Multidimensional Scaling Examples

(datasets: eurodist, airline)

STAT 32950-24620

Spring 2023 (4/13)

MDS: Recover a map from proximity

Example 1: Given pairwise distance of 21 European cities only.

MDS works from Distance → **Configuration**:

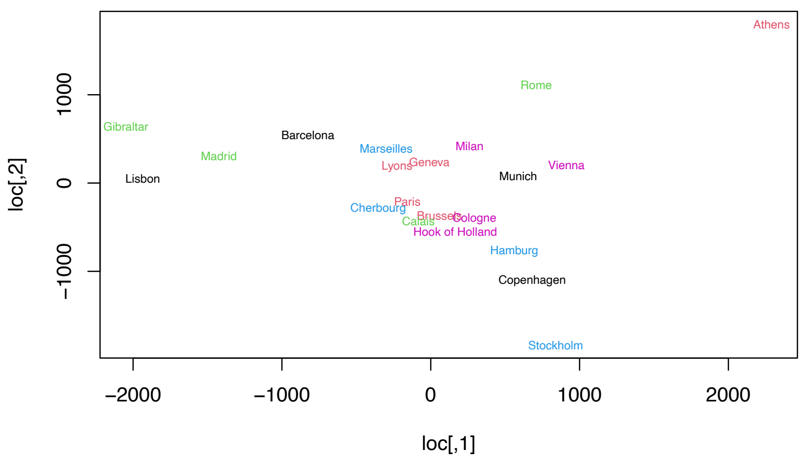
Assign coordinates preserving the orignal pairwise distances

```
require(graphics)
as.matrix(eurodist)[1:7,1:7]
```

```
##             Athens Barcelona Brussels Calais Cherbourg Co
## Athens           0      3313     2963   3175      3339
## Barcelona     3313         0     1318   1326      1294
## Brussels      2963      1318        0    204       583
## Calais        3175      1326      204      0       460
## Cherbourg     3339      1294      583    460         0
## Cologne       2762      1498      206    409       785
## Copenhagen    3276      2218      966   1136      1545
```


Map produced by MDS (Example 1)

```
loc <- cmdscale(eurodist); plot(loc, type = "n")
text(loc,rownames(loc),cex = 0.6,col=c(2,rep(c(1:4,6),5)))
```



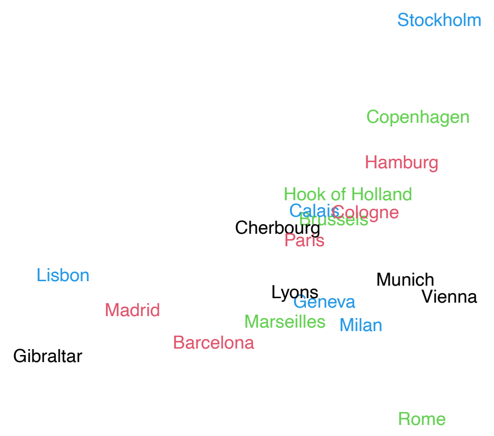
Distance to Configuration

Multidimensional Scaling (MDS):

- Start with pairwise distances of objects.
- Create a low dimensional configuration of objects.
- The objective is to best preserve the original pairwise distance.
- The exact location can not be recovered with distance info only.
- The exact orientation can not be recovered with distance info only.

Adjust directions of MDS map (Example 1)

```
loc <- cmdscale(eurodist);
x <- loc[, 1]; y <- -loc[, 2] # so North at top
plot(x, y,type = "n",xlab = "",ylab = "",asp = 1,axes =F)
text(x, y, rownames(loc), cex = 0.6, col=rep(1:4,5))
```



Example 2

Example 2: Airline-distance Data (Table 12.7 in J&W)

Data: Pairwise airline distances (in miles) between 12 U.S. cities.

The distances can be used as dissimilarity measures.

The true coordinates of the airlines are of $p > 2$ dimensions.

```
Airline = read.table("T12-7m.DAT",header=T)
class(Airline)
```

```
## [1] "data.frame"
```

```
typeof(Airline[1,1])
```

```
## [1] "integer"
```


Common distance data format

Example 2 data

```
Airline # Note: the off-diagonal zeros are misleading
```

```
##     Ata  Bos  Cin  Col  Dal  Ind  LRk  LAs  Mem  StL  Sp
## 1     0    0    0    0    0    0    0    0    0    0
## 2  1068    0    0    0    0    0    0    0    0    0
## 3   461  867    0    0    0    0    0    0    0    0
## 4   549  769  107    0    0    0    0    0    0    0
## 5   805 1819  943 1050    0    0    0    0    0    0
## 6   508  941  108  172  882    0    0    0    0    0
## 7   505 1494  618  725  325  562    0    0    0    0
## 8  2197 3052 2186 2245 1403 2080 1701    0    0    0
## 9   366 1355  502  586  464  436  137 1831    0    0
## 10  558 1178  338  409  645  234  353 1848  294    0
## 11 2467 2747 2067 2131 1891 1959 1988 1227 2042 1820
## 12  467 1379  928  985 1077  975  912 2480  779 1016 28
```

Most dissimilarity matrices are near symmetric.

