Lecture 8: Multidimensional scaling

Advanced Applied Multivariate Analysis

STAT 2221, Fall 2013

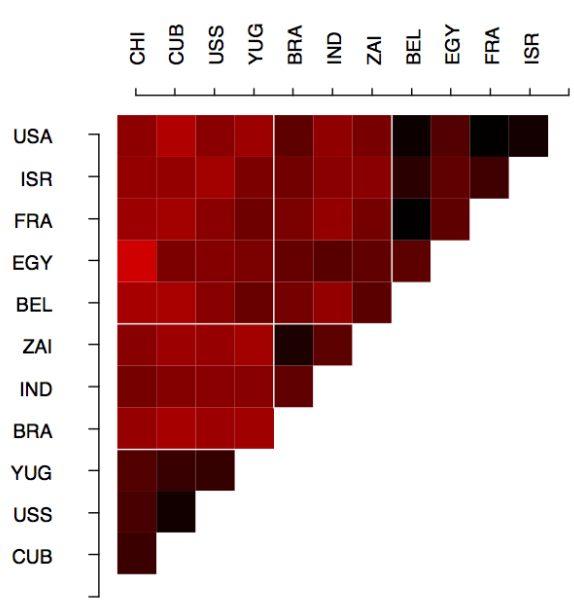
Sungkyu Jung
Department of Statistics
University of Pittsburgh

E-mail: sungkyu@pitt.edu
http://www.stat.pitt.edu/sungkyu/AAMA/

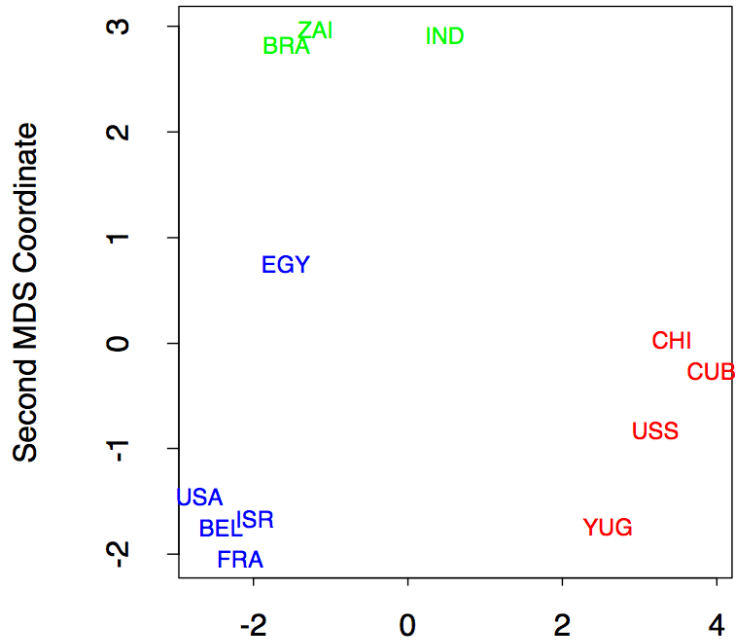
Multidimensional scaling

Goal of Multidimensional scaling (MDS): Given pairwise dissimilarities, reconstruct a map that preserves distances.

- From any dissimilarity (no need to be a metric)
- Reconstructed map has coordinates $\mathbf{x}_i = (x_{i1}, x_{i2})$ and the natural distance ($\|\mathbf{x}_i - \mathbf{x}_j\|_2$)



Reordered Dissimilarity Matrix



Multidimensional scaling

- MDS is a family of different algorithms, each designed to arrive at optimal low-dimensional configuration ($p = 2$ or 3)
- MDS methods include
  1. Classical MDS
  2. Metric MDS
  3. Non-metric MDS

Perception of Color in human vision

- To study the perceptions of color in human vision (Ekman, 1954, Izenman 13.2.1)
- 14 colors differing only in their hue (i.e., wavelengths from 434 μm to 674 μm)
- 31 people to rate on a five-point scale from 0 (no similarity at all) to 4 (identical) for each of $\binom{14}{2}$ pairs of colors.
- Average of 31 ratings for each pair (representing similarity) is then scaled and subtracted from 1 to represent dissimilarities

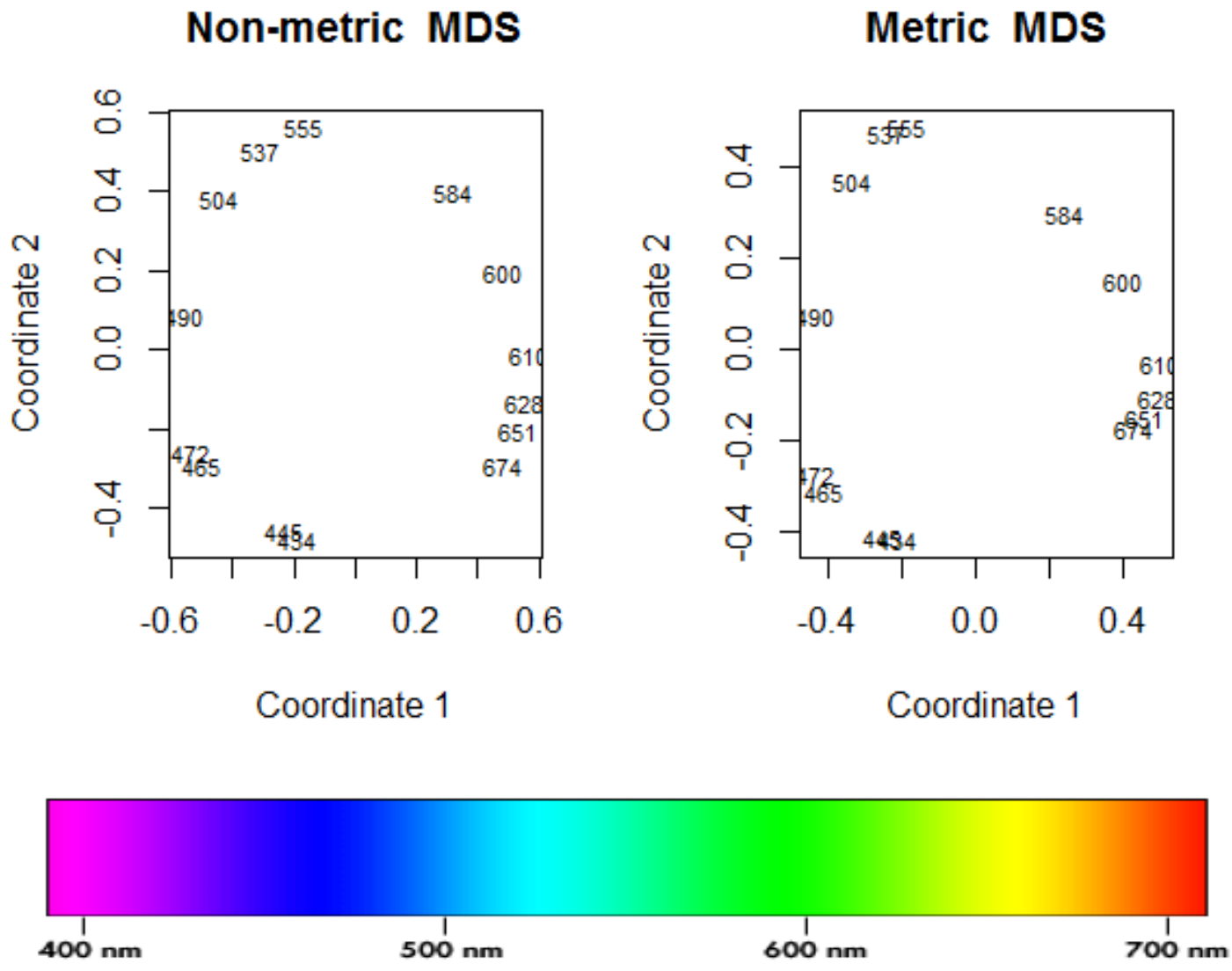
Perception of Color in human vision

The resulting 14×14 dissimilarity matrix is symmetric, and contains zeros in the diagonal. MDS seeks a 2D configuration to represent these colors.

	434	445	465	472	490	504	537	555	584	600	610	628	651
445	0.14												
465	0.58	0.50											
472	0.58	0.56	0.19										
490	0.82	0.78	0.53	0.46									
504	0.94	0.91	0.83	0.75	0.39								
537	0.93	0.93	0.90	0.90	0.69	0.38							
555	0.96	0.93	0.92	0.91	0.74	0.55	0.27						
584	0.98	0.98	0.98	0.98	0.93	0.86	0.78	0.67					
600	0.93	0.96	0.99	0.99	0.98	0.92	0.86	0.81	0.42				
610	0.91	0.93	0.98	1.00	0.98	0.98	0.95	0.96	0.63	0.26			
628	0.88	0.89	0.99	0.99	0.99	0.98	0.98	0.97	0.73	0.50	0.24		
651	0.87	0.87	0.95	0.98	0.98	0.98	0.98	0.98	0.80	0.59	0.38	0.15	
674	0.84	0.86	0.97	0.96	1.00	0.99	1.00	0.98	0.77	0.72	0.45	0.32	0.24

Perception of Color in human vision

MDS reproduces the well-known two-dimensional *color circle*.



