw (2019a). In smacof notation (De Leeuw and Mair (2009), Mair, Groenen, and De Leeuw (2022)) a necessary and sufficient solution for X to be the global minimum of the FMDS problem is $X=V^+B(X)X$ and $V^+B(X)\lesssim I$ (all eigenvalues less than or equal to one). At the global minimum of FDS the rank of X is the Gower Rank of the dissimilarities (De Leeuw (2016)), which is also the number of eigenvalues of $V^+B(X)$ equal to one.

In p-dimensional multidimensional scaling (pMDS) we minimize stress over all $n\times p$ configurations, where generally p is much smaller than $n-1$. The condition $X=V^+B(X)X$ remains necessary for a local minimum, but it is no longer sufficient, and in pMDS there can be (and usually are) many local minima. Finding the global minimum is a difficult problem for which only approximate and rather tentative approaches are available.

From the MDS theory we know that if in pMDS solution $V^+B(X)\lesssim I$ then the pMDS solution has its global minimum at $X, because in that case the FMDS and the pMDS solution are identical (De Leeuw (2014)). This sufficient condition for a global minimum is, however, very seldom met in pMDS.

In De Leeuw (2019b) we use FMDS to approximate the global minimum of pMDS, using a quadratic penalty function that ultimate ensures that the final $n-p$ dimensions of the FDS solution are zero. So far, this has been relatively successfull, although there are, of course, no guarantees.

In this paper we lay the foundation for a second way to globally minimize stress in pMDS using FMDS.

2 Data

The example in this paper uses the "wish" data from the smacof package, also analyzed in Kruskal and Wish (1978). The data are averages over 18 subjects of the similarity judgments between 12 countries. We converted to dissimilarities by subtracting the similarity averages from seven. All weights in the stress loss function were set equal to one.

```r
data(wish, package = "smacof")
countries <- as.matrix(wish)
```

```r
w <- matrix(1, 12, 12) - diag(12)
countries <- 7 * w - countries
mPrint(countries, digits = 2)
```

```
##          BRAZIL   CONGO    CUBA     EGYPT    FRANCE   INDIA    ISRAEL
## BRAZIL      +0.00    +2.17    +1.72    +3.56    +2.28    +2.50    +3.17
## CONGO       +2.17    +0.00    +2.44    +2.00    +3.00    +2.17    +3.67
## CUBA        +1.72    +2.44    +0.00    +1.83    +2.89    +3.00    +3.39
## EGYPT       +3.56    +2.00    +1.83    +0.00    +2.22    +1.17    +2.33
## FRANCE      +2.28    +3.00    +2.89    +2.22    +0.00    +3.56    +3.00
## INDIA       +2.50    +2.17    +3.00    +1.17    +3.56    +0.00    +2.89
## ISRAEL      +3.17    +3.67    +3.39    +2.33    +3.00    +2.89    +0.00
## JAPAN       +3.50    +3.61    +4.06    +3.17    +2.78    +2.50    +2.17
## CHINA       +4.61    +3.00    +1.50    +2.61    +3.33    +2.89    +4.00
## RUSSIA      +3.94    +3.61    +1.56    +2.61    +1.94    +2.50    +2.83
## USA         +1.61    +4.61    +3.83    +3.67    +1.06    +2.72    +1.06
## YUGOSLAV    +3.83    +3.50    +1.89    +2.72    +2.28    +3.00    +2.56
##          JAPAN    CHINA    RUSSIA   USA      YUGOSLAV
## BRAZIL      +3.50    +4.61    +3.94    +1.61    +3.83
## CONGO       +3.61    +3.00    +3.61    +4.61    +3.50
## CUBA        +4.06    +1.50    +1.56    +3.83    +1.89
## EGYPT       +3.17    +2.61    +2.61    +3.67    +2.72
## FRANCE      +2.78    +3.33    +1.94    +1.06    +2.28
## INDIA       +2.50    +2.89    +2.50    +2.72    +3.00
## ISRAEL      +2.17    +4.00    +2.83    +1.06    +2.56
## JAPAN       +0.00    +2.83    +2.39    +0.94    +2.72
## CHINA       +2.83    +0.00    +1.28    +4.44    +1.94
## RUSSIA      +2.39    +1.28    +0.00    +2.00    +0.33
## USA         +0.94    +4.44    +2.00    +0.00    +3.44
## YUGOSLAV    +2.72    +1.94    +0.33    +3.44    +0.00
```