The (symmetric) distance matrix from Example 2 data

```
as.dist(Airline, diag=T, upper=T)
```

```
##      Ata  Bos  Cin  Col  Dal  Ind  LRk  LAs  Mem  StL  S
## Ata    0 1068  461  549  805  508  505 2197  366  558 24
## Bos 1068    0  867  769 1819  941 1494 3052 1355 1178 27
## Cin  461  867    0  107  943  108  618 2186  502  338 20
## Col  549  769  107    0 1050  172  725 2245  586  409 21
## Dal  805 1819  943 1050    0  882  325 1403  464  645 18
## Ind  508  941  108  172  882    0  562 2080  436  234 19
## LRk  505 1494  618  725  325  562    0 1701  137  353 19
## LAs 2197 3052 2186 2245 1403 2080 1701    0 1831 1848 12
## Mem  366 1355  502  586  464  436  137 1831    0  294 20
## StL  558 1178  338  409  645  234  353 1848  294    0 18
## Spo 2467 2747 2067 2131 1891 1959 1988 1227 2042 1820
## Tpa  467 1379  928  985 1077  975  912 2480  779 1016 28
```

Fit MDS models with dimension q

Example 2 using MDS: **Distance to coordinates configuration**

Assigns $q = 4$ dimensional coordinates to the N items (the airlines)

```
Air4 = cmdscale(as.dist(Airline),k=4); round(Air4)
```

```
##       [,1] [,2] [,3] [,4]
## Ata   -540  296 -135   63
## Bos  -1111 -710  -72   30
## Cin   -361 -233   51  -47
## Col   -420 -288  -22   71
## Dal    275  491  -27  -39
## Ind   -254 -238   50    1
## LRk      4  290  -98 -113
## LAs   1704  481  -49   76
## Mem   -118  224   24  -29
## StL    -56 -110   -6  -40
## Spo   1654 -817   76   -9
## Tpa   -777  615  208   36
```


Pairwise distance in lower dimentions

Example 2: Using the q-dimensional coordinates by the model,

we can compute fitted pairwise distances of items in q-dimensional space,
such as Euclidean distance

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + \cdots + (x_q - y_q)^2}$$

For example, the fitted Euclidean distance between Atlanta and Boston in $q = 4$ dimension is

$$d^{(4)}_{Ata,Bos}$$
$$= \sqrt{(-1111 + 540)^2 + (-710 - 296)^2 + (-72 + 135)^2 + (30 - 63)^2}$$
$$= 1159$$

$q = 4$ dim. Euclidean distances by MDS model (Ex.2)

```
round(dist(Air4))
```

```
##      Ata  Bos  Cin  Col  Dal  Ind  LRk  LAs  Mem  StL  S
## Bos 1159
## Cin  598  901
## Col  606  812  161
## Dal  851 1836  967 1050
## Ind  636  987  118  201  905
## LRk  573 1505  658  744  354  616
## LAs 2253 3057 2191 2260 1434 2090 1722
## Mem  465 1368  518  604  479  483  203 1845
## StL  653 1218  334  421  687  246  421 1861  341
## Spo 2470 2771 2099 2145 1904 1994 1997 1308 2056 1852
## Tpa  526 1395  961  999 1088 1014  912 2499  791 1048 28
```


The original airline distances (to be compared with)

```
as.dist(Airline)
```

```
##      Ata  Bos  Cin  Col  Dal  Ind  LRk  LAs  Mem  StL  S
## Bos 1068
## Cin  461  867
## Col  549  769  107
## Dal  805 1819  943 1050
## Ind  508  941  108  172  882
## LRk  505 1494  618  725  325  562
## LAs 2197 3052 2186 2245 1403 2080 1701
## Mem  366 1355  502  586  464  436  137 1831
## StL  558 1178  338  409  645  234  353 1848  294
## Spo 2467 2747 2067 2131 1891 1959 1988 1227 2042 1820
## Tpa  467 1379  928  985 1077  975  912 2480  779 1016 28
```

Compare MDS $q = 4$ distances and the original dist.

```
round(dist(Air4)) - as.dist(Airline)
```

```
##     Ata Bos Cin Col Dal Ind LRk LAs Mem StL Spo
## Bos  91
## Cin 137  34
## Col  57  43  54
## Dal  46  17  24   0
## Ind 128  46  10  29  23
## LRk  68  11  40  19  29  54
## LAs  56   5   5  15  31  10  21
## Mem  99  13  16  18  15  47  66  14
## StL  95  40  -4  12  42  12  68  13  47
## Spo   3  24  32  14  13  35   9  81  14  32
## Tpa  59  16  33  14  11  39   0  19  12  32   4
```


MDS desires low dimensional representation

Based on the coordinate configuration from the model for Ex. 2,

$$\binom{N}{2} = \frac{N(N-1)}{2}$$

q dimensional pair-ewise distance between N items can be derived.

which are compared with the original distances givne in the data.

