Multidimensional Scaling

Jan Graffelman[1]

[1]Department of Statistics and Operations Research
Universitat Politècnica de Catalunya
Barcelona, Spain

jan.graffelman@upc.edu

March 1, 2020

Contents

Multidimensional scaling

Objective

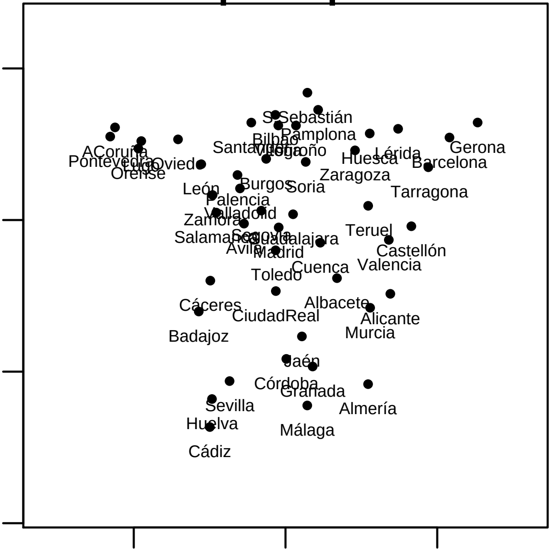
On the basis of information regarding the distances (or similarities) of n objects, construct a configuration of n points in a low-dimensional space (a map).

Example data set

	Albacete	Alicante	Almería	Avila	Badajoz	Barcelona	Bilbao	Burgos	...
Albacete	0	171	369	366	525	540	646	488	...
Alicante	171	0	294	537	696	515	817	659	...
Almería	369	294	0	663	604	809	958	800	...
Avila	366	537	663	0	318	717	401	243	...
Badajoz	525	696	604	318	0	1022	694	536	...
Barcelona	540	515	809	717	1022	0	620	583	...
Bilbao	646	817	958	401	694	620	0	158	...
Burgos	488	659	800	243	536	583	158	0	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	

Download SpainDist.dat

Map of Spain


SSebastián
Pamplona
Bilbao
Logroño
Huesca
Lérida
Gerona
Barcelona
ACoruña
Oviedo
Orense
Zaragoza
León
Burgos
Soria
Tarragona
Palencia
Zamora
Teruel
Guadalajara
Castellón
Madrid
Valencia
Cuenca
Toledo
Cáceres
Albacete
CiudadReal
Alicante
Badajoz
Murcia
Jaén
Córdoba
Granada
Almería
Sevilla
Huelva
Málaga
Cádiz

Some basic terminology

Terminology

- proximity
- similarity (s_{rs})
- dissimilarity or distance (d_{rs})

A similarity measure satisfies:

- $s(A, B) = s(B, A)$
- $s(A, B) > 0$
- $s(A, B)$ increases as the similarity between A and B increases

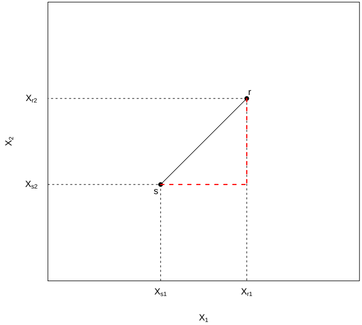
A distance measure, $\delta(A, B)$ satisfies:

- $\delta(A, B) = \delta(B, A)$
- $\delta(A, B) \geq 0$
- $\delta(A, A) = 0$

The distance function $\delta(A, B)$ called a metric if also

- $\delta(A, B) = 0$ iff $A = B$
- the triangle inequality holds: $\delta(A, B) \leq \delta(A, C) + \delta(C, B)$.

Euclidean Distance



$$\begin{aligned}\delta_{rs}^2 &= (x_{r1} - x_{s1})^2 + (x_{r2} - x_{s2})^2 \\ &= (\mathbf{x}_r - \mathbf{x}_s)'(\mathbf{x}_r - \mathbf{x}_s)\end{aligned}$$

Generalizes to p variables.