3 Results

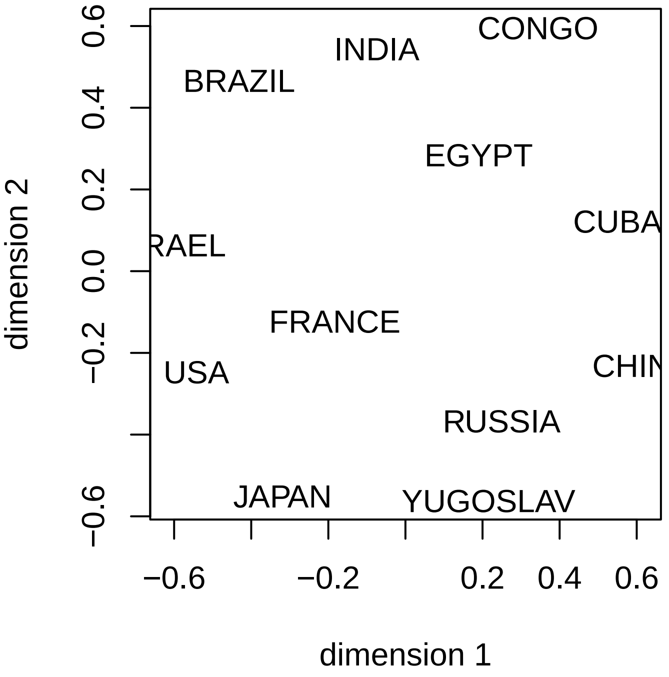
We are interested in what happens to both FMDS and pMDS solutions (with $p = 2$) when we analyze dissimilarities that are transformed by raising them to different powers. Empirically we have found that the Gower rank of powered dissimilarities decreases as a function of the power r, commonly starting near $n - 1$ for $r = 1$ and ending up at one for large r. In a sense, this is not too surprising. For large r the dissimilarity matrix will be dominated by the largest dissimilarity, and the points with the largest dissimilarity can be placed at the endpoints of a line, with the rest in between. What is surprising is that for relatively low powers (say $r < 10$) we have a low Gower rank and still plenty of detail in the solution, which means that pMDS with p equal to the Gower rank finds the global minimum.

We give the results for a sequence of powers on the Wish example, for now with little comment.

The smacof solutions all start with the classical Torgerson-Gower solution and iterate until the stress values from one iteration to the next change by less than $1e-15$.

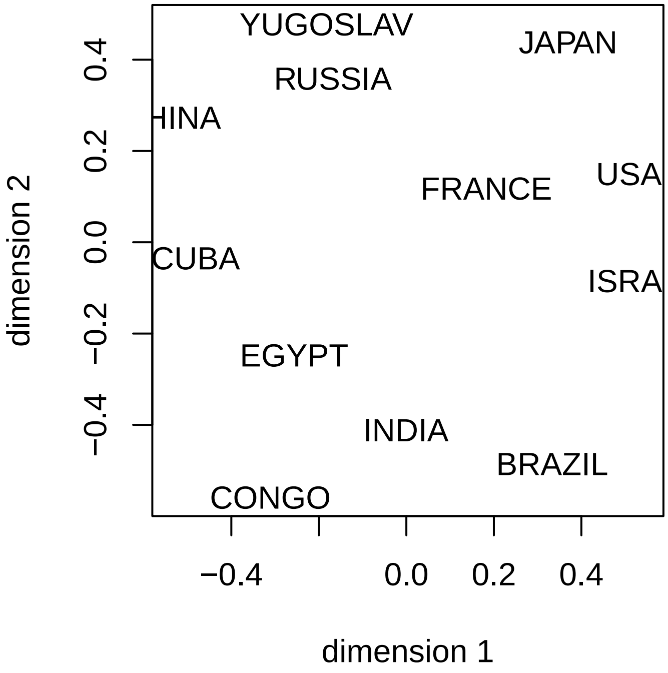
3.1 r = .25

```
## RESULTS FOR 0.2500
## [1] "raw stress for full dimensional and two dimensional"
## 0.000000000000000 2.084589217452540
## [1] "normalized stress for full dimensional and two dimensional"
## 0.000000000000000 0.020986853099850
## [1] "singular values X for full dimensional"
##  [1] +0.911823 +0.848901 +0.733064 +0.636770 +0.621680 +0.584975 +0.577622
##  [8] +0.485632 +0.406526 +0.349673 +0.240971
## [1] "singular values X for two dimensional"
## [1] +1.394606 +1.358795
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +1.000000 +1.000000 +1.000000 +1.000000 +1.000000 +1.000000
##  [8] +1.000000 +1.000000 +1.000000 +1.000000 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.510326 +1.472451 +1.413206 +1.382267 +1.333851 +1.316327 +1.259150
##  [8] +1.198866 +1.184617 +1.000000 +1.000000 +0.000000
```