The original objects can be viewed as from $p = N - 1 = 11$ dimensional space.

The desire is a good-enough preservation of the distances

in $q = 2$ or $q = 3$ dimensions.

Fit MDS with dimension $q = 2$

The most commonly used dimension in MDS is $q = 2$

The resulting model assigns 2-dim. coordinates to the N items.

```
Air2 = cmdscale(as.dist(Airline),k=2); round(Air2)
```

```
##      [,1] [,2]
## Ata  -540  296
## Bos -1111 -710
## Cin  -361 -233
## Col  -420 -288
## Dal   275  491
## Ind  -254 -238
## LRk     4  290
## LAs  1704  481
## Mem  -118  224
## StL   -56 -110
## Spo  1654 -817
## Tpa  -777  615
```


MDS (background PCA)

MDS produces Euclidean distance

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$$

e.g., in Example 2,

$$d^{(2)}_{Ata,Bos} = \sqrt{(-1111 + 540)^2 + (-710 - 296)^2} = 1157$$

Compare with the first two coordinates $q = 4$ results.

Note: PCA is in the background of classical metric MDS.

From model fitted q-coordinates to distance

Use 2-norm distances: (Euclidean)

$$d_2(x, y) = \left((x_1 - y_1)^2 + \cdots + (x_q - y_q)^2\right)^{1/2}$$

```
round(dist(Air2)) - as.dist(Airline)
```

```
##     Ata Bos Cin Col Dal Ind LRk LAs Mem StL Spo
## Bos  89
## Cin  97  22
## Col  46  41 -26
## Dal  33  15  21  -6
## Ind  97  37  -1   2  19
## LRk  38   3  19  -9  13  25
## LAs  54   5  -1  14  26   6  10
## Mem  61   8  15   8  12  45   2  10
## StL  73  36  -9  -3  41   2  51   9  46
## Spo  -7  20  31  10  10  35  -1  72  14  30
## Tpa -69 -12  17 -14 -17  26 -66   5 -13   7   1
```


Try a non-Euclidean distance

Use 3-norm distances:

$$d_3(x, y) = \left((x_1 - y_1)^3 + \cdots + (x_q - y_q)^3\right)^{1/3}$$

```
round(dist(Air2,method="minkowski",p=3))-as.dist(Airline)
```

```
##      Ata  Bos  Cin  Col  Dal  Ind  LRk  LAs  Mem  StL  S
## Bos   -4
## Cin   74  -57
## Col   36  -29  -35
## Dal   14 -181  -83 -118
## Ind   51  -39   -1   -4  -70
## LRk   38 -158  -42  -80  -21  -15
## LAs   47 -167  -93  -87   26  -90    0
## Mem   56 -140  -24  -41  -33   30   -9   -7
## StL    7  -62  -26  -31  -12  -19   47  -66   41
## Spo -182   18  -36  -45 -197  -33 -186   71 -157  -71
## Tpa -108  -46  -48  -64  -24  -61 -113    2  -77 -105 -2
```


Objectives of MDS

MDS gives a q-dimensional approximation

of the original p dimensional data,

aiming to conserve pairwise distances.

Dimension reduction:

$$q << p$$

For $q = 2$, the model approximation produces a 2-dimensional map.

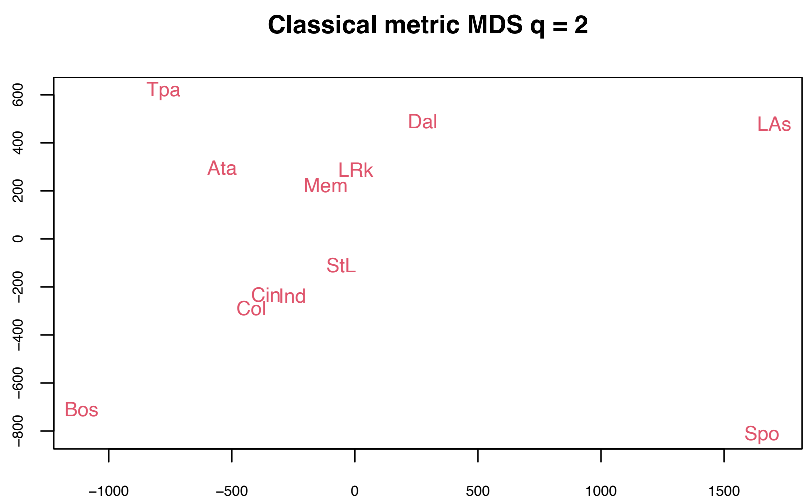
Plot MDS maps

```
plot(Air2[,1],Air2[,2],type="n",
     xlab = "", ylab = "", cex.axis=.7)  # rev dirs!
text(Air2[,1],Air2[,2],colnames(Airline),
     cex=.9,lwd=2,col=2)
title(main="Classical metric MDS q = 2")
```