Distance, dissimilarity and similarity

Distance, dissimilarity and similarity (or proximity) are defined for any pair of objects in any space. In mathematics, a distance function (that gives a distance between two objects) is also called *metric*, satisfying

1. $d(x, y) \geq 0$,
2. $d(x, y) = 0$ if and only if $x = y$,
3. $d(x, y) = d(y, x)$,
4. $d(x, z) \leq d(x, y) + d(y, z)$.

Given a set of dissimilarities, one can ask whether these values are distances and, moreover, whether they can even be interpreted as Euclidean distances

Euclidean and non-Euclidean distance

Given a dissimilarity (distance) matrix $D = (d_{ij})$, MDS seeks to find $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^p$ so that

$$d_{ij} \approx \|\mathbf{x}_i - \mathbf{x}_j\|_2 \text{ as close as possible.}$$

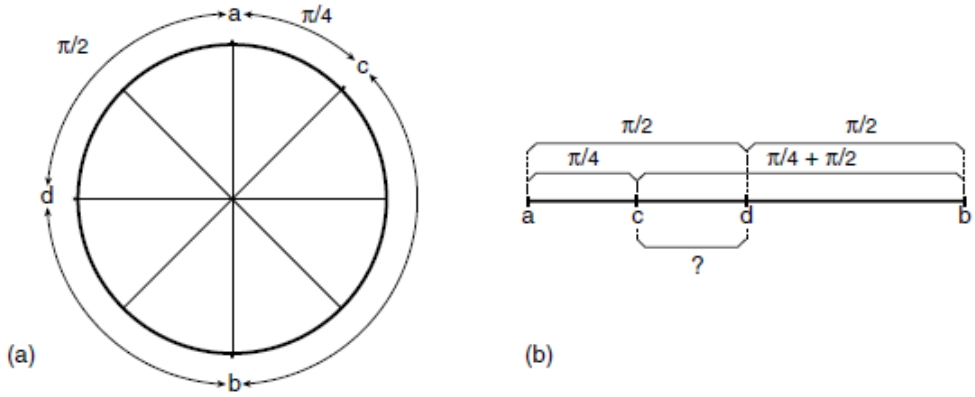
Oftentimes, for some large p, there exists a configuration $\mathbf{x}_1, \ldots, \mathbf{x}_n$ with exact distance match $d_{ij} \equiv \|\mathbf{x}_i - \mathbf{x}_j\|_2$. In such a case the distance d involved is called a Euclidean distance. There are, however, cases where the dissimilarity is distance, but there exists no configuration in any p with perfect match

$$d_{ij} \neq \|\mathbf{x}_i - \mathbf{x}_j\|_2, \text{ for some } i, j.$$

Such a distance is called non-Euclidean distance.

non-Euclidean distance

- Radian distance function on a circle is a metric.
- Cannot be embedded in $\mathbb{R}$. (Not for any $\mathbb{R}^p$, not shown here)



Point	a	b	c	d
a	0.0000	3.1416	0.7854	1.5708
b	3.1416	0.0000	2.3562	1.5708
c	0.7854	2.3562	0.0000	2.3562
d	1.5708	1.5708	2.3562	0.0000

- Nevertheless, MDS seeks to find an optimal configuration $\mathbf{x}_i$ that gives $d_{ij} \approx \|\mathbf{x}_i - \mathbf{x}_j\|_2$ as close as possible.

classical Multidimensional Scaling–theory

Suppose for now we have Euclidean distance matrix $D = (d_{ij})$.

The objective of classical Multidimensional Scaling (cMDS) is to find $X = [x_1, \ldots, x_n]$ so that $\|x_i - x_j\| = d_{ij}$. Such a solution is not unique, because if X is the solution, then $X^* = X + c$, $c \in \mathbb{R}^q$ also satisfies $\left\|x_i^* - x_j^*\right\| = \|(x_i + c) - (x_j + c)\| = \|x_i - x_j\| = d_{ij}$. Any location c can be used, but the assumption of centered configuration, i.e.,

$$\sum_{i=1}^{n} x_{ik} = 0, \quad \text{for all } k, \tag{1}$$

serves well for the purpose of dimension reduction.

classical Multidimensional Scaling–theory

In short, the cMDS finds the centered configuration $\mathbf{x}_1, \ldots, \mathbf{x}_n \in \mathbb{R}^q$ for some $q \geq n-1$ so that their pairwise distances are the same as those corresponding distances in D.

We may find the $n \times n$ Gram matrix $B = X'X$, rather than X. The Gram matrix is the inner product matrix since X is assumed to be centered. We have

$$d_{ij}^2 = b_{ii} + b_{jj} - 2b_{ij} \tag{2}$$

from $\|x_i - x_j\|^2 = x_i'x_i + x_j'x_j - 2x_i'x_j$.

classical Multidimensional Scaling–theory

The constraints (1) leads to

$$\sum_{i=1}^{n} b_{ij} = \sum_{i=1}^{n}\sum_{k=1}^{q} x_{ik}x_{jk} = \sum_{k=1}^{q} x_{jk}\sum_{i=1}^{n} x_{ik} = 0,$$

for $j = 1, \ldots, n$.
With a notation $T = \text{trace}(B) = \sum_{i=1}^{n} b_{ii}$, we have

$$\sum_{i=1}^{n} d_{ij}^2 = T + nb_{jj}, \ \sum_{j=1}^{n} d_{ij}^2 = T + nb_{ii}, \ \sum_{j=1}^{n}\sum_{i=1}^{n} d_{ij}^2 = 2nT. \quad (3)$$

classical Multidimensional Scaling–theory

Combining (2) and (3), the solution is unique:

$$b_{ij} = -1/2(d_{ij}^2 - d_{\cdot j}^2 - d_{i\cdot}^2 + d_{\cdot\cdot}^2)$$

or

$$B = -1/2 C D_2 C.$$

A solution X is then given by the eigen-decomposition of B. That is, for $B = V \Lambda V'$,

$$X = \Lambda^{1/2} V'. \tag{4}$$

classical Multidimensional Scaling–theory

The space which X lies is the eigenspace where the first coordinate contains the largest variation, and is identified with $\mathbb{R}^q$.

If we wish to reduce the dimension to $p \leq q$, then the first p rows of $X_{(p)}$ best preserves the distances d_{ij} among all other linear dimension reduction of X (to p). Then

$$X_{(p)} = \Lambda_p^{1/2} V_p',$$

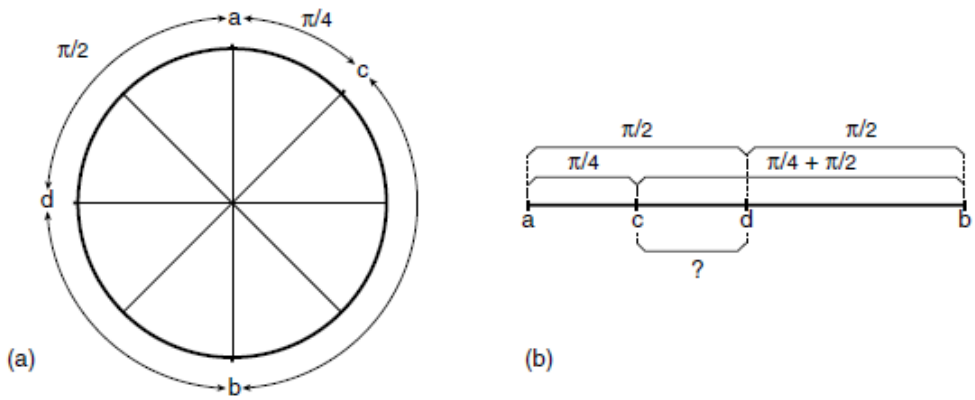
where Λ_p is the first $p \times p$ sub matrix of Λ, V_p is collected through the first p columns of V.

classical Multidimensional Scaling

- cMDS gives configurations $X_{(p)}$ in $\mathbb{R}^p$ for any dimension $p = 1, 2, \ldots, q$.
- Configuration is centered.
- The coordinates are given by the principal order of largest-to-smallest variances.
- Dimension reduction from $X = X_{(q)}$ to $X_{(p)}$ $(p < q)$ is same as PCA.
- Leads exact solution for Euclidean distances
- *Can be used for non-Euclidean distances, in fact, for any dissimilarities*.

cMDS examples

- Consider two worked examples:
  one with Euclidean geometry (tetrahedron–edge length 1),
  the other from the circular geometry, shown below.