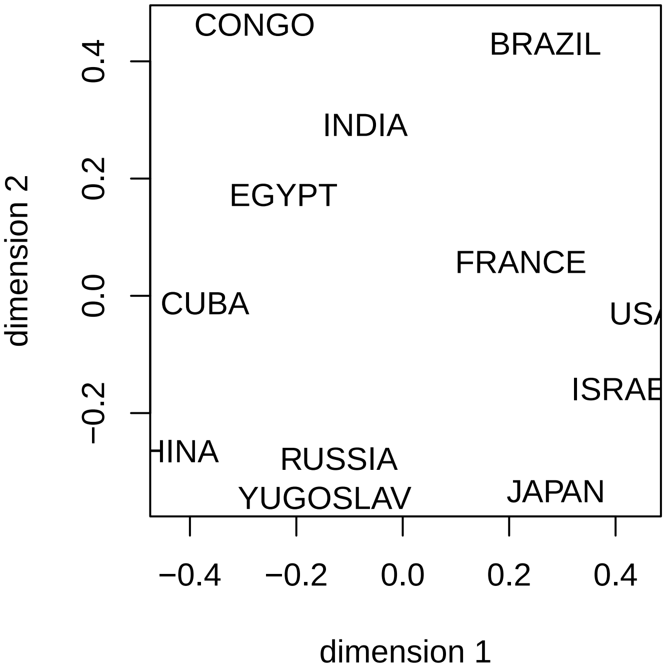
3.2 r = .50

```
## RESULTS FOR 0.5000
## [1] "raw stress for full dimensional and two dimensional"
## 0.016813189776687 1.176073277740564
## [1] "normalized stress for full dimensional and two dimensional"
## 0.000217318468206 0.015201306034834
## [1] "singular values X for full dimensional"
##  [1] +0.958007 +0.871687 +0.704340 +0.566183 +0.522508 +0.463092 +0.430679
##  [8] +0.231413 +0.005028 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +1.257791 +1.202314
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +1.000000 +1.000000 +1.000000 +1.000000 +1.000000 +1.000000
##  [8] +1.000000 +0.999846 +0.983008 +0.965245 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.457879 +1.384164 +1.352360 +1.306342 +1.274894 +1.237496 +1.180982
##  [8] +1.139662 +1.126206 +1.000000 +1.000000 +0.000000
```



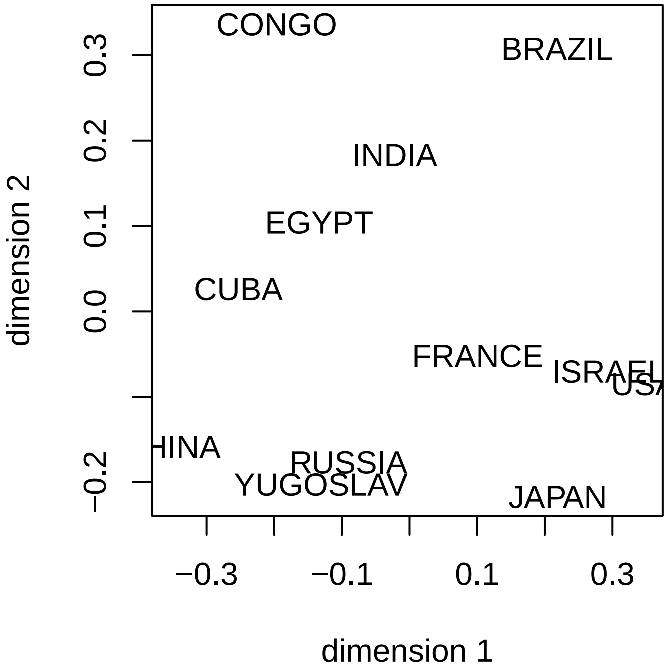
3.3 r = 1

```
## RESULTS FOR 1.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.201887542090101 0.603629942647604
## [1] "normalized stress for full dimensional and two dimensional"
## 0.003992489316306 0.011937269987403
## [1] "singular values X for full dimensional"
##  [1] +0.912077 +0.805746 +0.579953 +0.389891 +0.288252 +0.144025 +0.000002
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +1.044677 +0.956555
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +1.000000 +1.000000 +1.000000 +1.000000 +1.000000 +0.995184
##  [8] +0.959376 +0.918836 +0.898833 +0.849330 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.374950 +1.294842 +1.234365 +1.186110 +1.107626 +1.076704 +1.037704
##  [8] +1.004519 +1.000000 +1.000000 +0.954487 +0.000000
```



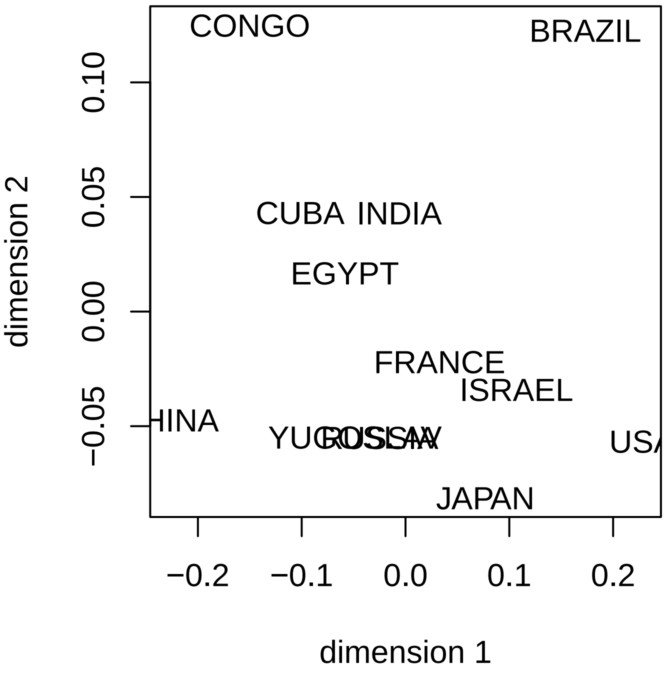
3.4 r = 2

```
## RESULTS FOR 2.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.600836884949776 0.660808460717618
## [1] "normalized stress for full dimensional and two dimensional"
## 0.022642989456655 0.024903063349961
## [1] "singular values X for full dimensional"
##  [1] +0.730826 +0.598564 +0.320623 +0.101539 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.766584 +0.638631
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +1.000000 +1.000000 +1.000000 +0.945852 +0.909598 +0.867725
##  [8] +0.837378 +0.767077 +0.717658 +0.664230 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.269220 +1.103698 +1.039430 +1.000000 +1.000000 +0.967035 +0.907976
##  [8] +0.860553 +0.805728 +0.775803 +0.689188 +0.000000
```