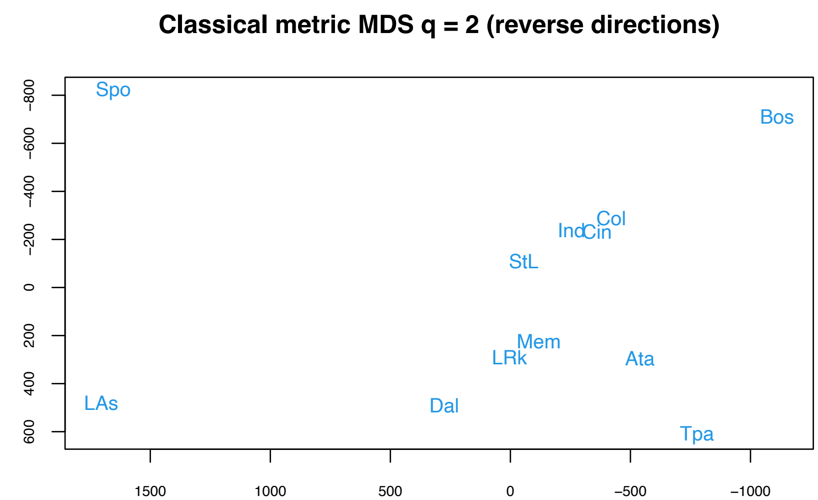
Reverse axes:

```
plot(Air2[,1],Air2[,2],type="n",
     xlab = "", ylab = "", asp = 1, cex.axis=.7,
     xlim=rev(range(Air2[,1])), ylim=rev(range(Air2[,2])))
text(Air2[,1],Air2[,2],colnames(Airline),
     cex=.9,lwd=2,col=4)  # ~ geo location
title(main=
    "Classical metric MDS q = 2 (reverse directions)")
```

MDS map q=2



MDS map q=2 (reversed axes)



Fit MDS with min. dimension q=1

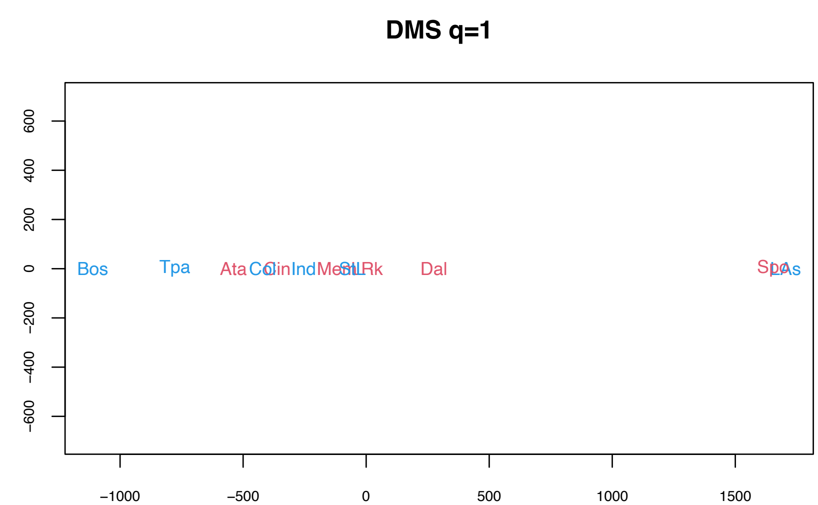
(not always desirable)

```
#AirEig = cmdscale(as.dist(Airline),2,eig=T)
#barplot(AirEig$eig)
Air1 = cmdscale(as.dist(Airline),1)
#round(Air1, 1)  # the same as the first column of Air2
```

The resulting model assigns 1-dimensional coordinates to each of the N item.

```
plot(Air1,rep(1,12),type="n",
     xlab = "", ylab = "", asp = 1,cex.axis=.7)
text(Air1,rep(1,12),colnames(Airline),
     cex=.8,col=rep(c(2,4),6)); title("DMS q=1")
```


Plot MDS q=1



Compared MDS q=1 with the original distances

```r
# order(Air1)
round(dist(Air1)) - as.dist(Airline)
```

```
##        Ata   Bos   Cin   Col   Dal   Ind   LRk   LAs   Me
## Bos   -497
## Cin   -283  -117
## Col   -430   -79   -48
## Dal     10  -433  -306  -354
## Ind   -223   -84    -1    -6  -352
## LRk     38  -379  -253  -301   -53  -304
## LAs     47  -237  -121  -120    26  -122     0
## Mem     55  -363  -259  -284   -70  -300   -15    -8
## StL    -74  -123   -33   -44  -314   -36  -293   -88  -2
## Spo   -273    18   -52   -57  -512   -51  -338 -1177  -2
## Tpa   -229 -1045  -512  -628   -24  -452  -131     2  -1
```

Not so good...