Some dissimilarity measures (quantitative data)

- Euclidean distance:

$$\delta_{rs} = \sqrt{(\mathbf{x}_r - \mathbf{x}_s)'(\mathbf{x}_r - \mathbf{x}_s)} = \left\{ \sum_{i=1}^{p} (x_{ri} - x_{si})^2 \right\}^{\frac{1}{2}}$$

- Mahalanobis distance:

$$\delta_{rs} = \left\{ (\mathbf{x}_r - \mathbf{x}_s)' \mathbf{S}^{-1} (\mathbf{x}_r - \mathbf{x}_s) \right\}^{\frac{1}{2}}$$

- Minkowski distance

$$\delta_{rs} = \left\{ \sum_{i=1}^{p} |x_{ri} - x_{si}|^{\lambda} \right\}^{\frac{1}{\lambda}}$$

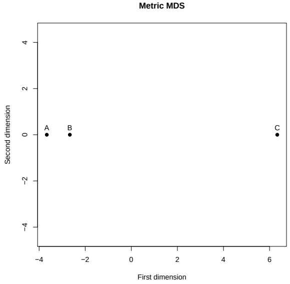
Metric versus Non-metric MDS

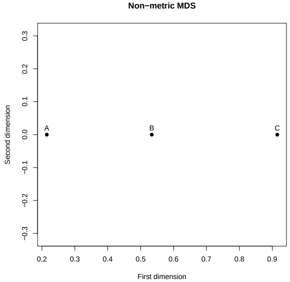
- In metric MDS, the configuration of points is directly obtained from the distances.
- In non-metric MDS, only the rank order of the distances is important.

- $d_{rs} \approx \delta_{rs}$: Classical scaling.
- $d_{rs} \approx f(\delta_{rs})$ with $f(\delta_{rs}) = \alpha + \beta\delta_{rs}$: Metric scaling.
- $d_{rs} \approx f(\delta_{rs})$ with $f(\delta_{rs})$ arbitrary, monotone: Non-metric scaling.

Metric versus Non-metric MDS

	A	B	C
A	0	1	10
B	1	0	9
C	10	9	0



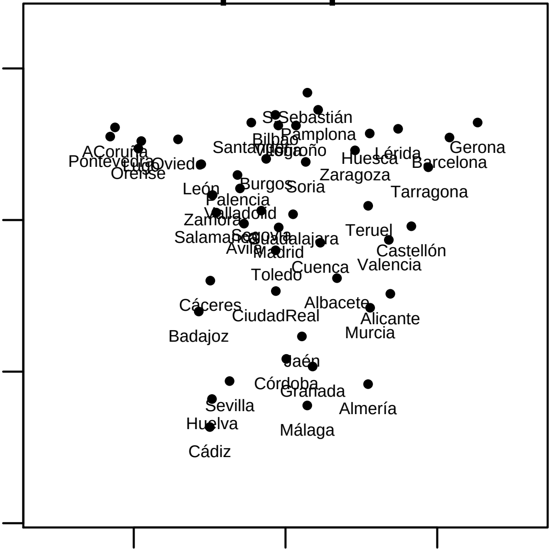


Metric MDS

- Also known as: classical scaling, principal coordinate analysis (PCO).
- Given n objects with dissimiliarities (δ_{rs}) find a set of points in Euclidean space such that $d_{rs} \approx \delta_{rs}$.
- Classical application: given a distance matrix (in km or in travel time) between cities, construct a map of the cities.

	Albacete	Alicante	Almería	Avila	Badajoz	Barcelona	Bilbao	Burgos	⋯
Albacete	0	171	369	366	525	540	646	488	⋯
Alicante	171	0	294	537	696	515	817	659	⋯
Almería	369	294	0	663	604	809	958	800	⋯
Avila	366	537	663	0	318	717	401	243	⋯
Badajoz	525	696	604	318	0	1022	694	536	⋯
Barcelona	540	515	809	717	1022	0	620	583	⋯
Bilbao	646	817	958	401	694	620	0	158	⋯
Burgos	488	659	800	243	536	583	158	0	⋯
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	

Map of Spain


S.Sebastián
Pamplona
Bilbao
Santander
Vitoria
Logroño
Huesca
Lérida
Gerona
Barcelona
ACoruña
Pontevedra
Lugo
Oviedo
Orense
León
Burgos
Soria
Zaragoza
Tarragona
Palencia
Valladolid
Zamora
Salamanca
Segovia
Guadalajara
Teruel
Castellón
Avila
Madrid
Valencia
Cuenca
Toledo
Albacete
Cáceres
CiudadReal
Alicante
Badajoz
Murcia
Jaén
Córdoba
Granada
Almería
Sevilla
Huelva
Málaga
Cádiz

Theory (1)

Let $\mathbf{X}$ be the matrix of coordinates with the solution.
$\mathbf{x}_r, \mathbf{x}_s$ two rows of $\mathbf{X}$.

