Multidimensional Scaling

Mathias Bourel

IMERL - Facultad de Ingeniería, Universidad de la República, Uruguay

May 25, 2021

Introduction

Es otra técnica descriptiva y de interpretación de los datos.

En vez de trabajar directamente con la matriz de datos $n \times p$ disponemos de una matriz $D \in \mathcal{M}_{n\times n}$ de distancia entre los n individuos de la población (distancias entre n candidatos políticos, distancias/similitudes entre n productos fabricados,...).

El objetivo será de representar a D, en un subespacio de dimensión menor, mediante un conjunto de variables ortogonales $y_1, \ldots, y_k$ con $k < n$ (obtenemos entonces una matriz $Y \in \mathcal{M}_{n\times k}$) de manera que las distancia euclideas entre las coordenadas de los elementos respecto a estas variables sean iguales o lo más próximo posible a las distancias o disimilaridades de la matriz original, para ayudarnos en entender la estructura de los datos.

Disimilaridades, semi-métrica y métrica

1. $d : V \times V \to [0, +\infty)$ es una *disimilaridad* si
   - para todo i $d_{ii} = 0$
   - para todos i, j $d_{ij} = d_{ji}$ (simetría)
2. $d : V \times V \to [0, +\infty)$ es una *semi-métrica* si
   - d es disimilaridad
   - para todos i, j, k $d_{ij} \leq d_{ik} + d_{kj}$ (desigualdad triangular)
3. $d : V \times V \to [0, +\infty)$ es una *métrica* o distancia si
   - d es semi-métrica
   - para todos i, j $d_{ij} = 0 \Leftrightarrow i = j$ (reflexiva)
4. $s : V \times V \to [0, +\infty)$ es una *similaridad* si
   - para todos i, j $0 \leq s_{ij} \leq s_{ii} = 1$
   - para todos i, j $s_{ij} = s_{ji}$

Observación: La transformación $d_{ij}^2 = s_{ii} + s_{jj} - 2s_{ij}$ permite obtener una distancia d a partir de una similaridad s, lo cual se expresa en forma matricial por

$$D^{(2)} = 2(\mathbf{1}\mathbf{1}' - \mathcal{S})$$

siendo $\mathcal{S}$ la matriz de similaridades y $D^{(2)}$ la matriz de cuadrados de distancias.

Ejemplos de distancias

(1) Distancias para variables numéricas

- **distancia euclídea:** $d(\mathbf{x_1}, \mathbf{x_2}) = \left(\sum_{j=1}^{p} (x_{1j} - x_{2j})^2 \right)^{1/2}$
- **distancia de Minkowski:** $d(\mathbf{x_1}, \mathbf{x_2}) = \left[\sum_{j=1}^{p} |x_{1j} - x_{2j}|^k \right]^{1/k}$

  Si $k = 1$ es la distancia de Manhattan y si $k = 2$ es la distancia euclídea.
- **distancia de Mahalanobis:** $d(\mathbf{x_1}, \mathbf{x_2}) = (\mathbf{x_1} - \mathbf{x_2})' W^{-1} (\mathbf{x_1} - \mathbf{x_2})$ donde W es la matriz de covarianzas entre las variables. Observar que si la correlación es nula y las variables estandarizadas, tenemos la distancia euclídea.

(2) Distancias para variables binarias. Si $x_1, \ldots, x_p$ son p variables binarias con posibles valores $\{0, 1\}$. Defino

- a=cantidad de variables con respuesta 1 en ambos individuos
- b=cantidad de variables con respuesta 0 en individuo i y 1 en individuo j.
- c=cantidad de variables con respuesta 1 en individuo i y 0 en individuo j.
- d=cantidad de variables con respuesta 0 en ambos individuos.

Observar que $a + b + c + d = p$.

Son coeficientes de similaridad:

$$\text{Sokal y Michener: } s_{ij} = \frac{a+d}{p}, \qquad \text{Jaccard: } s_{ij} = \frac{a}{a+b+c}$$

La distancia de Sokal y Michener representa la totalidad de coincidencias entre los individuos.

Ejemplos de distancias

(3) Distancias para variables mixtas Presencia de variables cualitativas y cuantitativas.
Si

- p_1 es la cantidad de variables cuantitativas,
- p_2 es la cantidad de variables binarias,
- p_3 es la cantidad de variables cualitativas no binarias,
- a es la cantidad de coincidencias 1-1 de las variables binarias,
- d es la cantidad de coincidencias 0-0 de las variables binarias,
- α es la cantidad de coincidencias en las variables cualitativas no binarias,
- G_h es el rango de la $h-$ésima variable cuantitativa

La distancia de Gower se define como $d_{ij}^2 = 1 - s_{ij}$ donde

$$s_{ij} = \frac{\sum_{h=1}^{p_1}(1 - |x_{ih} - x_{jh}|/G_h) + a + \alpha}{p_1 + (p_2 - d) + p_3}$$

Observar que si tenemos una matriz de similaridades $Q = ((q_{ij}))$ donde:

- $0 \leq q_{ij} \leq q_{ii} = 1$ para todo i, j.
- $q_{ij} = q_{ji}$ para todo i, j.