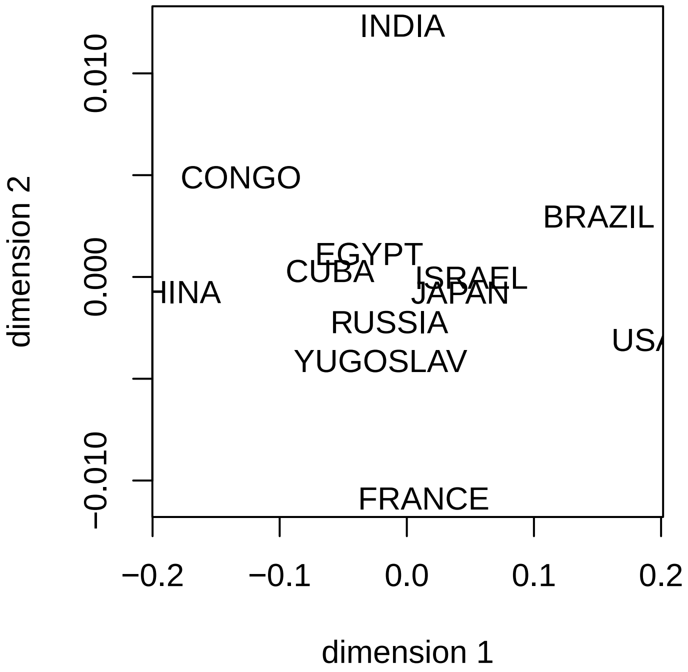
3.5 r = 5

```
## RESULTS FOR 5.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.889942897327712 0.890439775666370
## [1] "normalized stress for full dimensional and two dimensional"
## 0.094053041997496 0.094105554264737
## [1] "singular values X for full dimensional"
##  [1] +0.437804 +0.228253 +0.046478 +0.000000 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.438715 +0.231037
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +1.000000 +1.000000 +0.815716 +0.752369 +0.639559 +0.587647
##  [8] +0.531858 +0.516145 +0.458722 +0.387285 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.072992 +1.000000 +1.000000 +0.824166 +0.759072 +0.645328 +0.588648
##  [8] +0.540184 +0.518484 +0.458735 +0.388575 +0.000000
```



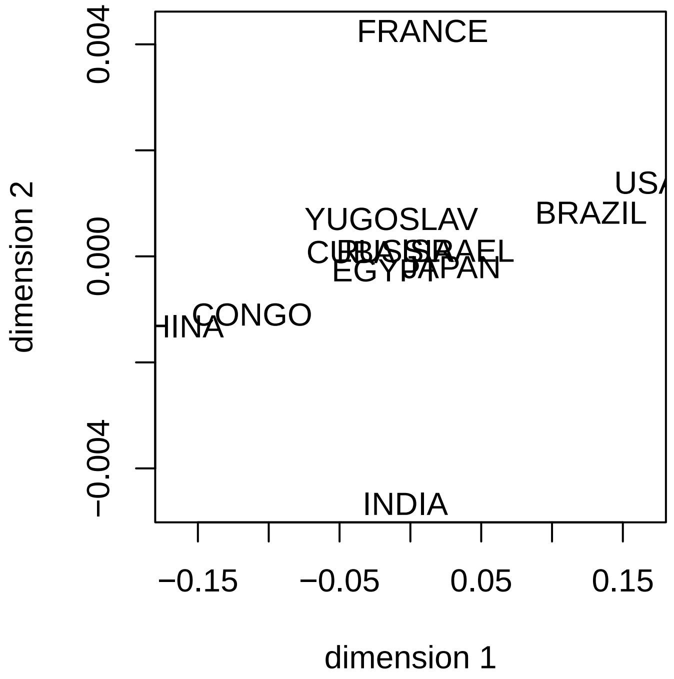
3.6 r = 8

```
## RESULTS FOR 8.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.861528292644553 0.861528292644499
## [1] "normalized stress for full dimensional and two dimensional"
## 0.136700436161952 0.136700436161944
## [1] "singular values X for full dimensional"
##  [1] +0.344493 +0.018232 +0.000001 +0.000000 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.344493 +0.018232
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +1.000000 +0.988788 +0.809165 +0.576103 +0.543085 +0.472001
##  [8] +0.388677 +0.344025 +0.307488 +0.239859 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.000000 +1.000000 +0.988788 +0.809165 +0.576102 +0.543084 +0.472001
##  [8] +0.388676 +0.344025 +0.307488 +0.239859 +0.000000
```