$$\delta_{rs}^2 = (\mathbf{x}_r - \mathbf{x}_s)'(\mathbf{x}_r - \mathbf{x}_s)$$

Let $\mathbf{B}$ be the inner product matrix with

$$b_{rs} = {\mathbf{x}_r}'\mathbf{x}_s$$

Assume the solution to be centered at the origin:

$$\sum_{r=1}^{n} x_{ri} = 0$$

Theory (2)

$$d_{rs}^2 = \mathbf{x}_r{}'\mathbf{x}_r + \mathbf{x}_s{}'\mathbf{x}_s - 2\mathbf{x}_r{}'\mathbf{x}_s$$

$$b_{rs} = \mathbf{x}_r{}'\mathbf{x}_s = -\frac{1}{2}\left(d_{rs}^2 - \mathbf{x}_r{}'\mathbf{x}_r - \mathbf{x}_s{}'\mathbf{x}_s\right)$$

$$b_{rs} = -\frac{1}{2}\left(d_{rs}^2 - \frac{1}{n}\sum_{s=1}^{n} d_{rs}^2 - \frac{1}{n}\sum_{r=1}^{n} d_{rs}^2 + \frac{1}{n^2}\sum_{r=1}^{n}\sum_{s=1}^{n} d_{rs}^2\right).$$

We define $a_{rs} = -\frac{1}{2}d_{rs}^2$ so that $b_{rs} = a_{rs} - a_{r.} - a_{.s} + a_{..}$
and build matrix $\mathbf{A}$

$$\mathbf{B} = \mathbf{HAH} \quad \mathbf{H} = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}',$$

and

$$\mathbf{B} = \mathbf{X}\mathbf{X}'$$

We wish to approximate $\mathbf{B}$ in a low dimensional space.
We approximate the distance matrix indirectly, via de matrix of scalar products.

Theory (4) Spectral Decomposition

Let $\mathbf{B}$ be any $n \times n$ symmetric matrix we want to approximate

$$\mathbf{B} = \mathbf{V}\mathbf{D}_\lambda \mathbf{V}' = \sum_{i=1}^{n} \lambda_i \mathbf{v}_i \mathbf{v}_i'$$

with $\mathbf{D}_\lambda = \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ and $\mathbf{V} = [\mathbf{v}_i, \ldots, \mathbf{v}_n]$

$$\tilde{\mathbf{B}} = \mathbf{V}_{(,1:k)} \mathbf{D}_{\lambda(1:k,1:k)} \left(\mathbf{V}_{(,1:k)}\right)'$$

gives the rank k least squares approximation to $\mathbf{B}$

Theory (5) Solution

$$\mathbf{B} = \mathbf{X}\mathbf{X}' = \mathbf{V}\mathbf{D}_\lambda\mathbf{V}'$$

The coordinates of the solution are obtained as:

$$\mathbf{X} = \mathbf{V}\mathbf{D}_\lambda^{\frac{1}{2}}$$

Notes:

- There will always be at least one eigenvalue equal to zero.
- There is nesting of the solution.

Algorithm for Classical Scaling

- Compute a distance or dissimilarity matrix.
- Compute $[a_{rs}] = -\frac{1}{2}\delta_{rs}^2$
- Double center $\mathbf{A}$ to obtain $\mathbf{B} = \mathbf{HAH}$
- Compute eigenvalues and eigen vectors of $\mathbf{B}$
- Compute the solution as $\mathbf{X} = \mathbf{V}\mathbf{D}_{\lambda}^{\frac{1}{2}}$

Goodness of Fit

How well do we manage to approximate the distance matrix?

$$\frac{\sum_{i=1}^{P} \lambda_i}{\sum_{i=1}^{n-1} \lambda_i}$$

If $\mathbf{B}$ is not positive semi-definite:

$$\frac{\sum_{i=1}^{P} \lambda_i}{\sum_{i=1}^{n-1} |\lambda_i|} \quad \text{or} \quad \frac{\sum_{i=1}^{P} \lambda}{\sum_{\lambda_i>0} \lambda_i}$$

Euclidean Distance matrix

- Definition

  A distance matrix $\mathbf{D}$ is called Euclidean if there exists a configuration of points in Euclidean space whose interpoint distances are given by $\mathbf{D}$. That is, for some p there exists points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ such that $d_{rs}^2 = (\mathbf{x}_r - \mathbf{x}_s)'(\mathbf{x}_r - \mathbf{x}_s)$.

- Theorem

  A distance matrix $\mathbf{D}$ is Euclidean if and only if $\mathbf{B}$ ($= \mathbf{HAH}$, as previously defined) is positive semi definite.