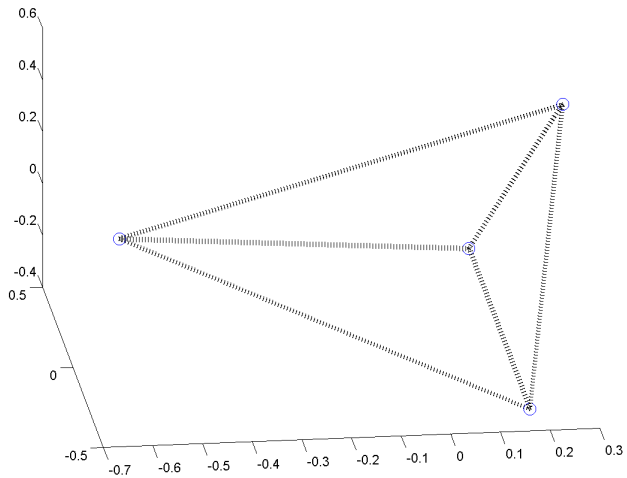
- And the airline distances example (Izenman 13.1.1)

cMDS examples: tetrahedron

Pairwise distance matrix for tetrahedron (with distance 1)

$$D = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix},$$

leading to the gram matrix $B_{(4\times 4)}$ with eigenvalues $(.5, .5, .5, 0)$. Using dimension $p = 3$, we have perfectly retrieved the tetrahedron.



cMDS examples: circular distances

Pairwise distance matrix

Point	a	b	c	d
a	0.0000	3.1416	0.7854	1.5708
b	3.1416	0.0000	2.3562	1.5708
c	0.7854	2.3562	0.0000	2.3562
d	1.5708	1.5708	2.3562	0.0000

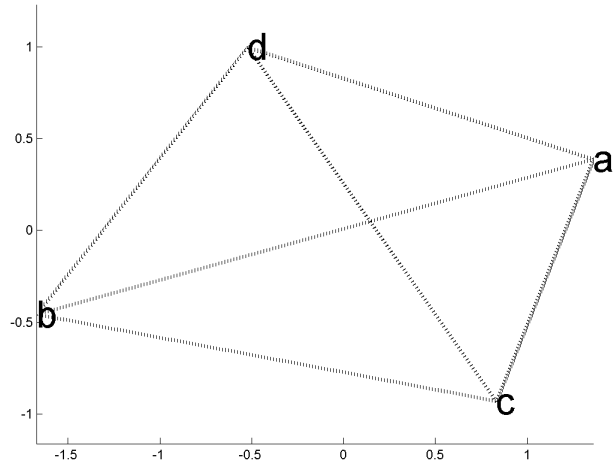
leading to the gram matrix $B_{(4\times 4)}$ with eigenvalues

$$\text{diag}(\Lambda) = (5.6117, -1.2039, -0.0000, 2.2234)$$

In retrieving the coordinate matrix X, we cannot take a squareroot of Λ since it gives complex numbers.
Remedy: *Keep only positive eigenvalues and corresponding coordinates*. In this case, take coordinates 1 and 4. This is the price we pay in approximating non-Euclidean geometry by Euclidean geometry.

cMDS examples: circular distances

Using dimension $p = 2$ (cannot use $p > 2$), configuration $X_{(2)}$ is



Compare the original distance matrix D and approximated distance matrix $\hat{D} = \|x_i - x_j\|_2$:

$$\begin{pmatrix} 0 & 3.1416 & 0.7854 & 1.5708 \\ 3.1416 & 0 & 2.3562 & 1.5708 \\ 0.7854 & 2.3562 & 0 & 2.3562 \\ 1.5708 & 1.5708 & 2.3562 & 0 \end{pmatrix}, \quad \hat{D} = \begin{pmatrix} 0 & 3.1489 & 1.4218 & 1.9784 \\ 3.1489 & 0 & 2.5482 & 1.8557 \\ 1.4218 & 2.5482 & 0 & 2.3563 \\ 1.9784 & 1.8557 & 2.3563 & 0 \end{pmatrix}$$

cMDS examples: Airline distances

TABLE 13.2. *Airline distances (km) between 18 cities. Source: Atlas of the World, Revised 6th Edition, National Geographic Society, 1995, p. 131.*

	Beijing	Cape Town	Hong Kong	Honolulu	London	Melbourne
Cape Town	12947					
Hong Kong	1972	11867				
Honolulu	8171	18562	8945			
London	8160	9635	9646	11653		
Melbourne	9093	10338	7392	8862	16902	
Mexico	12478	13703	14155	6098	8947	13557
Montreal	10490	12744	12462	7915	5240	16730
Moscow	5809	10101	7158	11342	2506	14418
New Delhi	3788	9284	3770	11930	6724	10192
New York	11012	12551	12984	7996	5586	16671
Paris	8236	9307	9650	11988	341	16793
Rio de Janeiro	17325	6075	17710	13343	9254	13227
Rome	8144	8417	9300	12936	1434	15987
San Francisco	9524	16487	11121	3857	8640	12644
Singapore	4465	9671	2575	10824	10860	6050
Stockholm	6725	10334	8243	11059	1436	15593
Tokyo	2104	14737	2893	6208	9585	8159

	Mexico	Montreal	Moscow	New Delhi	New York	Paris
Montreal	3728					
Moscow	10740	7077				
New Delhi	14679	11286	4349			
New York	3362	533	7530	11779		
Paris	9213	5522	2492	6601	5851	

cMDS examples: Airline distances

TABLE 13.6. *Eigenvalues of* $\mathbf{B}$ *and the eigenvectors corresponding to the first three largest eigenvalues (in red) for the airline distances example.*

	Eigenvalues	Eigenvectors		
1	471582511	0.245	–0.072	0.183
2	316824787	0.003	0.502	-0.347
3	253943687	0.323	–0.017	0.103
4	–98466163	0.044	–0.487	-0.080
5	–74912121	–0.145	0.144	0.205
6	–47505097	0.366	–0.128	–0.569
7	31736348	–0.281	–0.275	–0.174
8	–7508328	–0.272	–0.115	0.094
9	4338497	–0.010	0.134	0.202
10	1747583	0.209	0.195	0.110
11	–1498641	–0.292	–0.117	0.061
12	145113	–0.141	0.163	0.196
13	–102966	–0.364	0.172	–0.473
14	60477	–0.104	0.220	0.163
15	–6334	–0.140	–0.356	–0.009
16	–1362	0.375	0.139	–0.054
17	100	–0.074	0.112	0.215
18	0	0.260	–0.214	0.173

- Airline distance is non-Euclidean
- Take the first 3 largest eigenvalues (inspection of scree plot)

cMDS examples: Airline distances

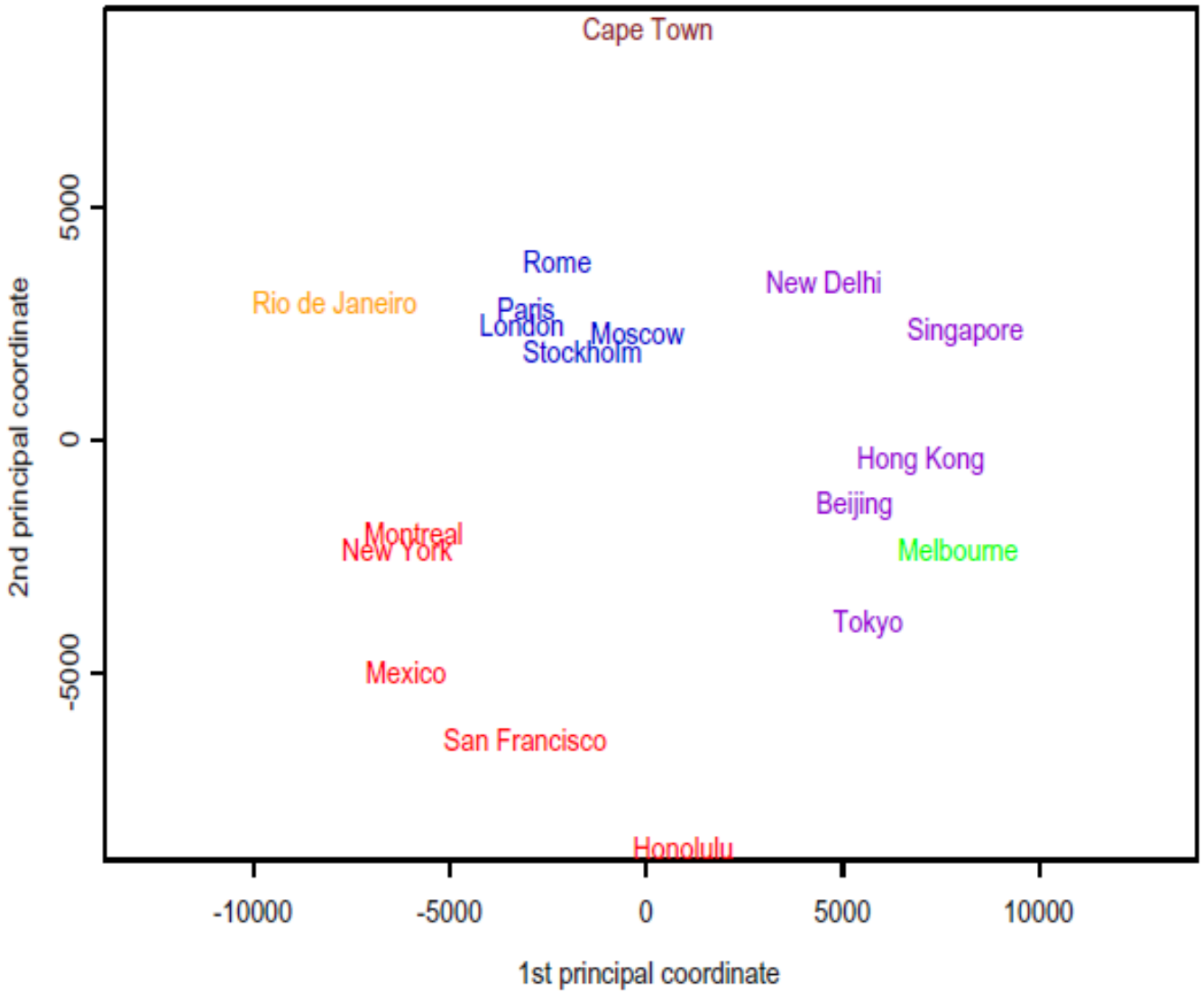


FIGURE 13.1. *Two-dimensional map of 18 world cities using the classical scaling algorithm on airline distances between those cities. The colors*