Recordar que planteando que $d_{ij}^2 = q_{ii} + q_{jj} - 2q_{ij} = 2(1 - q_{ij})$ o en notación matricial: $D^{(2)} = 2(\mathbf{1}\mathbf{1}' - Q)$ pasamos de una matriz de similaridades Q a una matriz de cuadrados de distancias $D^{(2)}$.
Supongamos que $Q \in \mathcal{M}_{n\times n}$ la podemos escribir como $Q = (PX)(PX)'$ donde $P = I_n - \frac{1}{n}\mathbf{1}\mathbf{1}'$ entonces $Q = PXX'P$ y sustituyendo en la expresión anterior tenemos

$$D^{(2)} = 2\mathbf{1}\mathbf{1}' - 2PXX'P \Rightarrow PQP = -\frac{1}{2}PD^{(2)}P$$

pero $PQP = Q$ ya que P es idempotente entonces

$$Q = -\frac{1}{2}PD^{(2)}P$$

La clave está ahí: vamos a querer obtener a partir de $D^{(2)}$ una matriz Q semidefinida positiva tal que

$$Q = -\frac{1}{2}PD^{(2)}P$$

Escalado Multidimensional

- El objetivo consiste en obtener una representación euclídea, exacta o aproximada, de los elementos de un conjunto de n puntos a los que se le conoce una matriz de de distancias D. No disponemos de una matriz de datos, si no solamente la distancia entre estos datos.

Escalado Multidimensional

- El objetivo consiste en obtener una representación euclídea, exacta o aproximada, de los elementos de un conjunto de n puntos a los que se le conoce una matriz de de distancias D. No disponemos de una matriz de datos, si no solamente la distancia entre estos datos.
- Sea $D = ((d_{ij}))_{1 \leq i,j \leq n}$ una matriz de distancias.
  Decimos que D tiene representación euclidea de dimensión k si existe un conjunto de n puntos $\mathbf{x}_1, \ldots, \mathbf{x}_n$ del espacio euclídeo $\mathbb{R}^k$ que verifica que las distancias eucídeas entre los $\mathbf{x}_i$ son iguales a las entradas de la matriz $\mathbf{D}$:

$$d_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j) \qquad 1 \leq i, j \leq n$$

Escalado Multidimensional

- El objetivo consiste en obtener una representación euclídea, exacta o aproximada, de los elementos de un conjunto de n puntos a los que se le conoce una matriz de de distancias D. No disponemos de una matriz de datos, si no solamente la distancia entre estos datos.
- Sea $D = ((d_{ij}))_{1 \leq i,j \leq n}$ una matriz de distancias.
  Decimos que D tiene representación euclidea de dimensión k si existe un conjunto de n puntos $\mathbf{x}_1, \ldots, \mathbf{x}_n$ del espacio euclídeo $\mathbb{R}^k$ que verifica que las distancias eucídeas entre los $\mathbf{x}_i$ son iguales a las entradas de la matriz $\mathbf{D}$:

$$d_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j) \qquad 1 \leq i, j \leq n$$

  ¿Cuando una distancia tiene representación euclidea?

Escalado Multidimensional

- El objetivo consiste en obtener una representación euclídea, exacta o aproximada, de los elementos de un conjunto de n puntos a los que se le conoce una matriz de de distancias D. No disponemos de una matriz de datos, si no solamente la distancia entre estos datos.
- Sea $D = ((d_{ij}))_{1 \leq i,j \leq n}$ una matriz de distancias.
  Decimos que D tiene representación euclidea de dimensión k si existe un conjunto de n puntos $\mathbf{x}_1, \ldots, \mathbf{x}_n$ del espacio euclídeo $\mathbb{R}^k$ que verifica que las distancias eucídeas entre los $\mathbf{x}_i$ son iguales a las entradas de la matriz $\mathbf{D}$:

$$d_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j) \qquad 1 \leq i, j \leq n$$

  ¿Cuando una distancia tiene representación euclidea?

Teorema: la matriz de distancias D tiene una representación euclidea de dimensión $k \leq n-1$ si y sólo si

$$Q = -\frac{1}{2} P D^{(2)} P$$

es semidefinida positiva con $k = \text{rango}(Q)$ y $D^{(2)}$ es la matriz de cuadrados de distancias.

Obtención de las coordenadas principales

De ser así, si Q es semidefinida positiva entonces su descomposición espectral es

$$Q = U\Lambda U' = YY'$$

siendo $Y = U\Lambda^{1/2}$ donde $U \in \mathcal{M}_{n\times k}$ es ortogonal formada por los vectores propios asociados a valores propios no nulos de Q y Λ es la matriz diagonal que contiene a los valores propios de Q:

$$\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k \geq \lambda_{k+1} = \cdots = \lambda_n = 0$$

Las n filas de Y son las coordenadas (coordenadas principales) de los individuos cuya matriz de distancias era D, y las dos primeras columnas de Y dan lugar a una representación de los n individuos sobre un plano.