Similarity data

- Sometimes data are given in the form of similarities (c_{rs}).
- A similarity matrix $\mathbf{C}$ has $c_{rs} = c_{sr}$ and $c_{rs} \leq c_{rr}$.
- Similarities can be transformed into distances with the transformation $\delta_{rs} = \sqrt{c_{rr} - 2c_{rs} + c_{ss}}$
- If $\mathbf{C}$ is psd, then the obtained distance matrix will be Euclidean.

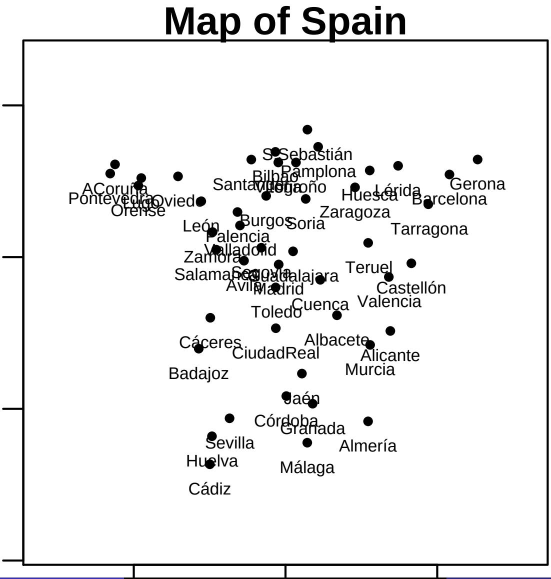
R code for classical scaling

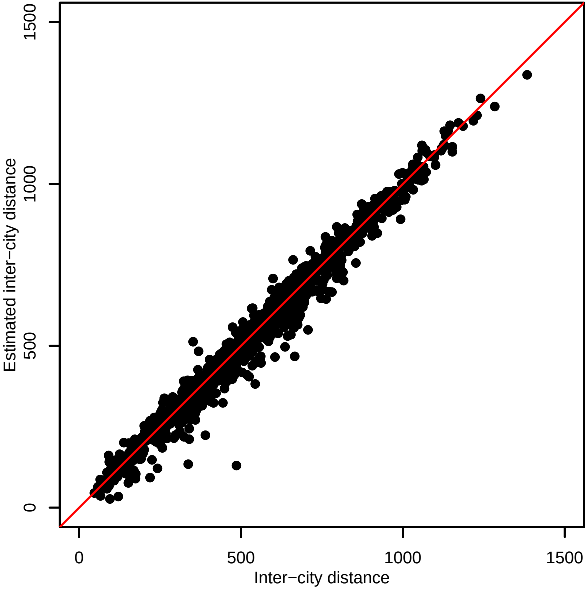
```
Spain <- as.matrix(read.table("http://www-eio.upc.es/~jan/data/SpainDist.dat",
         header=TRUE))
rownames(Spain) <- colnames(Spain)
n <- nrow(Spain)

mds.out <- cmdscale(Spain,k=n-1,eig=TRUE)

X <- mds.out$points[,1:2]
plot(X[,2],X[,1],type="n", xlab="", ylab="", main="Map of Spain",asp=1,
     xlim=c(-800,800),ylim=c(-800,500))
points(X[,2],X[,1],pch=19,cex=0.5)
text(X[,2],X[,1],rownames(Spain), cex=0.5,pos=1)
```

R code for classical scaling

```
> ev <- mds.out$eig
> gof <- mds.out$GOF
> print(round(ev,digits=2))
 [1] 4419357.73 3710242.86  523390.06  222914.52  215904.45  143955.45
 [7]  128021.63  103602.38   92361.07   77669.80   67866.94   55724.33
[13]   51347.16   38327.38   32347.58   29609.07   18785.64   14974.46
[19]    9473.34    9317.99    6911.58    4219.73    1459.24     105.43
[25]       0.00    -854.58   -3724.49   -4557.54   -5306.92   -8958.67
[31]  -11879.05  -15217.83  -16867.79  -24417.22  -34120.67  -43608.19
[37]  -50334.85  -63916.60  -77134.54  -80754.15  -91612.38  -97422.06
[43] -120383.81 -125973.49 -179445.66 -253056.31 -340735.97
> print(round(gof,digits=4))
[1] 0.8581 1.0000

> (ev[1]+ev[2])/sum(abs(ev))
[1] 0.6991297

> (ev[1]+ev[2])/sum(ev[ev>0])
[1] 0.8147615

> mds.out <- cmdscale(Spain,k=2,eig=TRUE)
> mds.out$GOF
[1] 0.6991297 0.8147615
```