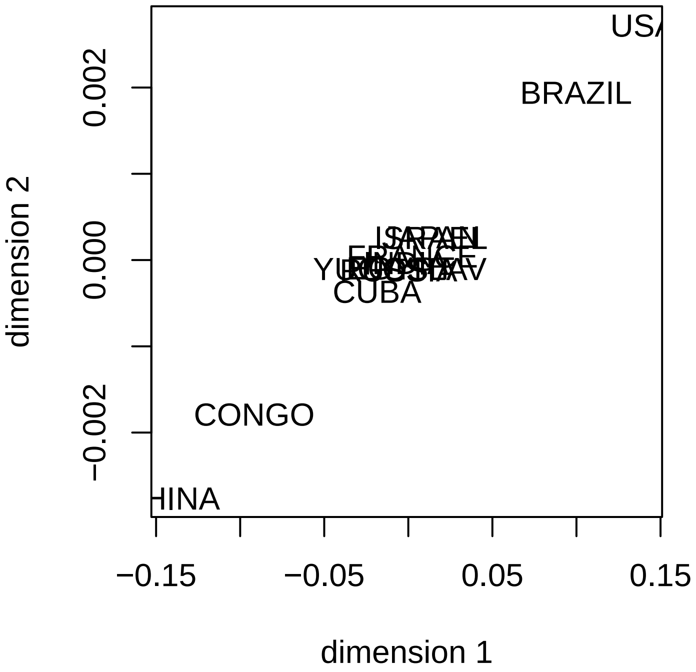
3.7 r = 10

```
## RESULTS FOR 10.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.849049190259466 0.849049190259466
## [1] "normalized stress for full dimensional and two dimensional"
## 0.153489609905732 0.153489609905732
## [1] "singular values X for full dimensional"
##  [1] +0.298222 +0.006342 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.298222 +0.006342
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +1.000000 +0.911000 +0.785987 +0.475049 +0.418898 +0.357726
##  [8] +0.298582 +0.250702 +0.201013 +0.158641 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.000001 +1.000000 +0.911000 +0.785987 +0.475049 +0.418898 +0.357726
##  [8] +0.298583 +0.250702 +0.201013 +0.158641 +0.000000
```



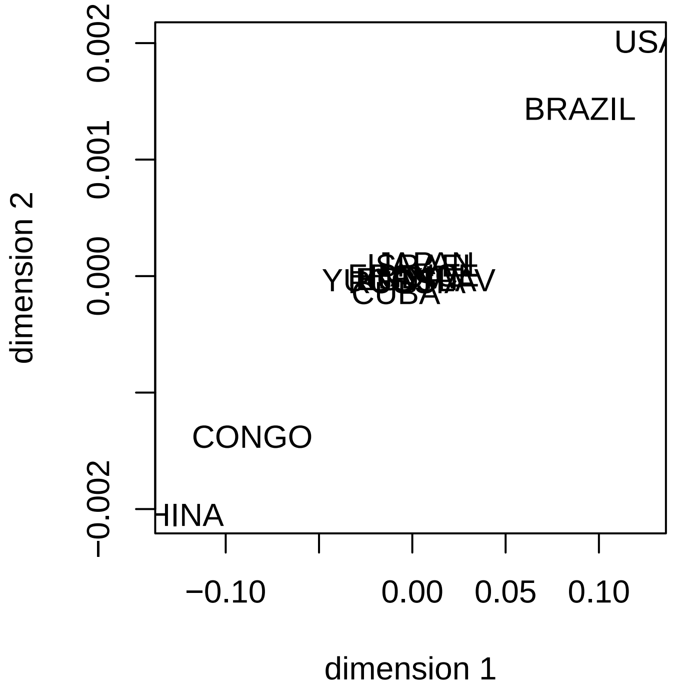
3.8 r = 15

```
## RESULTS FOR 15.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.838657370199356 0.838657370199360
## [1] "normalized stress for full dimensional and two dimensional"
## 0.176036341301155 0.176036341301156
## [1] "singular values X for full dimensional"
##  [1] +0.242340 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.242340 +0.000000
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +0.927400 +0.849990 +0.794779 +0.273187 +0.241039 +0.169889
##  [8] +0.130252 +0.101554 +0.072569 +0.042251 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.000000 +0.927400 +0.849990 +0.794779 +0.273187 +0.241039 +0.169889
##  [8] +0.130252 +0.101554 +0.072569 +0.042251 -0.000000
```