Comparisons with multiple q's

$$d^{(q)}_{Ata,Bos} = \begin{cases} 1068, & \text{original data} \\ 571.3, & \text{approximation by } q=1 \\ 1156.9, & \text{approximation by } q=2 \\ 1158.7, & \text{approximation by } q=3 \\ 1159.1, & \text{approximation by } q=4 \\ 1159.2, & \text{approximation by } q=5 \\ \dots \end{cases}$$

$$d^{(q)}_{Ata,Tpa} = \begin{cases} 2821, & \text{original data} \\ 2431.2, & \text{approximation by } q=1 \\ 2822.0, & \text{approximation by } q=2 \\ 2825.1, & \text{approximation by } q=3 \\ 2825.5, & \text{approximation by } q=4 \\ 2825.5, & \text{approximation by } q=5 \end{cases}$$

Fit MDS with max. dimension $q = N - 1$

Go for max. dimension: Euclidean distance of MDS with $q = 11$:

```r
Air11 = cmdscale(as.dist(Airline),11)
```

```
## Warning in cmdscale(as.dist(Airline), 11): only 8 of the
## are > 0
```

```r
round(dist(Air11))
```

```
##      Ata  Bos  Cin  Col  Dal  Ind  LRk  LAs  Mem  StL  S
## Bos 1159
## Cin  600  903
## Col  612  817  186
## Dal  852 1836  969 1055
## Ind  637  987  141  214  907
## LRk  576 1506  661  744  362  618
## LAs 2253 3057 2192 2261 1434 2090 1723
## Mem  471 1370  528  605  488  485  205 1846
## StL  655 1218  348  433  688  253  429 1862  351
```


Compared MDS q=N-1 with the original distances

```r
round(dist(Air11)) - as.dist(Airline)
```

```
##     Ata Bos Cin Col Dal Ind LRk LAs Mem StL Spo
## Bos  91
## Cin 139  36
## Col  63  48  79
## Dal  47  17  26   5
## Ind 129  46  33  42  25
## LRk  71  12  43  19  37  56
## LAs  56   5   6  16  31  10  22
## Mem 105  15  26  19  24  49  68  15
## StL  97  40  10  24  43  19  76  14  57
## Spo   3  24  33  15  13  36  10  81  15  33
## Tpa  60  17  35  16  12  39   1  19  14  33   4
```

Remark: Conclusions on the nature of the original ‘distance’ data.

$q = 8$

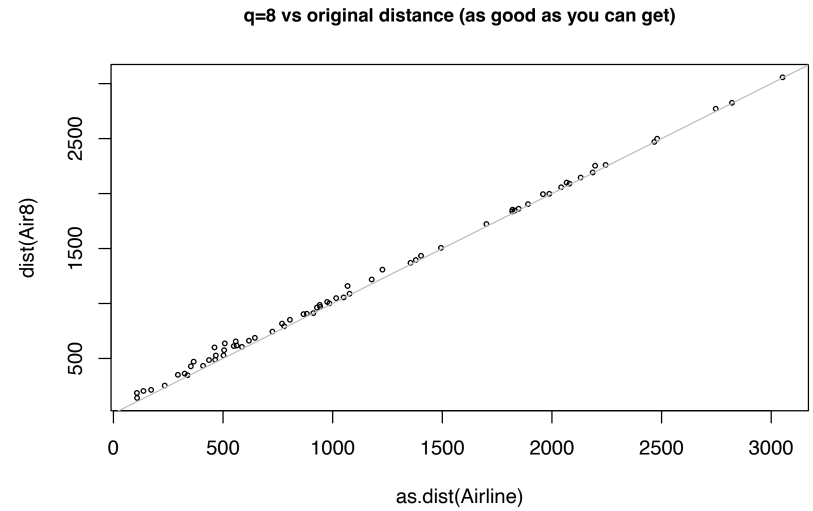
R warning indicates $q = 11$ configuration is the same as $q = 8$.

```
Air8 = cmdscale(as.dist(Airline),8)
round(dist(Air8)) - round(dist(Air11))
```

```
##     Ata Bos Cin Col Dal Ind LRk LAs Mem StL Spo
## Bos   0
## Cin   0   0
## Col   0   0   0
## Dal   0   0   0   0
## Ind   0   0   0   0   0
## LRk   0   0   0   0   0   0
## LAs   0   0   0   0   0   0   0
## Mem   0   0   0   0   0   0   0   0
## StL   0   0   0   0   0   0   0   0   0
## Spo   0   0   0   0   0   0   0   0   0   0
## Tpa   0   0   0   0   0   0   0   0   0   0   0
```

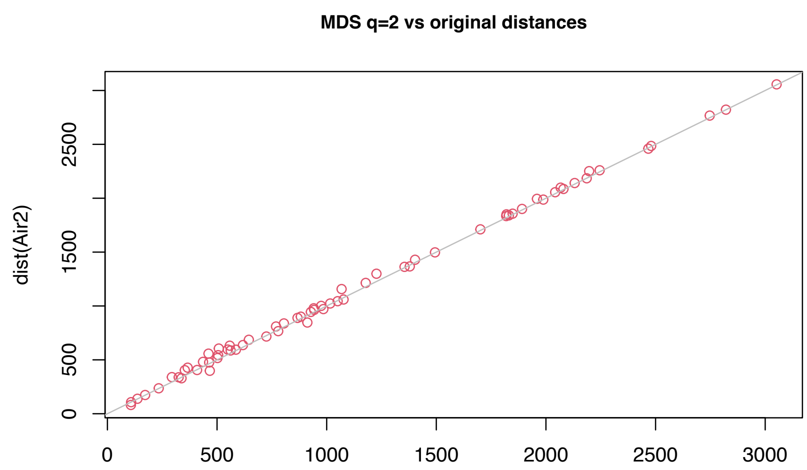

Plot: fitted vs orginal distance q=8

```
plot(as.dist(Airline),dist(Air8),cex=.5);abline(0,1,col="gr
title(main="q=8 vs original distance (as good as you can ge
```