PCO map of Spain



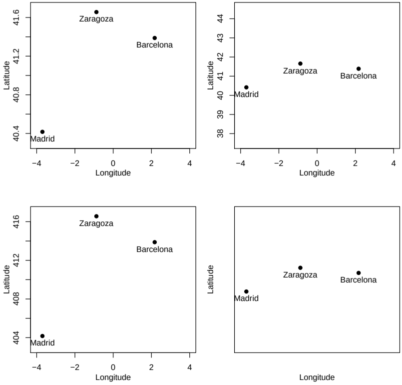

Estimated inter−city distance
0
500
1000
1500
0
500
1000
1500
Inter−city distance

About plotting maps

	Latitude (°)	Longitude (°)
Zaragoza	41.66	-0.88
Barcelona	41.39	2.17
Madrid	40.42	-3.70

	Zaragoza	Barcelona	Madrid
Zaragoza	0.00	158.67	170.25
Barcelona	158.67	0.00	313.74
Madrid	170.25	313.74	0.00

Distances in km.



Non-metric MDS: objective function

- $\text{STRESS} = \sqrt{\frac{\sum_{r \neq s}^{n} (f(\delta_{rs}) - d_{rs})^2}{\sum_{r \neq s} d_{rs}^2}}$
- $stress(\Delta, \hat{\mathbf{X}}) = \min_{all \mathbf{X}} \quad stress(\Delta, \mathbf{X})$
- We minimize the objective function numerically, starting from an initial configuration.
- Goodness-of-fit:

Stress	fit
20%	poor fit
10%	fair fit
5%	good fit
0%	perfect fit

Procedure for Non-metric MDS

- Choose a distance measure (e.g. $\delta_{rs} = \left\{\sum_{i=1}^{p} |x_{ri} - x_{si}|^{\lambda}\right\}^{\frac{1}{\lambda}}$)
- Choose a monotone transformation f
- Choose an algorithm to minimize Stress.

Global versus local minima

- Use different initial configurations
- Compare stress over 1,2,3,... dimensional solutions

Diagnostics

- Scatter plot of δ_{rs} versus d_{rs}
- Plot stress versus number of dimensions
- Degeneracy (many points with the same d_{rs})
- Compute residuals ($d_{rs} - f(\delta_{rs})$)

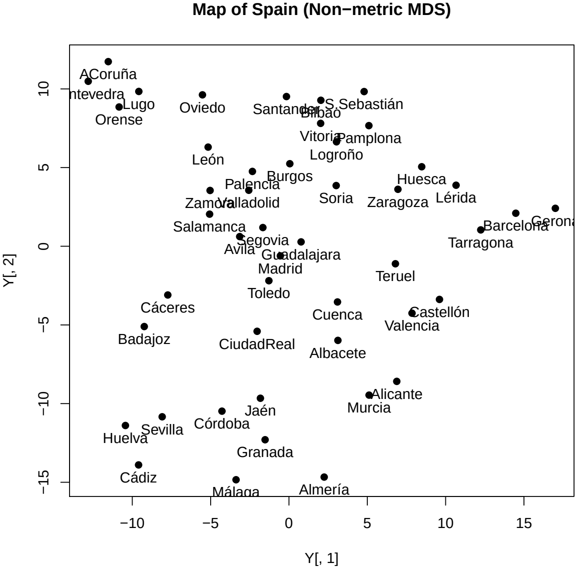
R code for non-metric MDS

```
> init <- scale(matrix(runif(n*2),ncol=2),scale=FALSE)
> nmmds.out <- isoMDS(Spain,y=init,k=2)
initial  value 41.659041
iter   5 value 40.219780
iter  10 value 37.286307
iter  15 value 30.177635
iter  20 value 22.661686
iter  25 value 14.483317
iter  30 value 10.703962
iter  35 value 7.756514
iter  40 value 6.116380
iter  45 value 5.360785
iter  50 value 5.145884
final  value 5.145884
stopped after 50 iterations
```