cMDS examples: Airline distances

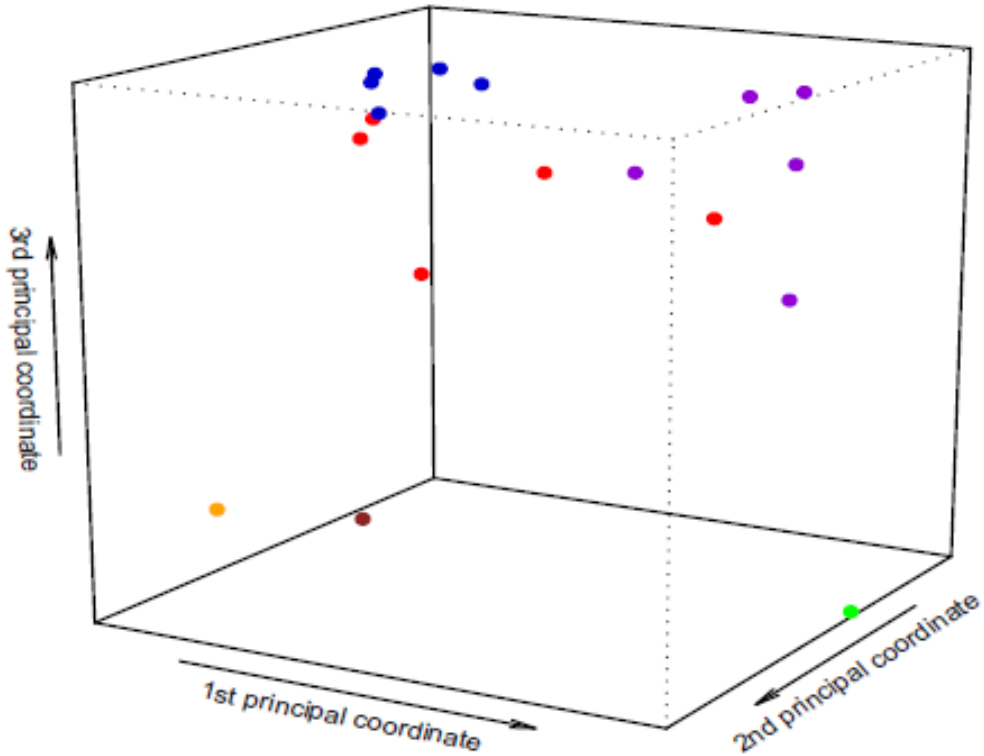


FIGURE 13.2. *Three-dimensional map of 18 world cities using the classical scaling algorithm on airline distances between those cities. The colors reflect the different continents: Asia (purple), North America (red), South America (yellow), Europe (blue), Africa (brown), and Australasia (green).*

Distance Scaling

classical MDS

seeks to find an optimal configuration $\mathbf{x}_i$ that gives $d_{ij} \approx \hat{d}_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$ as close as possible.

Distance Scaling

- Relaxing $d_{ij} \approx \hat{d}_{ij}$ from cMDS by allowing

$$\hat{d}_{ij} \approx f(d_{ij}), \text{ for some monotone function } f.$$

- Called metric MDS if dissimilarities d_{ij} are quantitative
- Called non-metric MDS if dissimilarities d_{ij} are qualitative (e.g. ordinal).
- Unlike cMDS, distance scaling is an optimization process minimizing *stress* function, and is solved by iterative algorithms.

The (usual) metric MDS

Given a (low) dimension p and a monotone function f, metric MDS seeks to find an optimal configuration $\mathbf{X} \subset \mathbb{R}^p$ that gives $f(d_{ij}) \approx \hat{d}_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$ *as close as possible*.

- The function f can be taken to be a parametric monotonic function, such as $f(d_{ij}) = \alpha + \beta d_{ij}$.
- '*As close as possible*' is now explicitly stated by square loss

$$\text{stress} = \mathcal{L}(\hat{d}_{ij}) = \left(\sum_{i<j} (\hat{d}_{ij} - f(d_{ij}))^2 / \sum d_{ij}^2 \right)^{\frac{1}{2}},$$

  and the metric MDS minimizes $\mathcal{L}(\hat{d}_{ij})$ over all $\hat{d}_{ij}$ and α, β.
- The *usual* metric MDS is the special case $f(d_{ij}) = d_{ij}$;
  The usual metric MDS solution (from optimization) $\neq$ that of classical MDS.

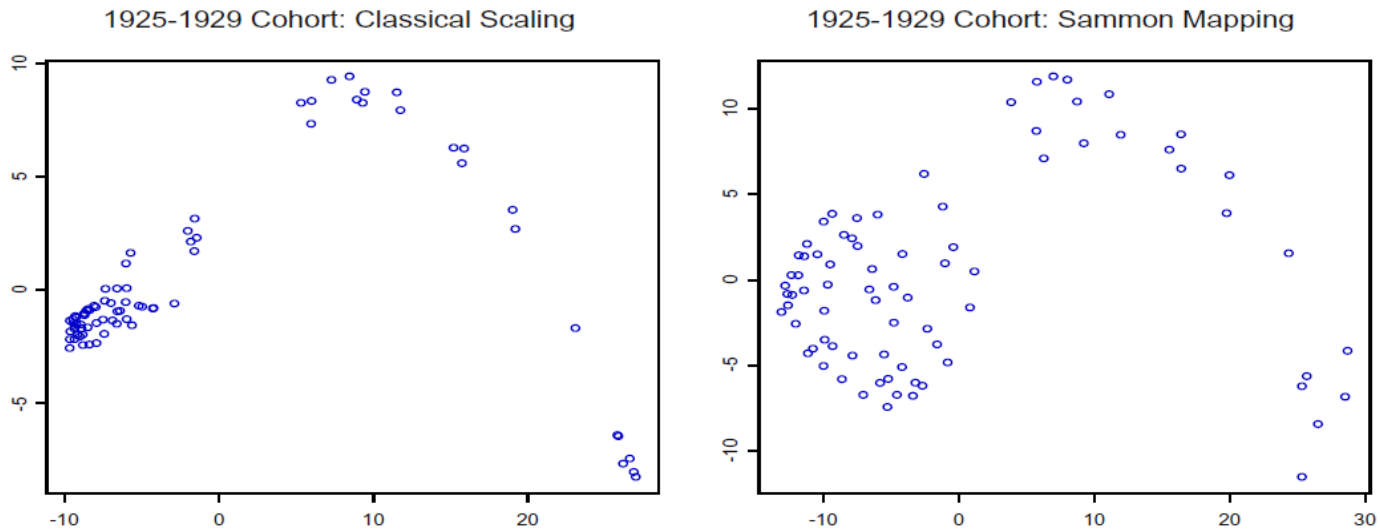
Sammon mapping

- Sammon mapping is a generalization of the usual metric MDS.
- Sammon's stress (to be minimized) is

$$\text{Sammon's stress}(\hat{d}_{ij}) = \frac{1}{\sum_{\ell<k} d_{\ell k}} \sum_{i<j} \frac{(\hat{d}_{ij} - d_{ij})^2}{d_{ij}}$$

- This weighting system normalizes the squared-errors in pairwise distances by using the distance in the original space. As a result, Sammon mapping preserves the small d_{ij} , giving them a greater degree of importance in the fitting procedure than for larger values of d_{ij}
- Optimal solution is found by numerical computation (initial value by cMDS).

cMDS vs. Sammon Mapping



- Izenman Figure 13.9 (lower panel)
- Results of cMDS and Sammon mapping for $p = 2$: Sammon mapping better preserves inter-distances for smaller dissimilarities, while proportionally squeezes the inter-distances for larger dissimilarities.

Non-metric MDS

In many applications of MDS, dissimilarities are known only by their rank order, and the spacing between successively ranked dissimilarities is of no interest or is unavailable

Non-metric MDS

Given a (low) dimension p, non-metric MDS seeks to find an optimal configuration $\mathbf{X} \subset \mathbb{R}^p$ that gives $f(d_{ij}) \approx \hat{d}_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$ as close as possible.