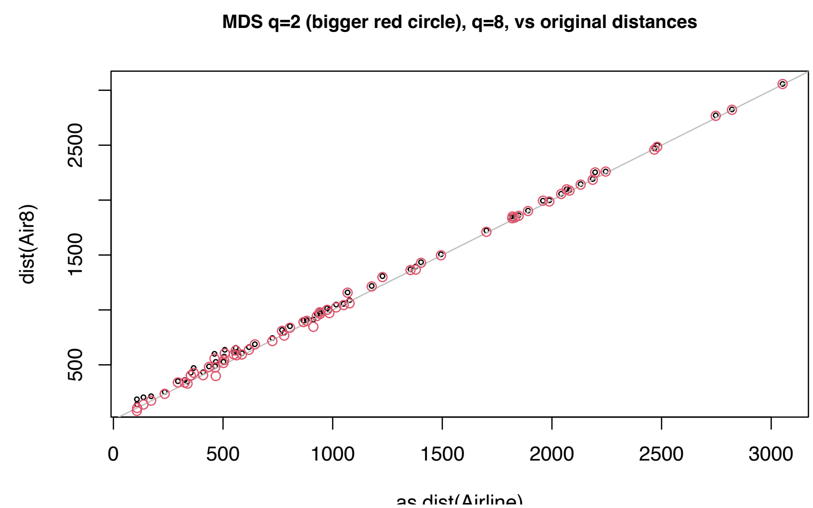
Plot: fitted vs orginal distance q=2

```
plot(as.dist(Airline),dist(Air2),col=2,xlab="");abline(0,1,
title(main="MDS q=2 vs original distances",cex.main=.9)
```



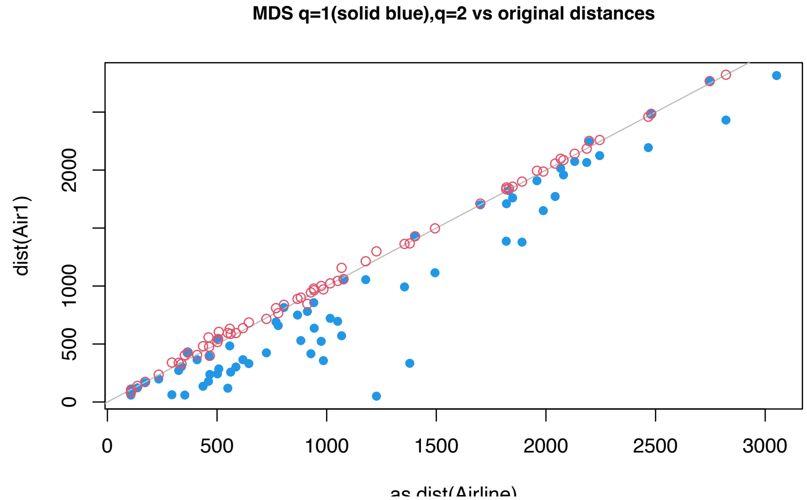
Plot: fitted vs orginal distance q=4

```
plot(as.dist(Airline),dist(Air8),cex=.5);abline(0,1,col="gr
points(as.dist(Airline),dist(Air2),col=2)  #q=2
title(main="MDS q=2 (bigger red circle), q=8, vs original d
```



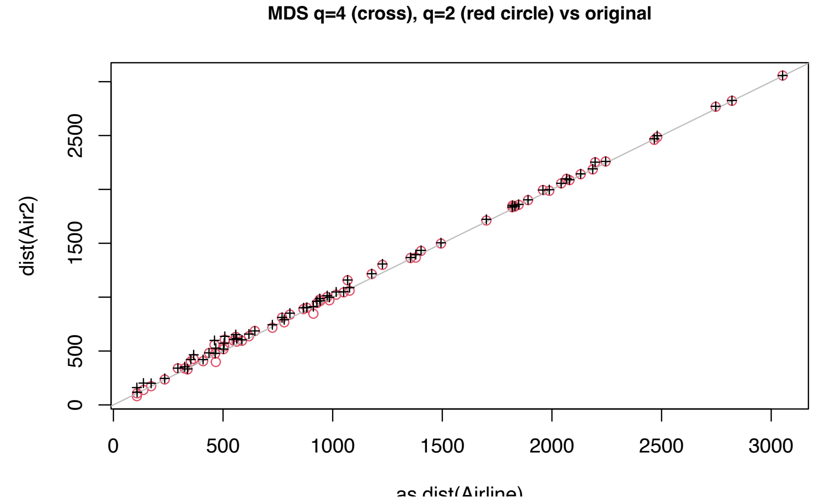
Plot: fitted vs orginal distance q=1

```
plot(as.dist(Airline),dist(Air1),pch=16, col=4);abline(0,1,
points(as.dist(Airline),dist(Air2),col=2)  #q=2
title(main="MDS q=1(solid blue),q=2 vs original distances",
```