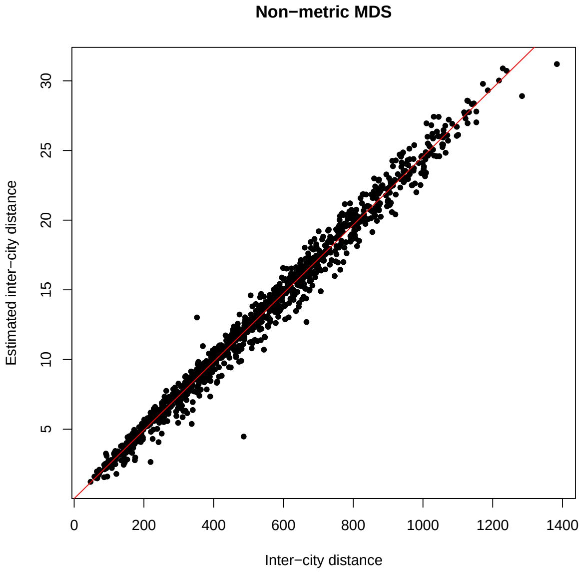
R code for non-metric MDS

```
> nmmds.out <- isoMDS(Spain,y=init,k=2,maxit=100)
initial  value 41.659041
iter   5 value 40.219780
iter  10 value 37.286307
iter  15 value 30.177635
iter  20 value 22.661686
iter  25 value 14.483317
iter  30 value 10.703962
iter  35 value 7.756514
iter  40 value 6.116380
iter  45 value 5.360785
iter  50 value 5.145884
iter  55 value 5.088756
final  value 5.057439
converged
> Y <- nmmds.out$points
> nmmds2.out <- isoMDS(Spain,y=X2,k=2) # PCO solution as initial configuration
initial  value 6.252429
final  value 6.252214
converged
> Y2 <- nmmds2.out$points
> plot(Y[,2],Y[,1],pch=19)
> text(Y[,2], Y[,1], rownames(Spain), cex=0.5,pos=1)
```

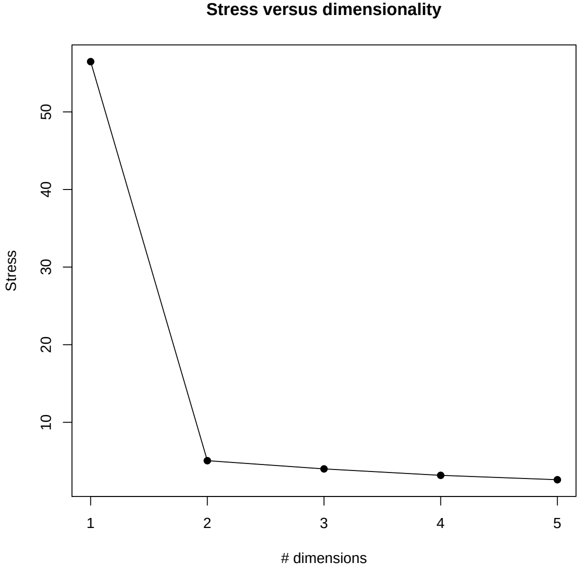
Non-metric MDS map of Spain



Diagnostics non-metric MDS



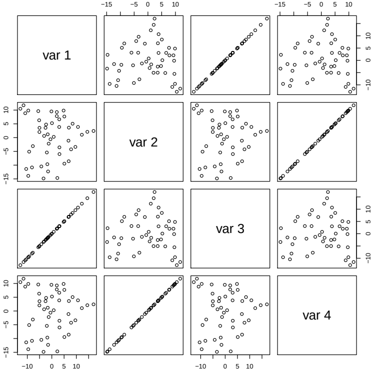
Diagnostics non-metric MDS



A property of non-metric MDS (1/2)

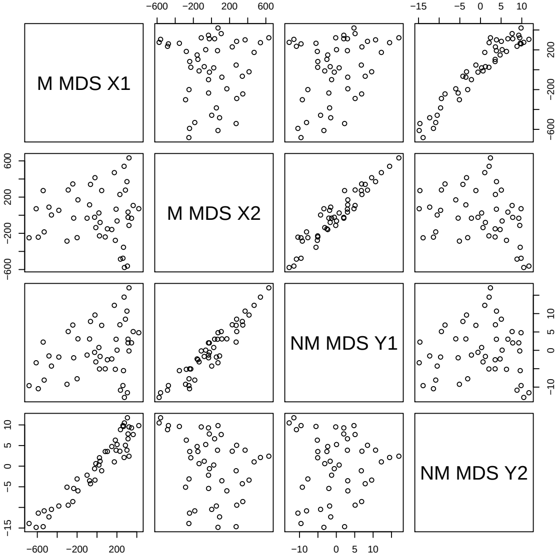
```
> n <- nrow(Spain)
> init <- cen(matrix(runif(n*2),ncol=2))
> Spain.out <- isoMDS(Spain,y=init,k=2,maxit=100)
initial  value 41.659041
iter   5 value 40.219780
iter  10 value 37.286307
.
iter  55 value 5.088756
final  value 5.057439
converged
> X.original <- Spain.out$points
> DistVec <- Spain[lower.tri(Spain)]
> DistVecRank <- rank(DistVec)
> SpainRank <- Spain
> SpainRank[lower.tri(SpainRank)] <- DistVecRank
> SpainRank[upper.tri(SpainRank)] <- 0
> diag(SpainRank) <- 0
> SpainRank <- SpainRank + t(SpainRank)
> SpainRank.out <- isoMDS(SpainRank,y=init,k=2,maxit=100)
initial  value 41.659041
iter   5 value 40.219780
iter  10 value 37.286307
.
iter  55 value 5.088756
final  value 5.057439
converged
> X.rank <- SpainRank.out$points
```