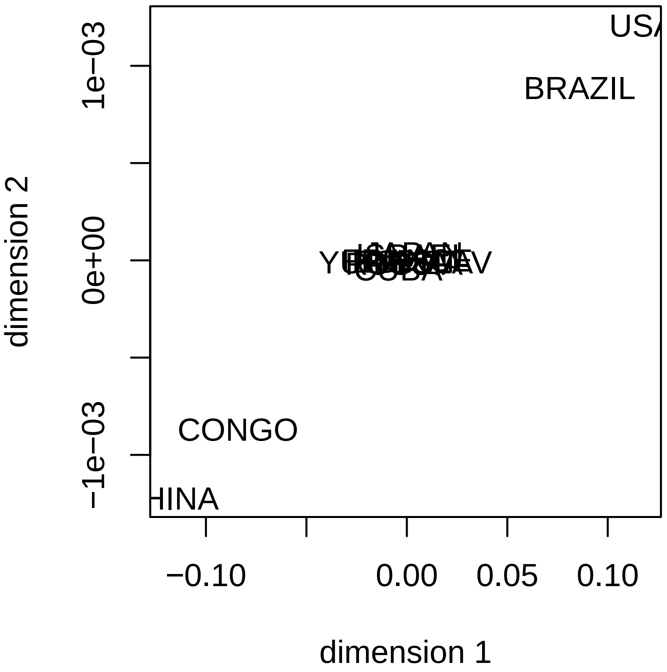
3.9 r = 20

```
## RESULTS FOR 20.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.831422757862696 0.831422757862693
## [1] "normalized stress for full dimensional and two dimensional"
## 0.185948497873557 0.185948497873557
## [1] "singular values X for full dimensional"
##  [1] +0.218476 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.218476 +0.000000
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +0.913874 +0.860575 +0.816735 +0.176723 +0.078394 +0.054922
##  [8] +0.045072 +0.026352 +0.024002 +0.009839 -0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.000000 +0.913874 +0.860575 +0.816735 +0.176723 +0.078394 +0.054922
##  [8] +0.045072 +0.026352 +0.024002 +0.009839 -0.000000
```



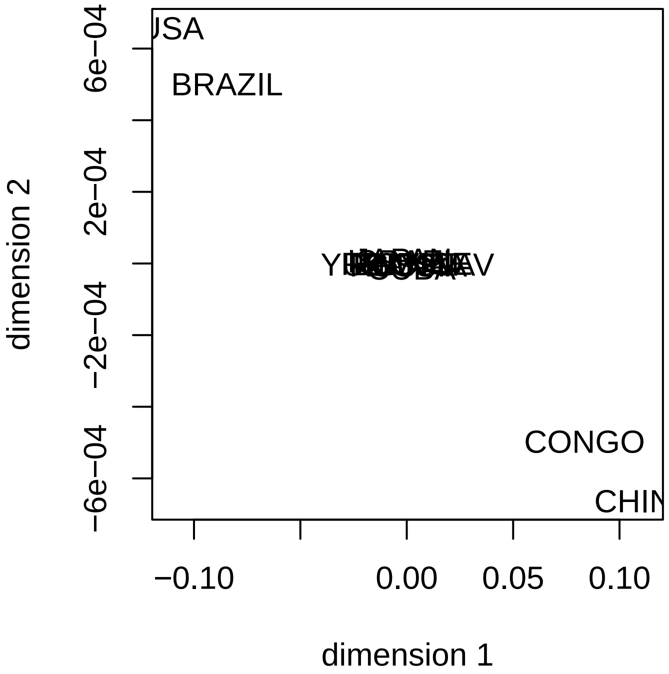
3.10 r = 25

```
## RESULTS FOR 25.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.824730694057211 0.824730694057208
## [1] "normalized stress for full dimensional and two dimensional"
## 0.191265785156373 0.191265785156373
## [1] "singular values X for full dimensional"
##  [1] +0.205451 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.205451 +0.000000
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +0.925980 +0.888078 +0.839451 +0.136030 +0.037088 +0.024640
##  [8] +0.015599 +0.009970 +0.006826 +0.002460 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.000000 +0.925980 +0.888078 +0.839451 +0.136030 +0.037088 +0.024640
##  [8] +0.015599 +0.009970 +0.006826 +0.002460 +0.000000
```



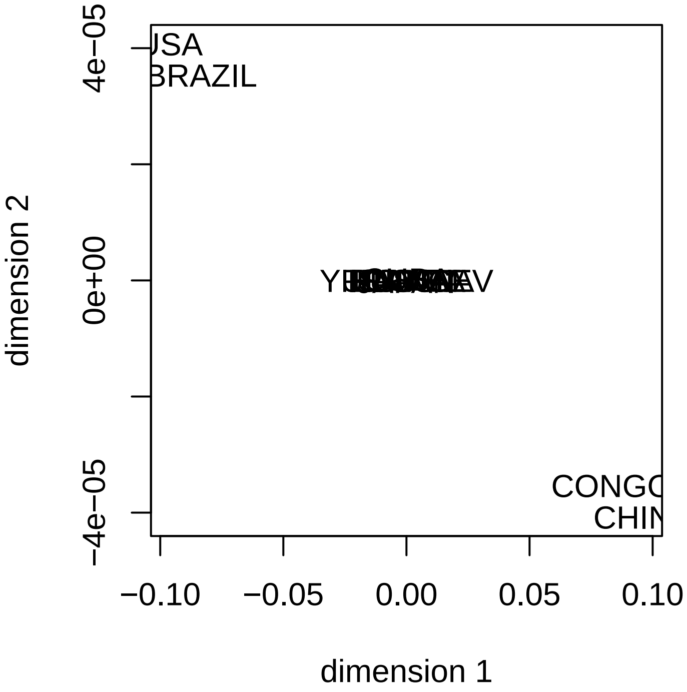
3.11 r = 30

```
## RESULTS FOR 30.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.819914013028502 0.819914013028503
## [1] "normalized stress for full dimensional and two dimensional"
## 0.194685562943146 0.194685562943146
## [1] "singular values X for full dimensional"
##  [1] +0.197043 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.197043 +0.000000
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +0.941888 +0.913793 +0.861674 +0.112800 +0.017815 +0.011144
##  [8] +0.005756 +0.003828 +0.002049 +0.000637 -0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.000000 +0.941888 +0.913793 +0.861674 +0.112800 +0.017815 +0.011144
##  [8] +0.005756 +0.003828 +0.002049 +0.000637 +0.000000
```