Is MDS $q = 4$ better than $q = 2$?

```
plot(as.dist(Airline),dist(Air2),col=2); abline(0,1,col="gr
points(as.dist(Airline),dist(Air4),pch=3,cex=.7)  #q=4
title(main="MDS q=4 (cross), q=2 (red circle) vs original",
```



Stress as a meassure of goodness of fit of MDS model

$$Stress(q) = \left\{ \frac{\sum\sum_{i<k}(d_{ik} - \hat{d}_{ik})^2}{\sum\sum_{i<k} d_{ik}^2} \right\}^{1/2} \qquad \in [0,1]$$

- d_{ik} – the "ideal", desirable distance between objects i and k
- $\hat{d}_{ik} = \hat{d}_{ik}^{(q)}$ – the fitted distances by MDS model for dimension q

Computing Stress

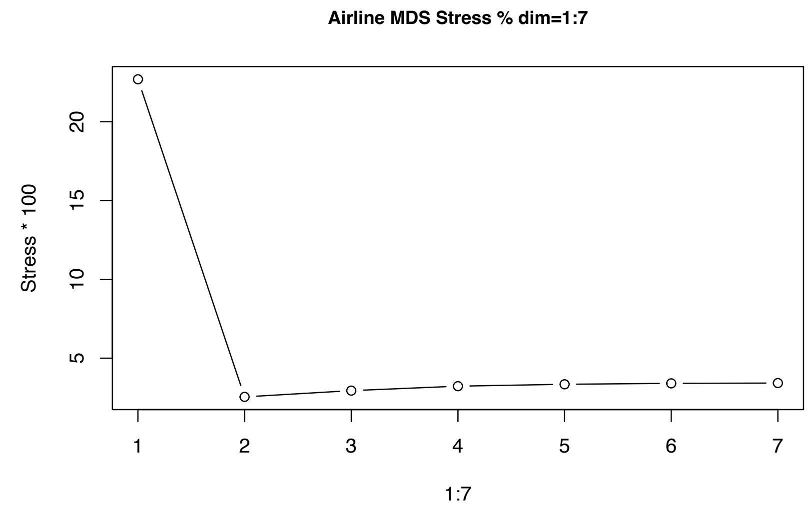
```
Stress=rep(0,7)
for (i in 1:7)
  {
      fit = cmdscale(as.dist(Airline),i)
      diffS2 = sum((as.dist(Airline)-dist(fit))^2)
      dist2 = sum((as.dist(Airline))^2)
      Stress[i]=sqrt(diffS2/dist2)
}
```

SStress — another meassure of stress

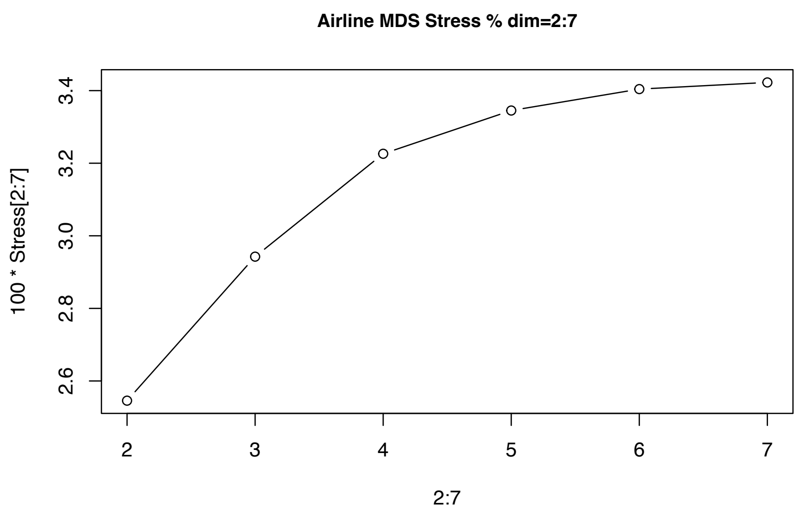
$$SStress(q) = \left\{ \frac{\sum\sum_{i<k}(d_{ik}^2 - \hat{d}_{ik}^2)^2}{\sum\sum_{i<k} d_{ik}^4} \right\}^{1/2} \quad \in [0,1]$$

```
SStress=rep(0,7)
for (i in 1:7)
  {
      fit = cmdscale(as.dist(Airline),i)
      diffS4 = sum(((as.dist(Airline))^2-(dist(fit))^2)^2)
      dist4 = sum((as.dist(Airline))^4)
      SStress[i]=sqrt(diffS4/dist4)
}
```