A property of non-metric MDS (2/2)




```
> pairs(cbind(X.original,X.rank))
> cor(cbind(X.original,X.rank))
           [,1]       [,2]       [,3]       [,4]
[1,] 1.00000000 0.02394018 1.00000000 0.02394018
[2,] 0.02394018 1.00000000 0.02394018 1.00000000
[3,] 1.00000000 0.02394018 1.00000000 0.02394018
[4,] 0.02394018 1.00000000 0.02394018 1.00000000
>
```

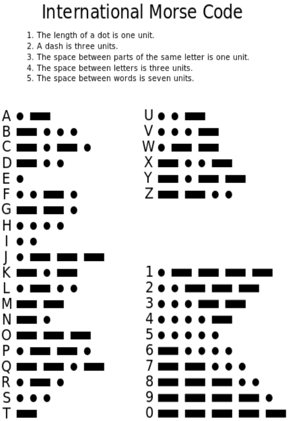
Relation metric MDS and non-metric MDS solutions



Correlation solutions MDS versus non-metric MDS

	M MDS X1	M MDS X2	NM MDS Y1	NM MDS Y2
M MDS X1	1.00	0.00	0.31	0.96
M MDS X2	0.00	1.00	0.95	-0.29
NM MDS Y1	0.31	0.95	1.00	0.02
NM MDS Y2	0.96	-0.29	0.02	1.00

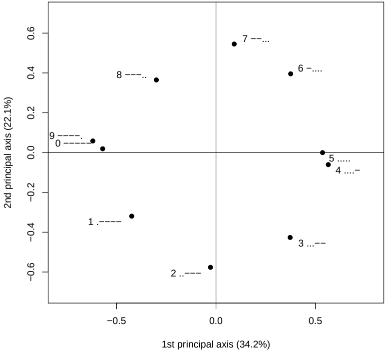
Example: Morse code data



	1	2	3	4	5	6	7	8	9	0
1	84	62	16	6	12	12	20	37	57	52
2	62	89	59	23	8	14	25	25	28	18
3	16	59	86	38	27	33	17	16	9	9
4	6	23	38	89	56	34	24	13	7	7
5	12	8	27	56	90	30	18	10	5	5
6	12	14	33	34	30	86	65	22	8	18
7	20	25	17	24	18	65	85	65	31	15
8	37	25	16	13	10	22	65	88	58	39
9	57	28	9	7	5	8	31	58	91	79
0	52	18	9	7	5	18	15	39	79	94

% of times a signal is declared identical

Morse data: metric MDS



	λ_i	Fraction	Acc.
1	1.874	0.342	0.342
2	1.210	0.221	0.563
3	0.954	0.174	0.738
4	0.554	0.101	0.839
5	0.466	0.085	0.924
6	0.315	0.058	0.982
7	0.096	0.018	0.999
8	0.045	0.008	1.007
9	0.000	0.000	1.007
10	-0.041	-0.007	1.000

MDS with genetic data

- There is a rich literature on how to measure genetic distance
- The allele sharing distance is an often used measure

$i \backslash j$	AA	AB	BB
AA	2	1	0
AB	1	2	1
BB	0	1	2

$i \backslash j$	AA	AB	BB
AA	0	1	2
AB	1	0	1
BB	2	1	0

- Let x_{ijk} be the number of shared alleles of individual i and j for variant k
- $d_{ijk} = 2 - x_{ijk}$
- Often scaled by multiplying by $\frac{1}{2}$
- Typically averaged over K genetic variants:

$$d_{ij} = \frac{1}{K} \sum_{k=1}^{K} d_{ijk}$$

- The so obtained $\mathbf{D} = [d_{ij}]$ is used as input for MDS.