- Unlike metric MDS, here f is much general and is only implicitly defined.
- $f(d_{ij}) = d^*_{ij}$ are called *disparities*, which only preserve the order of d_{ij}, i.e.,

$$\begin{aligned} d_{ij} < d_{k\ell} &\Leftrightarrow f(d_{ij}) \leq f(d_{k\ell}) \\ &\Leftrightarrow d^*_{ij} \leq d^*_{k\ell} \end{aligned} \tag{5}$$

Kruskal's non-metric MDS

- Kruskal's non-metric MDS minimizes the stress-1

$$\text{stress-1}(\hat{d}_{ij}, d^*ij) = \left(\sum_{i<j} \frac{(\hat{d}_{ij} - d^*_{ij})^2}{\sum \hat{d}^2_{ij}} \right)^{\frac{1}{2}}.$$

- Note that the original dissimilarities are only used in checking (5). In fact only the *order* $d_{ij} < d_{k\ell} < \ldots < d_{mf}$ among dissimilarities is needed.
- the function f works as if it were a regression curve (approximated dissimilarities $\hat{d}_{ij}$ as y, disparities d^*_{ij} as $\hat{y}$, and the order of dissimilarities as explanatory)

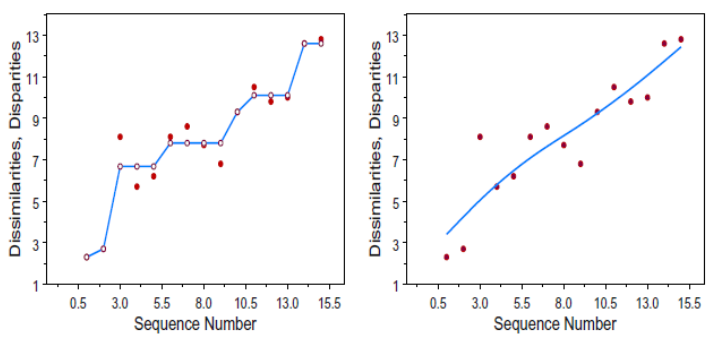


FIGURE 13.10. *Shepard diagram for the artificial example. Left panel: Isotonic regression. Right panel: Monotone spline. Horizontal axis is rank order. For the red points, the vertical axis is the dissimilarity d_{ij}, whereas for the fitted blue points, the vertical axis is the disparity $\hat{d}_{ij}$.*

Example: Letter recognition

Wolford and Hollingsworth (1974) were interested in the confusions made when a person attempts to identify letters of the alphabet viewed for some milliseconds only. A confusion matrix was constructed that shows the frequency with which each stimulus letter was mistakenly called something else. A section of this matrix is shown in the table below.

Letter	C	D	G	H	M	N	Q	W
C	–							
D	5	–						
G	12	2	–					
H	2	4	3	–				
M	2	3	2	19	–			
N	2	4	1	18	16	–		
Q	9	20	9	1	2	8	–	
W	1	5	2	5	18	13	4	–

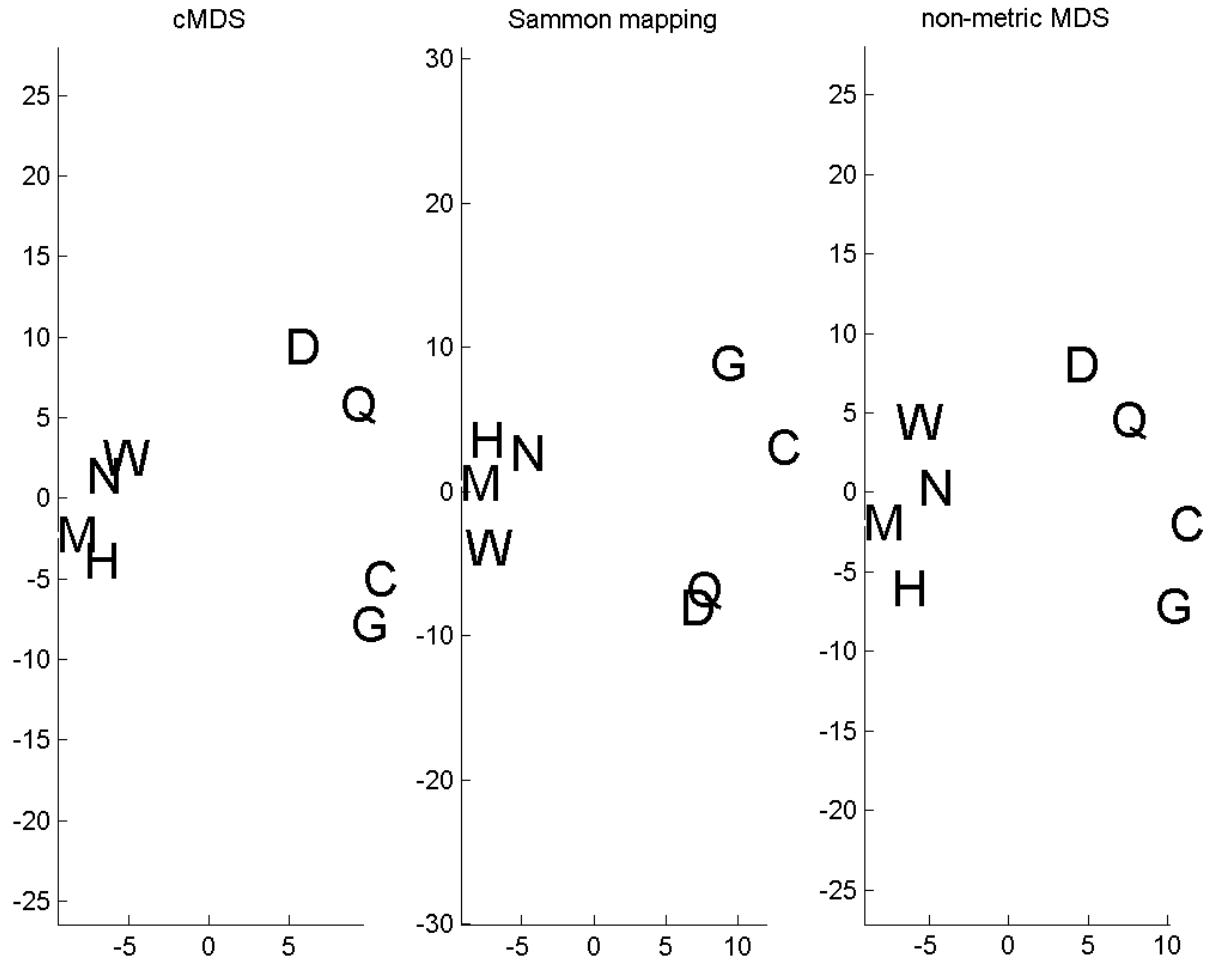
Is this a dissimilarity matrix?

Example: Letter recognition

- How to deduce dissimilarities from a similarity matrix?
  From similarities δ_{ij}, choose a maximum similarity $c \geq \max \delta_{ij}$, so that $d_{ij} = c - \delta_{ij}$, if $i \neq j$, 0 if $i = j$.
- Which method is more appropriate?
  Because we have deduced dissimilarities from similarities, the absolute dissimilarities d_{ij} depend on the value of personally chosen c. This is the case where the non-metric MDS makes most sense.
  However, we will also see that metric scalings (cMDS and Sammon mapping) do the job as well.
- How many dimension?
  By inspection of eigenvalues from the cMDS solution.