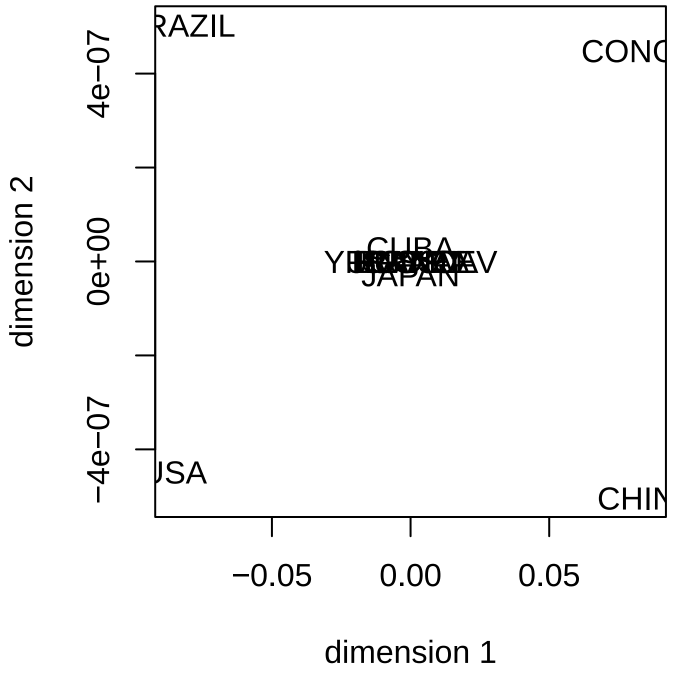
3.12 r = 50

```
## RESULTS FOR 50.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.817464715285994 0.817464715286003
## [1] "normalized stress for full dimensional and two dimensional"
## 0.202007792226659 0.202007792226661
## [1] "singular values X for full dimensional"
##  [1] +0.179912 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.179912 +0.000000
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +0.982760 +0.972614 +0.929044 +0.061865 +0.001017 +0.000469
##  [8] +0.000132 +0.000092 +0.000018 +0.000004 +0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.000000 +0.982758 +0.972614 +0.929044 +0.061865 +0.001017 +0.000469
##  [8] +0.000132 +0.000092 +0.000018 +0.000004 +0.000000
```



3.13 r = 100

```
## RESULTS FOR 100.0000
## [1] "raw stress for full dimensional and two dimensional"
## 0.829669422356802 0.829669422356802
## [1] "normalized stress for full dimensional and two dimensional"
## 0.207360848138136 0.207360848138136
## [1] "singular values X for full dimensional"
##  [1] +0.168623 +0.000001 +0.000000 +0.000000 +0.000000 +0.000000 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000
## [1] "singular values X for two dimensional"
## [1] +0.168623 +0.000001
## [1] "eigenvalues B(X) for full dimensional"
##  [1] +1.000000 +0.999661 +0.998675 +0.988462 +0.011275 +0.000001 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000 -0.000000
## [1] "eigenvalues B(X) for two dimensional"
##  [1] +1.000461 +1.000000 +0.998542 +0.988462 +0.011275 +0.000001 +0.000000
##  [8] +0.000000 +0.000000 +0.000000 +0.000000 -0.000000
```



4 Discussion

4.1 Global pMDS

4.2 Nonmetric MDS

5 Appendix: Code

5.1 traceGower.R

```
traceGower <- function(data, power) {
  pful <- nrow(data) - 1
  data <- data ^ power
  data <- data / max(data)
  deno <- sum(data ^ 2)
  h <- smacof(w, data, p = pful, eps = 1e-15)
  g <- smacof(w, data, p = 2, eps = 1e-15)
  cat("RESULTS FOR", formatC(power, digits = 4, format = "f"), "\n")
  print("raw stress for full dimensional and two dimensional")
  cat(formatC(c(h$s , g$s), digits = 15, format = "f"), "\n")
  print("normalized stress for full dimensional and two dimensional")
  cat(formatC(c(h$s / deno, g$s / deno), digits = 15, format = "f"), "\n")
  print("singular values X for full dimensional")
  mPrint(svd(h$x)$d, d = 6)
  print("singular values X for two dimensional")
  mPrint(svd(g$x)$d, d = 6)
  print("eigenvalues B(X) for full dimensional")
  mPrint((eigen(h$b)$values)/12, d = 6)
  print("eigenvalues B(X) for two dimensional")
  mPrint((eigen(g$b)$values)/12, d = 6)
  par(pty = "s")
  plot(g$x, type = "n", xlab = "dimension 1", ylab = "dimension 2")
  text(g$x, row.names(countries))
}
```