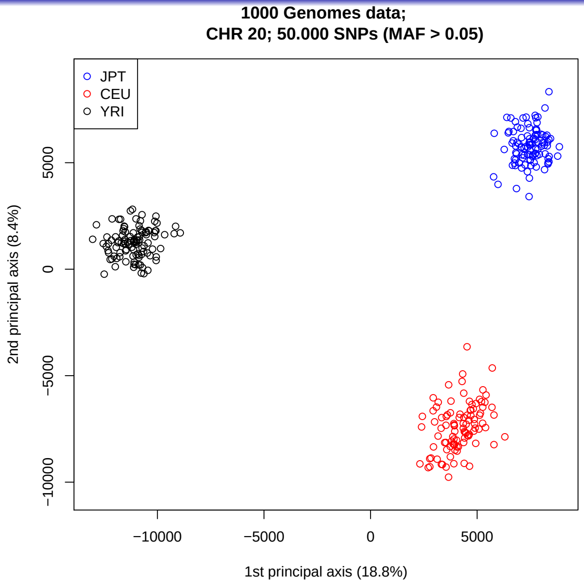
Genetic data of the 1000 Genomes project

	rs6078030	rs552482647	rs4814683	rs6076506	rs6139074	
1	0	0	0	0	0	...
2	2	0	2	0	2	...
3	0	0	1	0	0	...
4	0	0	0	0	0	...
5	0	0	0	0	0	...
⋮	⋮	⋮	⋮	⋮	⋮	⋱

	1	2	3	4	5	
1	0.00	0.43	0.44	0.47	0.43	...
2	0.43	0.00	0.44	0.46	0.48	...
3	0.44	0.44	0.00	0.46	0.44	...
4	0.47	0.46	0.46	0.00	0.45	...
5	0.43	0.48	0.44	0.45	0.00	...
⋮	⋮	⋮	⋮	⋮	⋮	⋱

We consider 310 individuals for 50,000 SNPs

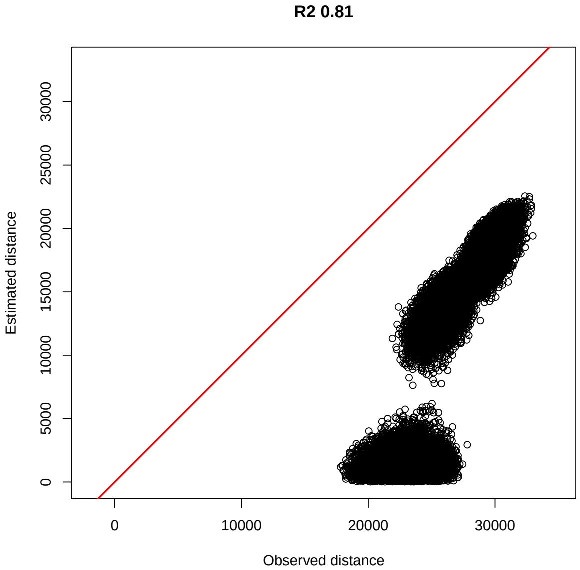
MDS with genetic data (CEU, JPT and YRI from the 1000 Genomes project)



Goodness of fit

Dim.	λ	%	% Cum.
1	8.3567	0.1816	0.1816
2	3.7353	0.0812	0.2628
3	0.6275	0.0136	0.2764
4	0.5274	0.0115	0.2879
5	0.4909	0.0107	0.2985
6	0.4675	0.0102	0.3087
7	0.4540	0.0099	0.3186
8	0.4493	0.0098	0.3283
9	0.4345	0.0094	0.3378
10	0.4211	0.0091	0.3469
.	.	.	.
.	.	.	.
.	.	.	.
260	0.0005	0.0000	0.9840
261	0.0001	0.0000	0.9840
262	0.0000	0.0000	0.9840
263	-0.0008	0.0000	0.9841
264	-0.0011	0.0000	0.9841
265	-0.0017	0.0000	0.9841
.	.	.	.
.	.	.	.
.	.	.	.
306	-0.0289	0.0006	0.9972
307	-0.0297	0.0006	0.9979
308	-0.0305	0.0007	0.9986
309	-0.0321	0.0007	0.9993
310	-0.0343	0.0007	1.0000

Goodness of fit



Bibliography

- Borg, I. & Groenen, P. (1997) Modern Multidimensional Scaling. Theory and Applications. Springer.
- Cox, T.F. & Cox, M.A. (2001) Multidimensional Scaling. Second edition. Chapman & Hall
- Cuadras, C. (2008) Nuevos métodos de Análisis Multivariante. Chapter 8. Download book here
- Kruskal, J.B. & Wish, M. (1978) Multidimensional Scaling. Sage university papers.
- Peña, D. (2002) Análisis de datos multivariantes. McGraw-Hill, Madrid.