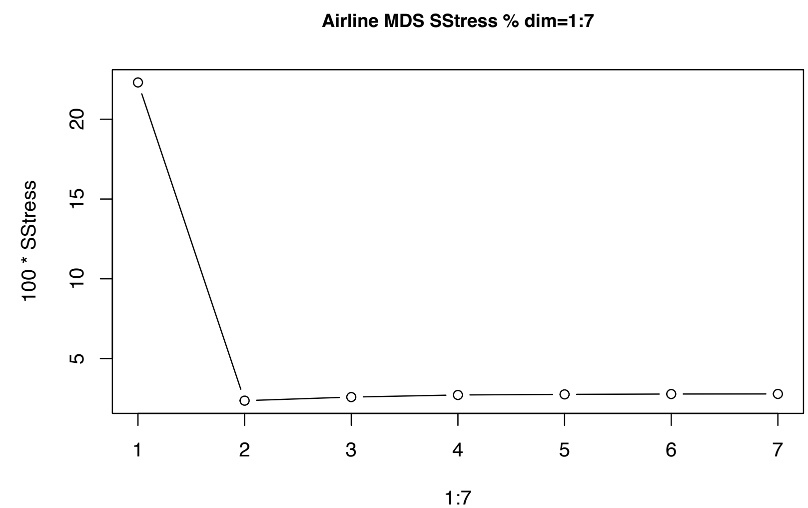

Stress plot



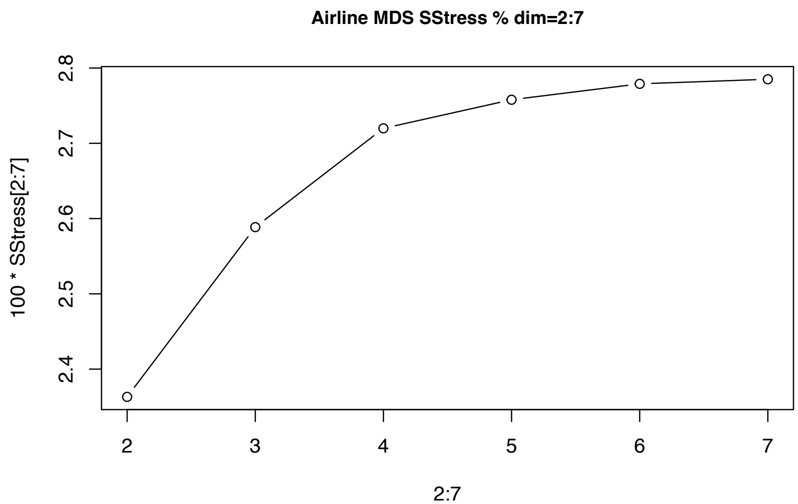
Stress: higher q may not be better



SStress plot

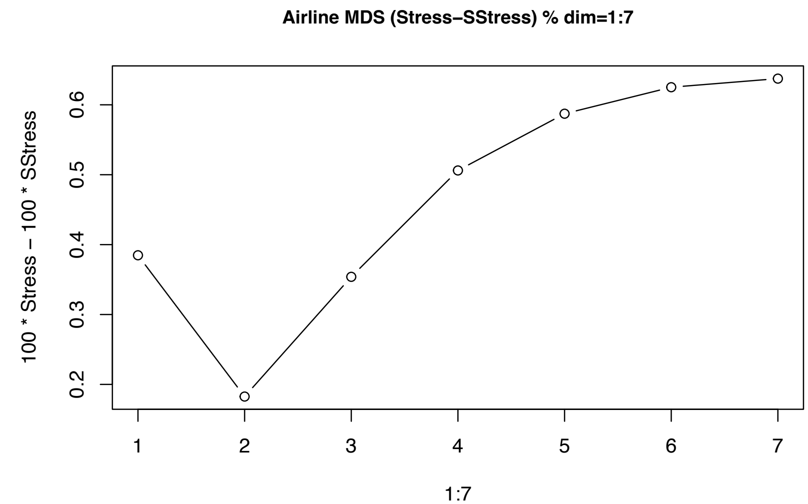


SStress plot details



SStress vs Stress

(SStress is smaller than Stress forn Example 2)



Code for Stress plots

Stress plot

```
plot(1:7,Stress*100,type="b")
title("Airline MDS Stress % dim=1:7",cex.main=.9)
```

Stress: higher q may not be better

```
plot(2:7,100*Stress[2:7],type="b")
title("Airline MDS Stress % dim=2:7",cex.main=.9)
```


Code for SStress plots

SStress plot

```
plot(1:7,100*SStress,type="b")
title("Airline MDS SStress % dim=1:7",cex.main=.9)
```

SStress plot details

```
plot(2:7,100*SStress[2:7],type="b")
title("Airline MDS SStress % dim=2:7",cex.main=.9)
```

SStress vs Stress

```
plot(1:7,100*Stress-100*SStress,type="b");
abline(h=0,lty=2,col="gray")
title("Airline MDS (Stress-SStress) % dim=1:7",
      cex.main=.9)
```