Euclideanización de una distancia

Si la matriz Q no es semidefinida positiva entonces no existe una factorización posible en Y como vimos antes.

¿Qué hacemos?

Euclideanización de una distancia

Si la matriz Q no es semidefinida positiva entonces no existe una factorización posible en Y como vimos antes.

¿Qué hacemos?

Puede haber dos soluciones:

1. **Teorema:** Si Q tiene valores propios negativos entonces haciendo la transformación sobre D:

$$\overline{d}_{ij}^2 = \begin{cases} d_{ij}^2 + c, & i \neq j \\ 0, & i = j \end{cases}$$

   donde $c \geq 2|\lambda|$ siendo λ el valor propio negativo con valor absoluto máximo, obtenemos una matria $\tilde{D}$ que admite una representación euclídea.

2. O sino: si Q tiene k valores propios positivos más grandes que el resto, entonces si $Y_{n\times k} = V_{n\times k}\Lambda_{k\times k}^{1/2}$ se considera la aproximación $Q = -\frac{1}{2}PD^{(2)}P \approx (V_{n\times k}\Lambda_{k\times k}^{1/2})(V_{n\times k}\Lambda_{k\times k}^{1/2})$

Varias observaciones

- Las filas de Y verifican que $d_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j)$.

Varias observaciones

- Las filas de Y verifican que $d_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j)$.
- Las columnas de Y son ortogonales por construcción y tienen media cero:
  Observemos primero que como $P\mathbf{1} = \mathbf{0}$ entonces $Q\mathbf{1} = \mathbf{0}$ y entonces $\mathbf{1}$ es un vector propio de Q asociado al valor propio 0. Entonces $\mathbf{1}$ es ortogonal a todos los otros vectores propios de Q asociados a otros valores y por lo tanto $\mathbf{1}'U$ es una matriz nula.
  Entonces:
  $$\bar{\mathbf{y}} = \frac{1}{n}1'Y = \frac{1}{n}1'U\Lambda^{1/2} = 0$$

Varias observaciones

- Las filas de Y verifican que $d_{ij}^2 = (\mathbf{x}_i - \mathbf{x}_j)'(\mathbf{x}_i - \mathbf{x}_j)$.
- Las columnas de Y son ortogonales por construcción y tienen media cero:
  Observemos primero que como $P\mathbf{1} = \mathbf{0}$ entonces $Q\mathbf{1} = \mathbf{0}$ y entonces $\mathbf{1}$ es un vector propio de Q asociado al valor propio 0. Entonces $\mathbf{1}$ es ortogonal a todos los otros vectores propios de Q asociados a otros valores y por lo tanto $\mathbf{1}'U$ es una matriz nula.
  Entonces:
  $$\bar{\mathbf{y}} = \frac{1}{n}1'Y = \frac{1}{n}1'U\Lambda^{1/2} = 0$$
- Las variables $Y_1, \ldots, Y_p$ son incorreladas pues:
  $$Var(Y) = \frac{1}{n}Y'Y = \frac{1}{n}\Lambda$$
  y por lo tanto sus varianzas proporcionales a los valores propios de Q.
- Observar que las variables Y no son las originales, en realidad son las componentes principales de una matriz (teórica) X y son combinaciones lineales de las variables originales.

Relación con ACP

Si partimos de la matriz de datos $\tilde{X}$ y calculamos a partir de ella $D^{(2)}$ y después mediante el procedimiento anterior calculamos Q y luego Y entonces no obtenemos las variables originales de $\tilde{X}$, pero sus componentes principales. Esto es porque:

1.
   - Si a_1 es vector propio de $S = \frac{1}{n}\tilde{X}'\tilde{X}$ asociado a λ_1 entonces $\tilde{X}a_1$ es vector propio de $Q = \tilde{X}\tilde{X}'$ asociado a $n\lambda_1$.
   - Si u_1 es vector propio de $Q = \tilde{X}\tilde{X}'$ asociado a λ_1 entonces $\tilde{X}'u_1$ es vector propio de S asociado a $\frac{\lambda_1}{n}$.
2. Recordar que u_1 y z_1 son colineales ($z_1 = \sqrt{\lambda_1}u_1$), entonces, para simplificar notaciones, si consideramos solo dos direcciones:

$$Q = YY' = \begin{pmatrix} | & | \\ y_1 & y_2 \\ | & | \end{pmatrix}\begin{pmatrix} | & | \\ y_1 & y_2 \\ | & | \end{pmatrix}' = \begin{pmatrix} | & | \\ u_1 & u_2 \\ | & | \end{pmatrix}\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}\begin{pmatrix} | & | \\ u_1 & u_2 \\ | & | \end{pmatrix}'$$

$$= \begin{pmatrix} | & | \\ z_1/\sqrt{\lambda_1} & z_2/\sqrt{\lambda_2} \\ | & | \end{pmatrix}\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}\begin{pmatrix} | & | \\ z_1/\sqrt{\lambda_1} & z_2/\sqrt{\lambda_2} \\ | & | \end{pmatrix}$$

$$= \begin{pmatrix} | & | \\ z_1 & z_2 \\ | & | \end{pmatrix}\begin{pmatrix} | & | \\ z_1 & z_2 \\ | & | \end{pmatrix}' = ZZ'$$