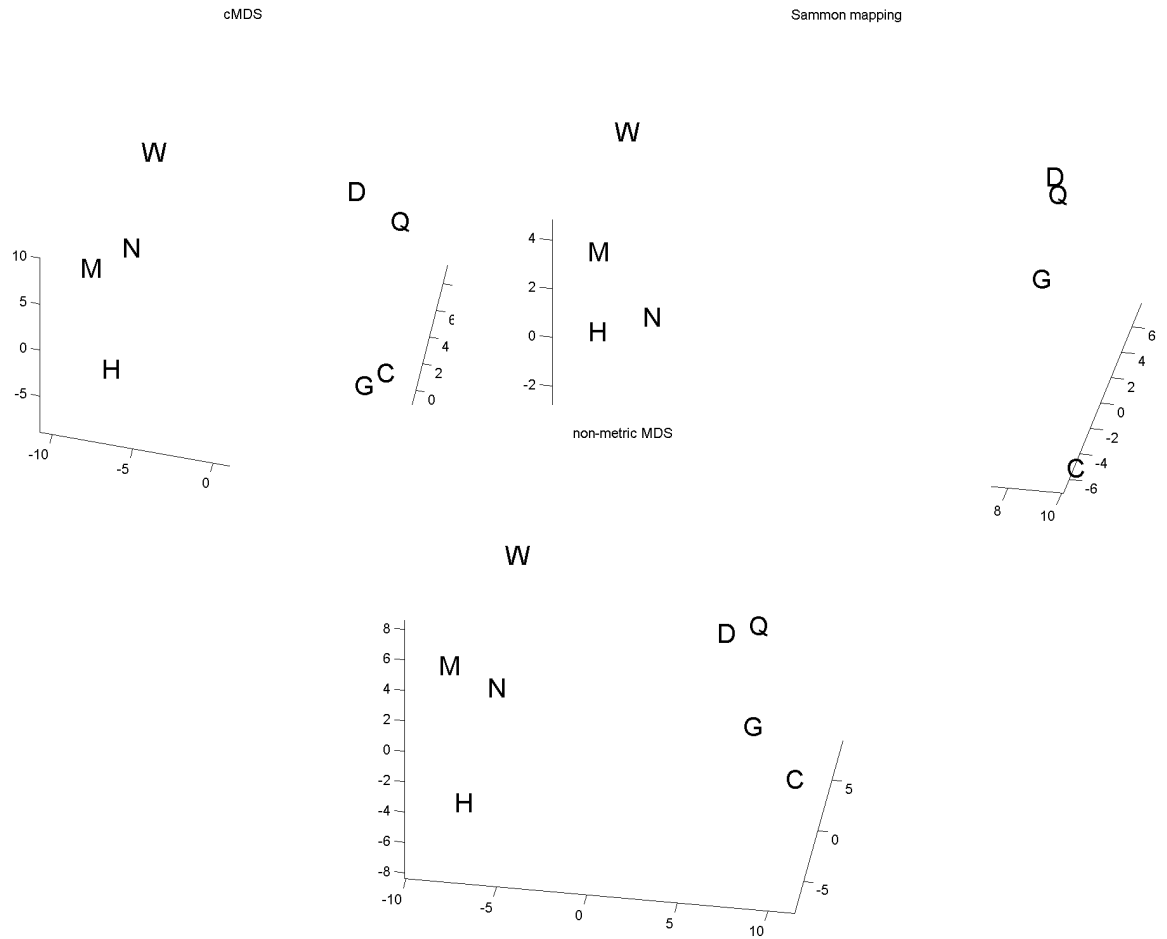
Letter recognition

- First choose $c = 21 = \max \delta_{ij} + 1$.
- Compare MDS with $p = 2$, from cMDS, Sammon mapping, and non-metric scaling (stress1):



Letter recognition:

- First choose $c = 21 = \max \delta_{ij} + 1$.
- Compare MDS with $p = 3$, from cMDS, Sammon mapping, and non-metric scaling (stress1):



Letter recognition:

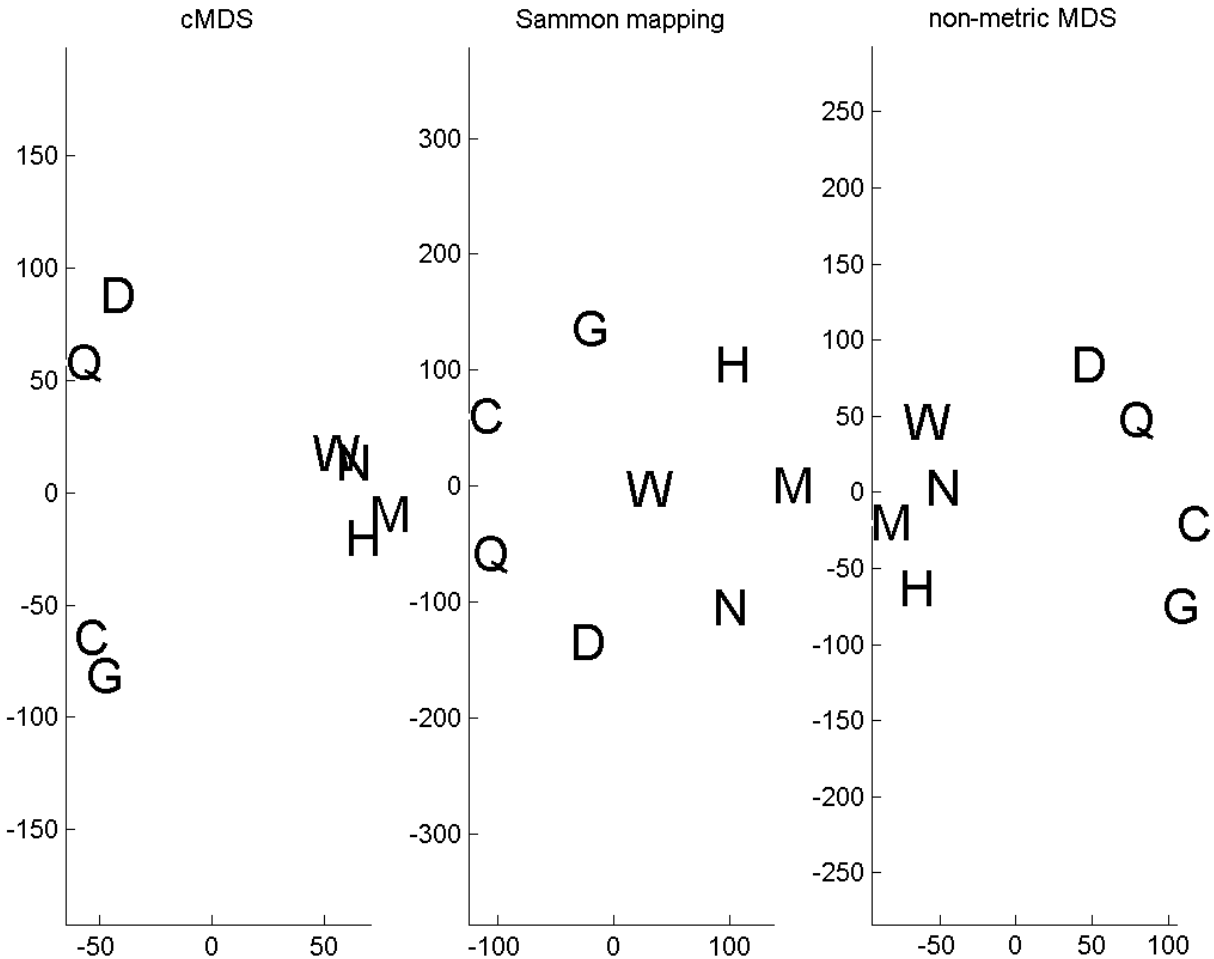
- Do you see any clusters?
- With $c = 21 = \max \delta_{ij} + 1$, the eigenvalues of the Gram-matrix B in the calculation of cMDS are:

```
508.5707
236.0530
124.8229
 56.0627
 39.7347
 -0.0000
-35.5449
-97.1992
```

- The choice of $p = 2$ or $p = 3$ seems reasonable.

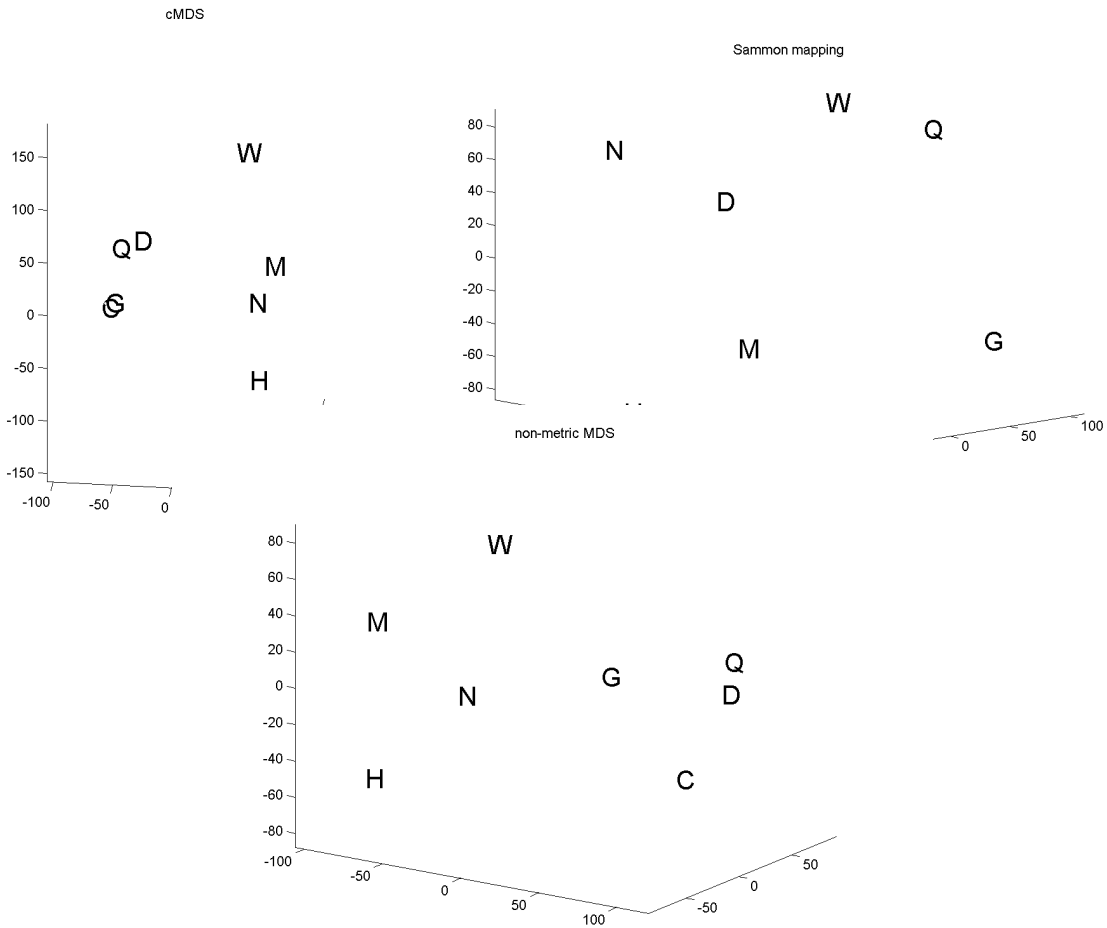
Letter recognition

- Second choice of $c = 210 = \max \delta_{ij} + 190$.
- Compare MDS with $p = 2$, from cMDS, Sammon mapping, and non-metric scaling (stress1):



Letter recognition:

- Second choice of $c = 210 = \max \delta_{ij} + 190$.
- Compare MDS with $p = 3$, from cMDS, Sammon mapping, and non-metric scaling (stress1):



Letter recognition:

- With $c = 210$, the eigenvalues of the Gram-matrix B in the calculation of cMDS are:

```
1.0e+04 *

 2.7210
 2.2978
 2.1084
 1.9623
 1.9133
 1.7696
 1.6842
 0.0000
```

- May need more than $p > 3$ dimensions.