5.2 smacof.R

```
source("smacofBasics.R")


smacof <-
  function(w,
           delta,
           p = 2,
           xold = torgerson(delta, p),
           itmax = 10000,
           eps = 1e-10,
           verbose = FALSE) {
    itel <- 1
    n <- nrow(xold)
    vmat <- smacofVmat(w)
    vinv <- solve(vmat + (1 / n)) - (1 / n)
    dold <- as.matrix(dist(xold))
    mold <- sum(w * delta * dold) / sum(w * dold * dold)
    xold <- xold * mold
    dold <- dold * mold
    sold <- smacofLoss(dold, w, delta)
    bold <- smacofBmat(dold, w, delta)
    repeat {
      xnew <- smacofGuttman(xold, bold, vinv)
      dnew <- as.matrix(dist(xnew))
      bnew <- smacofBmat(dnew, w, delta)
      snew <- smacofLoss(dnew, w, delta)
      if (verbose) {
        cat(
          "itel ",
          formatC(itel, width = 4, format = "d"),
          "sold ",
          formatC(
            sold,
            width = 10,
            digits = 6,
            format = "f"
          ),
          "snew ",
          formatC(
            snew,
            width = 10,
            digits = 6,
            format = "f"
          ),
         "\n"
        )
      }
      if (((sold - snew) < eps) || (itel == itmax)) {
        break
      }
      itel <- itel + 1
      xold <- xnew
      bold <- bnew
      sold <- snew
     }
    return(list(
      x = xnew,
      d = dnew,
      b = bnew,
      s = snew,
      itel = itel
    ))
  }
```

5.3 smacofBasics.R

```
smacofLoss <- function(d, w, delta) {
  return(sum(w * (delta - d) ^ 2) / 4)
}

smacofBmat <- function(d, w, delta) {
  dd <- ifelse(d == 0, 0, 1 / d)
  b <- -dd * w * delta
  diag(b) <- -rowSums(b)
  return(b)
}

smacofVmat <- function(w) {
  v <- -w
  diag(v) <- -rowSums(v)
  return(v)
}

smacofGuttman <- function(x, b, vinv) {
  return(vinv %*% b %*% x)
}

doubleCenter <- function(x) {
  rs <- apply(x, 1, mean)
  ss <- mean(x)
  return(x - outer(rs, rs, "+") + ss)
}

columnCenter <- function(x) {
  return(apply(x, 2, function(z) z - mean(z)))
}

torgerson <- function(delta, p) {
  e <- eigen(-.5 * doubleCenter(delta ^ 2))
  l <- sqrt(pmax(0, e$values[1:p]))
  return(e$vectors[, 1:p] %*% diag(l))
}
```

5.4 janUtil.R

```
mPrint <- function(x,
                   digits = 6,
                   width = 8,
                   format = "f",
                   flag = "+") {
  print(noquote(
    formatC(
      x,
      digits = digits,
      width = width,
      format = format,
      flag = flag
    )
  ))
}

butLast <- function(x, m = 1) {
  return(rev(rev(x)[-(1:m)]))
}

butFirst <- function(x, m = 1) {
  return(x[-(1:m)])
}
```

References

De Leeuw, J. 1993. "Fitting Distances by Least Squares." Preprint Series 130. Los Angeles, CA: UCLA Department of Statistics. https://jansweb.netlify.app/publication/deleeuw-r-93-c/deleeuw-r-93-c.pdf.

———. 2014. "Bounding, and Sometimes Finding, the Global Minimum in Multidimensional Scaling." UCLA Department of Statistics. https://jansweb.netlify.app/publication/deleeuw-u-14-b/deleeuw-u-14-b.pdf.

———. 2016. "Gower Rank." 2016. https://jansweb.netlify.app/publication/deleeuw-e-16-k/deleeuw-e-16-k.pdf.

———. 2019a. "Generalized Full-dimensional Scaling." 2019. https://jansweb.netlify.app/publication/deleeuw-e-19-f/deleeuw-e-19-f.pdf.

———. 2019b. "Global Minima by Penalized Full-dimensional Scaling." 2019. https://jansweb.netlify.app/publication/deleeuw-e-19-e/deleeuw-e-19-e.pdf.

De Leeuw, J., P. Groenen, and P. Mair. 2016. "Full-Dimensional Scaling." 2016. https://jansweb.netlify.app/publication/deleeuw-groenen-mair-e-16-e/deleeuw-groenen-mair-e-16-e.pdf.

De Leeuw, J., and P. Mair. 2009. "Multidimensional Scaling Using Majorization: SMACOF in R." *Journal of Statistical Software* 31 (3): 1–30. https://www.jstatsoft.org/article/view/v031i03.

Kruskal, J. B., and M. Wish. 1978. *Multidimensional Scaling*. Sage.

Mair, P., P. J. F. Groenen, and J. De Leeuw. 2022. "More on Multidimensional Scaling in R: smacof Version 2." *Journal of Statistical Software* 102 (10): 1–47. https://www.jstatsoft.org/article/view/v102i10.