Resumen MDS

Las diferentes etapas son:

1. Calcular la matriz de cuadrados de distancias $D^{(2)}$.
2. Calcular la matriz: $Q = -\frac{1}{2} P D^{(2)} P$ siendo $P = I_n - \frac{1}{n}\mathbf{1}\mathbf{1}'$
3. Q es diagonalizable , pero hay que distinguir si Q es semidefinida positiva (todos los valores propios son no negativos) o no (algún valor propio es negativo)
   - si Q es semidefinida positiva, decimos que D tiene representación euclídea y Q se puede diagonalizar como $Q = U\Lambda U' = (U\Lambda^{1/2})(U\Lambda^{1/2})' = YY'$
   - si Q no es semidefinida positiva, o hacemos una modificaci'on sobre la matriz de distancia o nos quedamos con los r valores propios más grandes y positivos y aproximo Q por $(U_r\Lambda_r^{1/2})(U_r\Lambda_r^{1/2})' = YY'$ donde U_r tiene como columnas los vectores propios asociados a los r valores propios conservados.
4. La columna k-esima de Y es $y_k = \sqrt{\lambda_k} v_k = \tilde{X} a_k$ siendo a_k el k-esimo vector propio de la matriz de varianzas covarianzas de una matriz de datos centrados $\tilde{X}$ Las filas de $Y = U\Lambda^{1/2}$ son las coordenadas principales (euclideas) de los elementos del conjunto $\mathbf{x_1}, \ldots, \mathbf{x}_n$

Evaluación del MDS

Una medida de la precisión conseguida mediante la aproximación a partir de los valores propios positivos de la matriz de similitud es el coeficiente dado por el *coeficiente de Mardia*:

$$m_k = \frac{\sum_{j=1}^{k} \lambda_i}{\sum |\lambda_j|} \quad k = 1, \ldots, p$$

Si es mayor que 0.8, la aproximación es buena.

Ejemplo

Consideramos el conjunto de individuos {león, girafa, vaca, oveja, gato, hombre} y medimos la siguientes variables aleatorias:

- X_1 tiene cola
- X_2 es salvaje
- X_3 tiene cuello largo
- X_4 es animal de granja
- X_5 es carnívoro
- X_6 camina sobre cuatro patas

Ejemplo

Consideramos el conjunto de individuos {león, girafa, vaca, oveja, gato, hombre} y medimos la siguientes variables aleatorias:

- X_1 tiene cola
- X_2 es salvaje
- X_3 tiene cuello largo
- X_4 es animal de granja
- X_5 es carnívoro
- X_6 camina sobre cuatro patas

La matriz de datos es:

Ejemplo

Consideramos el conjunto de individuos {león, girafa, vaca, oveja, gato, hombre} y medimos la siguientes variables aleatorias:

- X_1 tiene cola
- X_2 es salvaje
- X_3 tiene cuello largo
- X_4 es animal de granja
- X_5 es carnívoro
- X_6 camina sobre cuatro patas

La matriz de datos es:

$$X = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Ejemplo

Usame el coeficiente de similaridad de Sokal y Michener $S = \frac{a+d}{p}$ donde $a = XX'$ (a es la cantidad de variables con respuesta 1 en ambos individuos), $d = (1_n 1_p^{'} - X)(1_n 1_p^{'} - X)'$ (es el nmero de variable con respuesta 0 en ambos individuos), $p = 6$ es el número de variables observadas y $n = 6$ la cantidad de individuos.

Ejemplo

Usame el coeficiente de similaridad de Sokal y Michener $S = \frac{a+d}{p}$ donde $a = XX'$ (a es la cantidad de variables con respuesta 1 en ambos individuos), $d = (1_n 1_p^{'} - X)(1_n 1_p^{'} - X)'$ (es el nmero de variable con respuesta 0 en ambos individuos), $p = 6$ es el número de variables observadas y $n = 6$ la cantidad de individuos.

Entonces la matriz de similaridad es

$$S = \begin{pmatrix} 1 & 0.67 & 0.5 & 0.5 & 0.83 & 0.50 \\ 0.67 & 1 & 0.5 & 0.5 & 0.5 & 0.17 \\ 0.5 & 0.5 & 1 & 1 & 0.67 & 0.33 \\ 0.5 & 0.5 & 1 & 1 & 0.67 & 0.33 \\ 0.83 & 0.5 & 0.67 & 0.67 & 1 & 0.67 \\ 0.5 & 0.17 & 0.33 & 0.33 & 0.67 & 1 \end{pmatrix}$$

Ejemplo

Usando la transformación $d_{ij}^2 = s_{ii} + s_{jj} - 2s_{ij}$ que en notación matricial es

$$D^{(2)} = 2(1_n 1_n^{'} - S)$$

se obtiene la matriz de distancias al cuadrado:

Ejemplo

Usando la transformación $d_{ij}^2 = s_{ii} + s_{jj} - 2s_{ij}$ que en notación matricial es

$$D^{(2)} = 2(1_n 1_n^{'} - S)$$

se obtiene la matriz de distancias al cuadrado:

$$D^{(2)} = \begin{pmatrix} 0 & 0.67 & 1 & 1 & 0.33 & 1 \\ 0.67 & 0 & 1 & 1 & 1 & 0.67 \\ 1 & 1 & 0 & 0 & 0.67 & 1.33 \\ 1 & 1 & 0 & 0 & 0.67 & 1.33 \\ 0.33 & 1 & 0.67 & 0.67 & 0 & 0.67 \\ 1 & 1.67 & 1.33 & 1.33 & 0.67 & 0 \end{pmatrix}$$

Ejemplo

Usando la transformación $d_{ij}^2 = s_{ii} + s_{jj} - 2s_{ij}$ que en notación matricial es

$$D^{(2)} = 2(1_n 1_n^{'} - S)$$

se obtiene la matriz de distancias al cuadrado:

$$D^{(2)} = \begin{pmatrix} 0 & 0.67 & 1 & 1 & 0.33 & 1 \\ 0.67 & 0 & 1 & 1 & 1 & 0.67 \\ 1 & 1 & 0 & 0 & 0.67 & 1.33 \\ 1 & 1 & 0 & 0 & 0.67 & 1.33 \\ 0.33 & 1 & 0.67 & 0.67 & 0 & 0.67 \\ 1 & 1.67 & 1.33 & 1.33 & 0.67 & 0 \end{pmatrix}$$

Los valores propios de $Q = -\frac{1}{2} P D^{(2)} P$ son:

$$1\, 0.7958,\ 0.333,\ 0.0931,\ 0,\ 0$$

Ejemplo

Usando la transformación $d_{ij}^2 = s_{ii} + s_{jj} - 2s_{ij}$ que en notación matricial es

$$D^{(2)} = 2(1_n 1_n^{'} - S)$$

se obtiene la matriz de distancias al cuadrado:

$$D^{(2)} = \begin{pmatrix} 0 & 0.67 & 1 & 1 & 0.33 & 1 \\ 0.67 & 0 & 1 & 1 & 1 & 0.67 \\ 1 & 1 & 0 & 0 & 0.67 & 1.33 \\ 1 & 1 & 0 & 0 & 0.67 & 1.33 \\ 0.33 & 1 & 0.67 & 0.67 & 0 & 0.67 \\ 1 & 1.67 & 1.33 & 1.33 & 0.67 & 0 \end{pmatrix}$$

Los valores propios de $Q = -\frac{1}{2} P D^{(2)} P$ son:

$$1\, 0.7958,\ 0.333,\ 0.0931,\ 0,\ 0$$

Existe una representación euclidea de D de dimensión 4 (por ejemplo). Las coordenadas principales son las filas de la matriz :

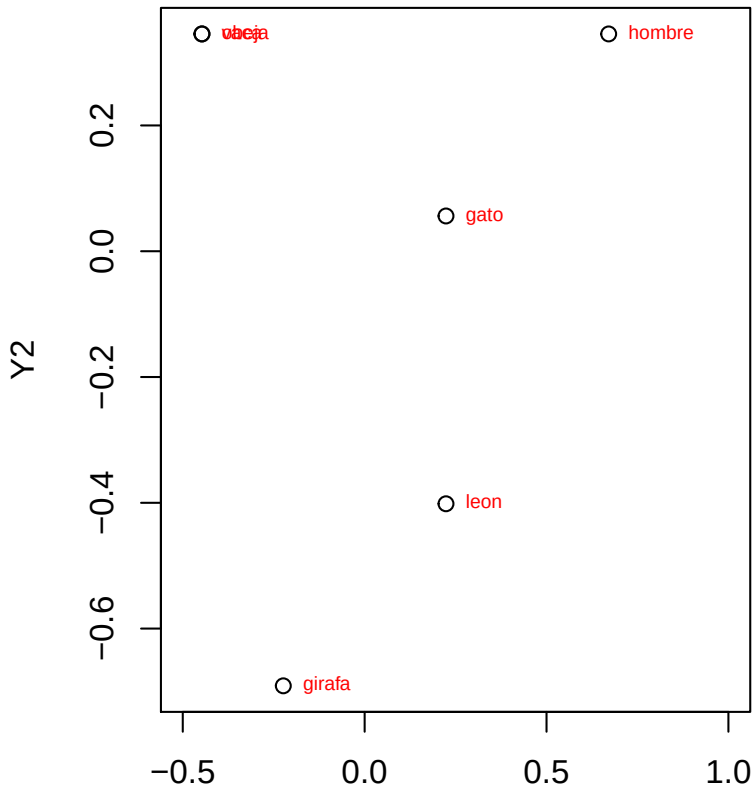
$$\begin{pmatrix} 0.22361 & -0.35823 & 0.86603 & 1.9993 \\ -0.22361 & -0.61643 & -0.86603 & -0.77460 \\ -0.44721 & 0.30822 & 0 & 0.38730 \\ -0.44721 & 0.30822 & 0 & 0.38730 \\ 0.22361 & 0.050016 & 0.86603 & -0.23866 \\ 0.67082 & 0.30822 & -0.86603 & 0.38730 \end{pmatrix}$$