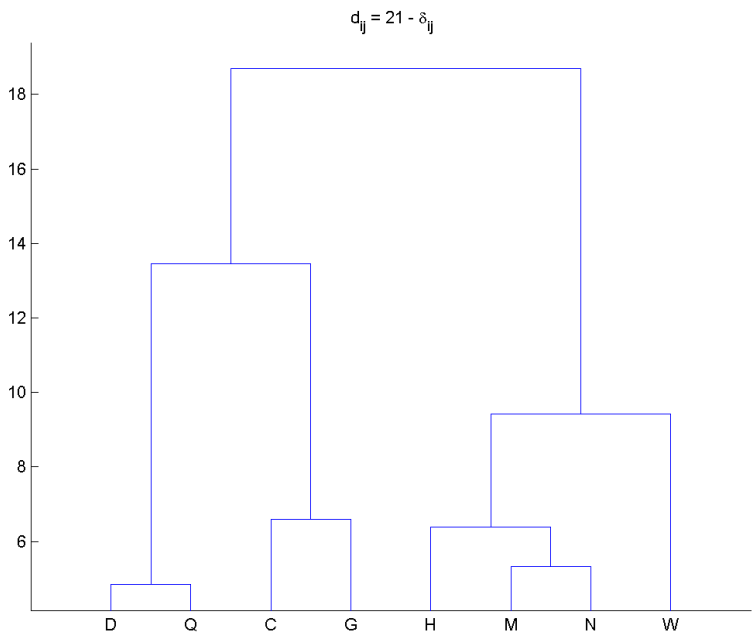
Letter recognition: Summary

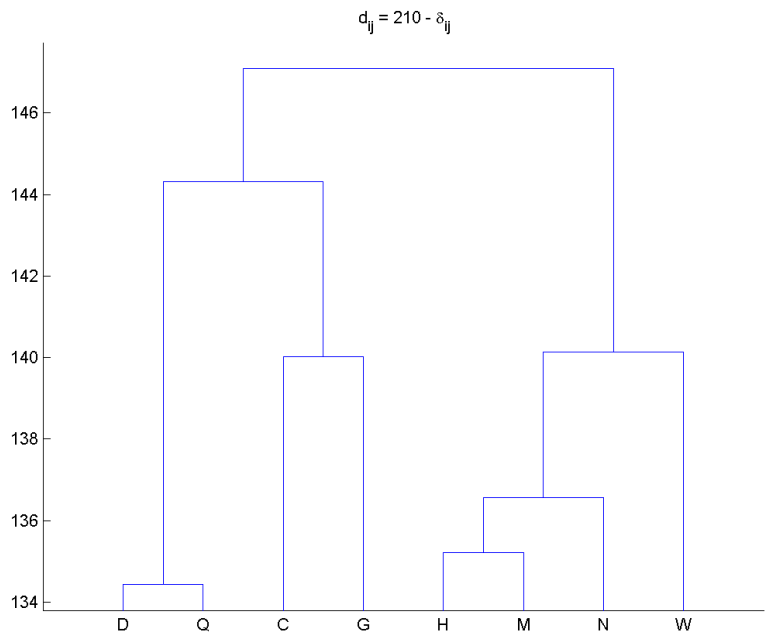
- The structure of the data appropriate for non-metric MDS.
- Kruskal's non-metric scaling:
  1. Appropriate for non-metric dissimilarities (only when their orders are preserved)
  2. Optimization: susceptible to local minima (leading to different configurations);
  3. Time-consuming
- cMDS fast, overall good.
- Sammon mapping fails when $c = 210$.

Letter recognition: Summary

- Clusters (C, G), (D, Q), (H, M, N, W) are confirmed by a cluster analysis for either choice of c.

Use agglomerative hierarchical clustering with average linkage:





MDS in R

```
library(MASS)

# compute dissimilarity matrix from a dataset
d <- dist(swiss)
# d is (n x n-1) lower triangle matrix

cmdscale(d, k =2) # classical MDS
sammon(d,k=1) # Sammon Mapping
isoMDS(d,k=2) # Kruskal's Non-metric MDS
```

Application: Stringing

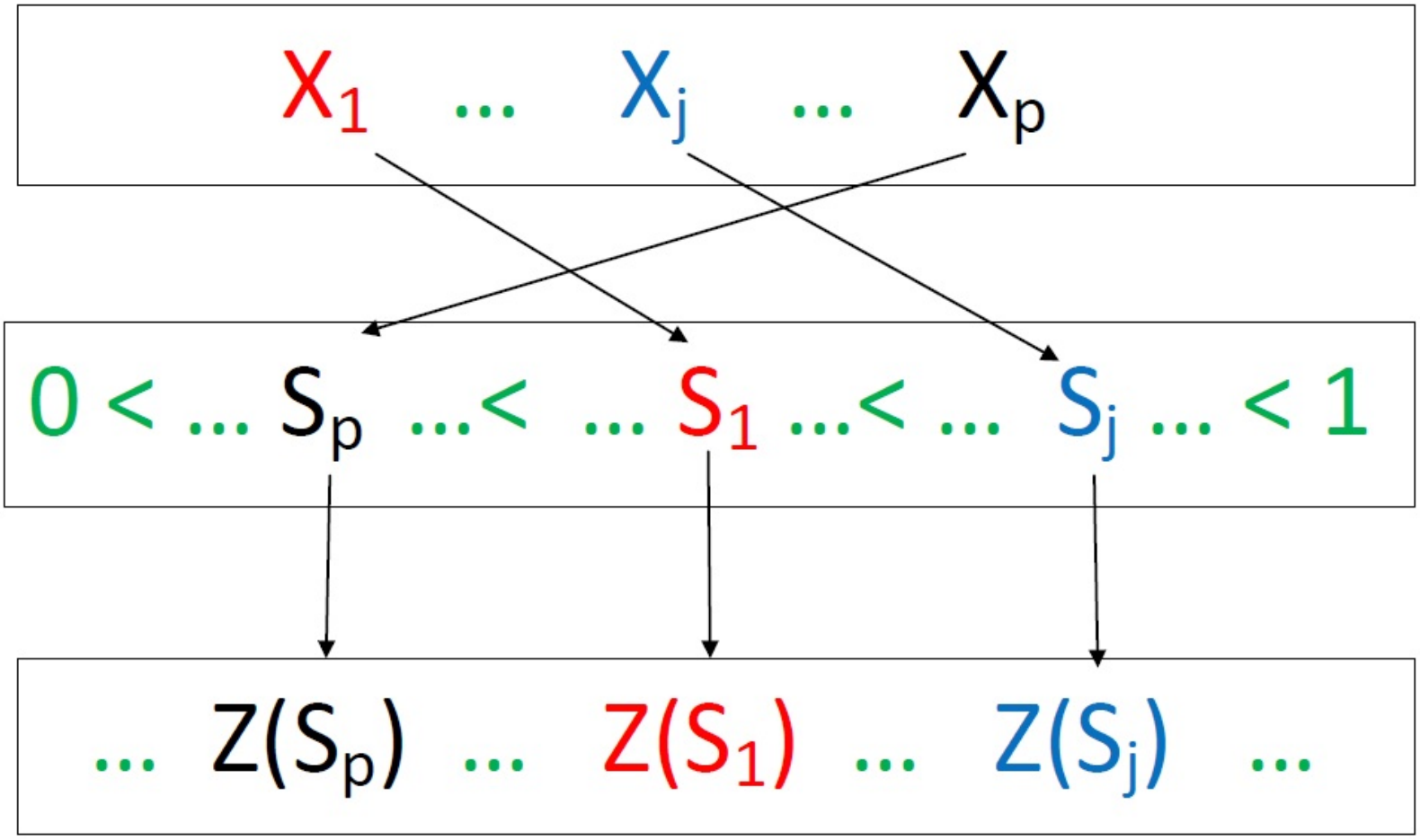
In K. Chen, K. Chen, H.G. Mueller and J.L. Wang (2011). Stringing high dimensional data for functional analysis. JASA 106, 275-284.

Basic idea: Multivariate data to functional data

- Compute $p \times p$ dissimilarity matrix for p variables.
- Use MDS to retrieve 1-dimensional configuration consisting of the p points.
- String variables by the order of variables given by MDS configuration.

Many thanks to K. Chen for sharing the next few slides.

Idea of Stringing



Why Stringing?

- Methods and results from low dimensional data can not be directly applied to large p problems.
- Large p is not a problem for FDA, we have $\sqrt{n}$ consistency for covariance, leading eigenvalues and eigenfunctions.
- Smooth structure: $G(s,t) = \sum_{k=1}^{\infty} \lambda_k \phi_k(s)\phi_k(t)$, with $\sum_{k=1}^{\infty} \lambda_k < \infty$.
- Neighbors defined over a continuum: Smoothing techniques for noisy, irregular or missing data.

How to Perform Stringing

- Define a distance between p predictors and estimate the $p \times p$ distance matrix D.
- Assign a location $s_j \in [0,1]$ to each predictor variable X_j to minimize stress function

$$S_{\mathbf{D}}(s_1, \ldots, s_p) = \sum_{j<k} (|s_j - s_k| - D_{jk})^2.$$

- Computational similar but conceptually different form MDS.
  1. Preserve the dissimilarity between p variables not n samples.
  2. The goal is not dimension reduction. The coordinate s_i is an auxiliary location to define random functions.

Application in High-Dimensional Regression

Two key assumptions essential for current Lasso type methodology:

- Sparsity: Only very few predictors matter
- Uncorrelatedness: Correlations between predictors are small

What if predictors are highly correlated and effects are not sparse?

Stringing:

$$E(Y|X) = g^{-1}(X\beta)$$

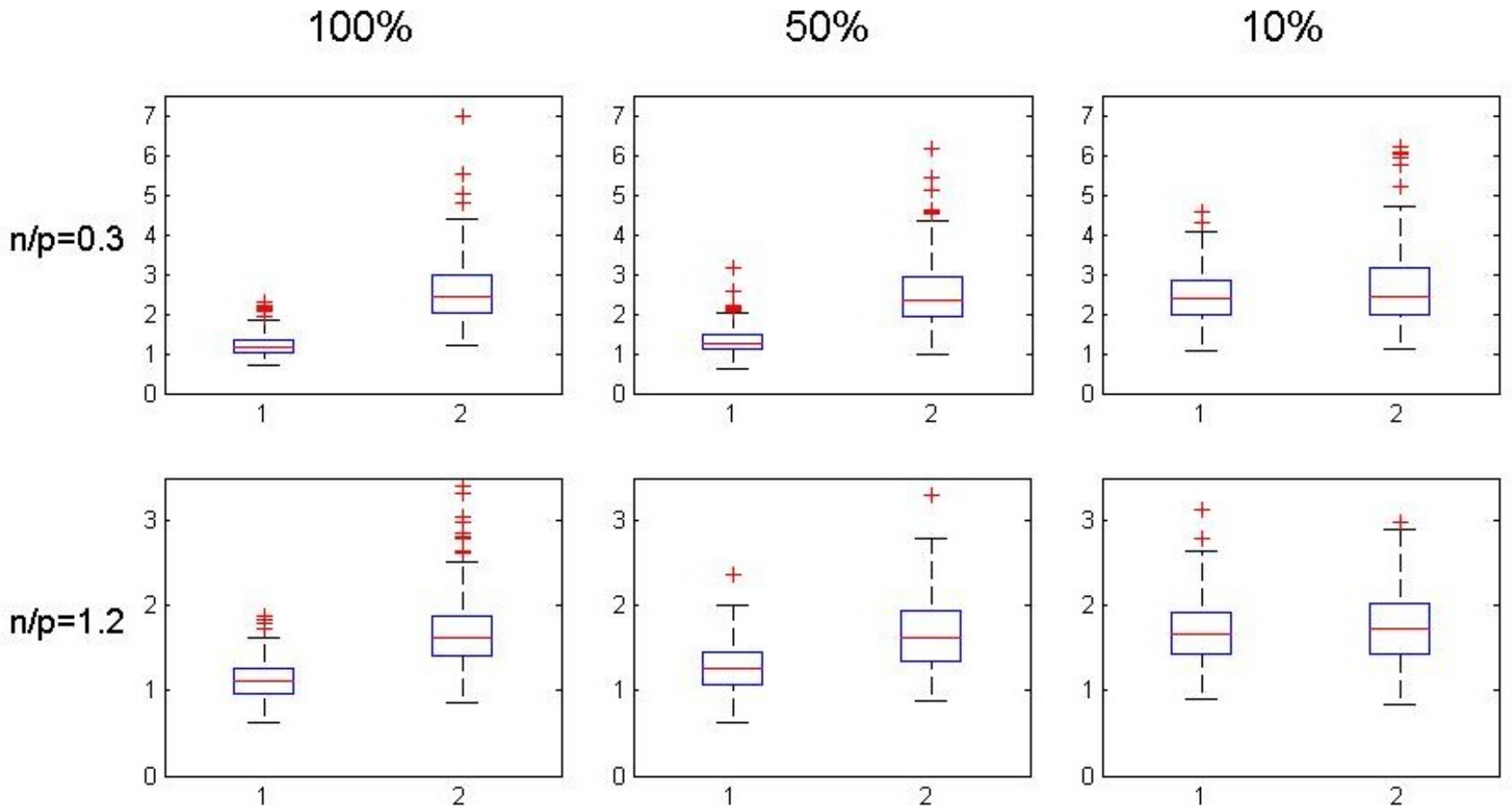
$$\Rightarrow E(Y|Z) = g^{-1}(\mu_Y + \int Z(s)\beta(s)ds)$$

Simulation Settings

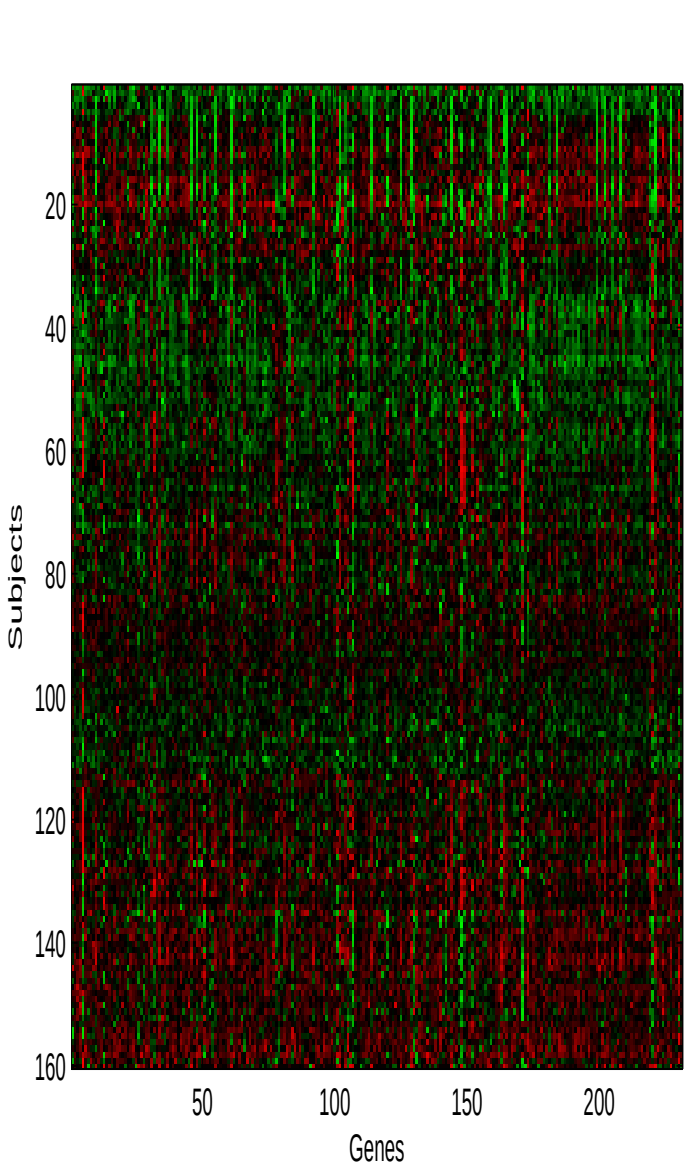
$$Y = X\beta + \varepsilon$$

- $\mathrm{cov}(X_i, X_j) = \sigma(i,j) = \mathscr{U}(0,1)$, chosen as i.i.d. uniform random numbers, projected to non-negative definite matrix space
- Varying sparsity of regression coefficients, generated as $\beta_j \sim \mathscr{U}(0,1), j = 1, \ldots, p$, where the fraction of nonzero coefficients is controlled at 100%, 50%, 10%.
- Varying p, n, test sets of size 50 and 200 simulations

Functional Regression Versus LASSO



Stringing Gene Expressions



Stringing
$\longrightarrow$