Ejemplo en R

```
> X=matrix(c(1,1,0,0,1,1,
+            1,1,1,0,0,1,
+            1,0,0,1,0,1,
+            1,0,0,1,0,1,
+            1,0,0,0,1,1,
+            0,0,0,0,1,0),6,6,byrow=T)
> library(clusterSim)
> #Matriz de distancia al cuadrado:
>   D2= as.matrix(2* dist.binary(X,method=2,diag=T,upper=T)^2)
>
>   P=diag(1,6)-1/6*rep(1,6)%*%t(rep(1,6))
>   Q=-0.5 * P%*%D2%*%P
>
>   eigen (Q)
eigen() decomposition
$values
[1]  1.000000e+00  7.958086e-01  3.333333e-01  9.308026e-02  8.881784e-16 -7.849624e

$vectors
           [,1]        [,2]          [,3]       [,4]       [,5]          [,6]
[1,]  0.2236068 -0.40156869  5.000000e-01  0.6099803 -0.4082483  0.000000e+00
[2,] -0.2236068 -0.69100302 -5.000000e-01 -0.2363221 -0.4082483 -5.417942e-16
[3,] -0.4472136  0.34550151 -3.608225e-16  0.1181611 -0.4082483 -7.071068e-01
[4,] -0.4472136  0.34550151 -1.387779e-16  0.1181611 -0.4082483  7.071068e-01
[5,]  0.2236068  0.05606718  5.000000e-01 -0.7281413 -0.4082483 -8.819857e-16
[6,]  0.6708204  0.34550151 -5.000000e-01  0.1181611 -0.4082483 -2.806521e-16
```

Ejemplo

```
> nombres=c('leon', 'girafa', 'vaca','obeja', 'gato', 'hombre')
> plot(x=eigen(Q)$vector[,1],y=eigen(Q)$vector[,2],xlim=c(-0.5,1),xlab='Y1',ylab='Y2
>
> text(x=eigen(Q)$vector[,1],y=eigen(Q)$vector[,2],
+      labels = nombres,
+      cex = 0.6, pos = 4, col = "red")
```



Ejemplo con la funcin de R

```
> ?cmdscale
> #Con la funcin mds
> fit=cmdscale(d=sqrt(D2),eig=TRUE, k=2)
> fit
$points
         [,1]        [,2]
1  0.2236068 -0.35823182
2 -0.2236068 -0.61643071
3 -0.4472136  0.30821536
4 -0.4472136  0.30821536
5  0.2236068  0.05001647
6  0.6708204  0.30821536

$eig
[1] 1.000000e+00 7.958086e-01 3.333333e-01 9.308026e-02 1.110223e-15 7.415943e-17

$x
NULL

$ac
[1] 0

$GOF
[1] 0.8081139 0.8081139
```

El coeficiente de Mardia es

$$m_2 = \frac{\lambda_1 + \lambda_2}{\sum_{i=1}^{6} \lambda_i} \approx 0.8081139$$

En R se calcula como, y est dentro de lo que devuelve la función:

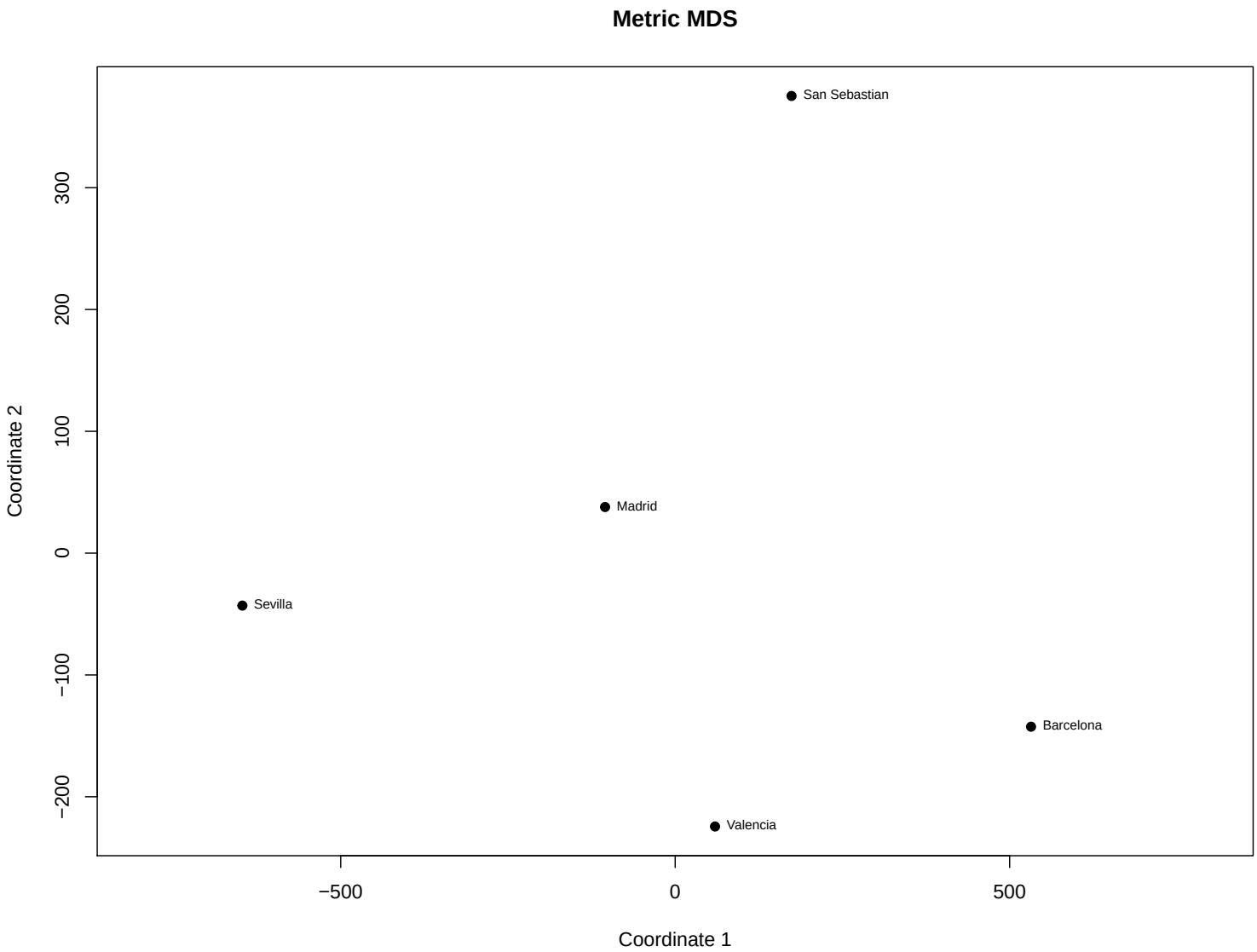
```
> (fit$eig[1]+fit$eig[2])/sum(fit$eig)
[1] 0.8081139
```

Ejemplo en R

Classical MDS

```
#Classical MDS
# N rows (objects) x p columns (variables)
# each row identified by a unique row name

#d <- dist(mydata) # euclidean distances between the rows
d=matrix(c(0,639,606,1181,364,639,0,474,542,355,606,474,0,908,
597,1181,542,908,0,679,364,355,597,639,0),nrow=5,ncol=5)
rownames(d)=c( "Barcelona", "Madrid","San Sebastian","Sevilla","Valencia")

#fit <- cmdscale(d,eig=TRUE, k=2) # k is the number of dim
fit=cmdscale(d,eig=TRUE, k=2)

#fit # view results
fit

# plot solution
x <- fit$points[,1]
y <- fit$points[,2]
plot(x, y, pch=19, xlab="Coordinate 1", ylab="Coordinate 2",
     main="Metric MDS", type="p")
text(x, y, pos=4, labels = row.names(d), cex=.7)

x <- x
y <- 0 - y
plot(x, y, pch=19, xlab="Coordinate 1", ylab="Coordinate 2",
     main="Metric MDS", type="p",xlim=c(-800,800))
```

Example in R


Metric MDS
San Sebastian
Madrid
Sevilla
Barcelona
Valencia
Coordinate 2
300
200
100
0
−100
−200
−500
0
500
Coordinate 1

Example in R

```
>cbind(x,y)
                      x           y
Barcelona      531.99080 -142.50059
Madrid        -104.79503   37.84071
San Sebastian  173.93435  375.30567
Sevilla       -646.99967  -43.06934
Valencia        59.57916 -224.40527
```

Example in R

```
> d
              [,1] [,2] [,3] [,4] [,5]
Barcelona        0  639  606 1181  364
Madrid         639    0  474  542  355
San Sebastian  606  474    0  908  597
Sevilla       1181  542  908    0  639
Valencia       364  355  597  679    0

> d^2
                [,1]   [,2]   [,3]    [,4]   [,5]
Barcelona          0 408321 367236 1394761 132496
Madrid        408321      0 224676  293764 126025
San Sebastian 367236 224676      0  824464 356409
Sevilla      1394761 293764 824464       0 408321
Valencia      132496 126025 356409  461041      0

>uno=rep(1,5)
>ones=uno%*%t(uno)
>P=diag(1,5)-ones/5
> Q=-1/2 * P * d^2 * P
```

Observar que Q no es semidefinida positiva

Referencias

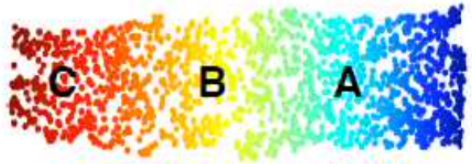
- Distancias estadísticas y Escalado Multidimensional, Aurea Grané, Departamento de Estadística de la Universidad Carlos III de Madrid.
- Análisis de Datos Multivariantes, Daniel Peña, Mac Graw, 2001
- 100 problemas resueltos de Estadística Multivariante, Amparo Baíllo y Aurea Grané, Delta publicaciones, 2008.

Manifold Learning

$d(A,C) < d(A,B)$

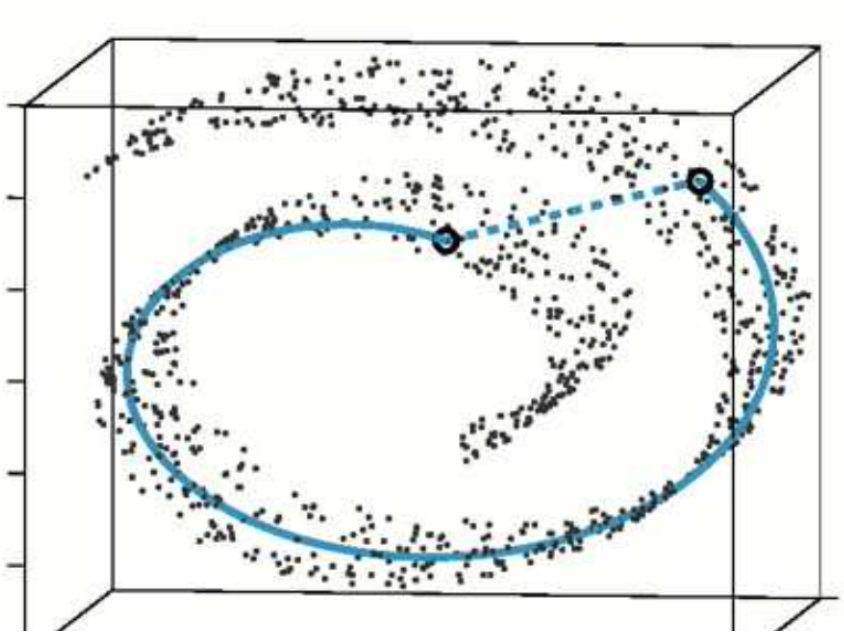
$d(A,C) > d(A,B)$

Manifold Learning



$d(A,C) < d(A,B)$

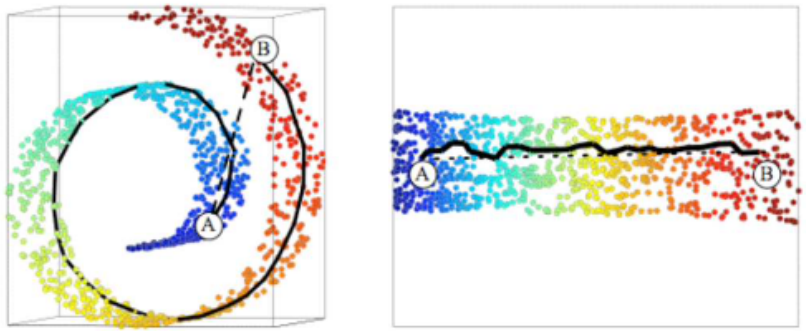
$d(A,C) > d(A,B)$



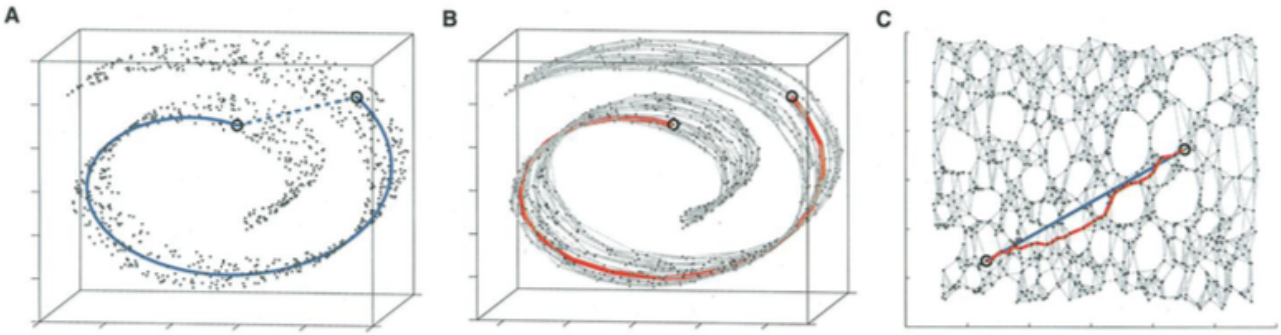
Isomap

En vez de usar la distancia euclidea que no es adecuada, se usa la distancia geodesica para armar la matriz Dy se utiliza MDS sobre la matriz de distancias geodsicas calculadas sobre la variedad a la que pertenecen los datos. El problema es que no se conoce la variedad.
Como primera etapa se construye un grafo de adyacencia, utilizándolo como insumo para estimar las distancias geodésicas a partir del camino más corto entre puntos del grafo. Entonces surgen dos posibilidades para la elaboración del mismo:

1. ϵ-Isomap, donde los puntos que se consideran directamente conectados a otro punto son aquellos cuya distancia euclídea en el espacio de origen es menor a ϵ.
2. K-Isomap, donde el punto i con el punto j si i es uno de los K vecinos ms prximos de j.



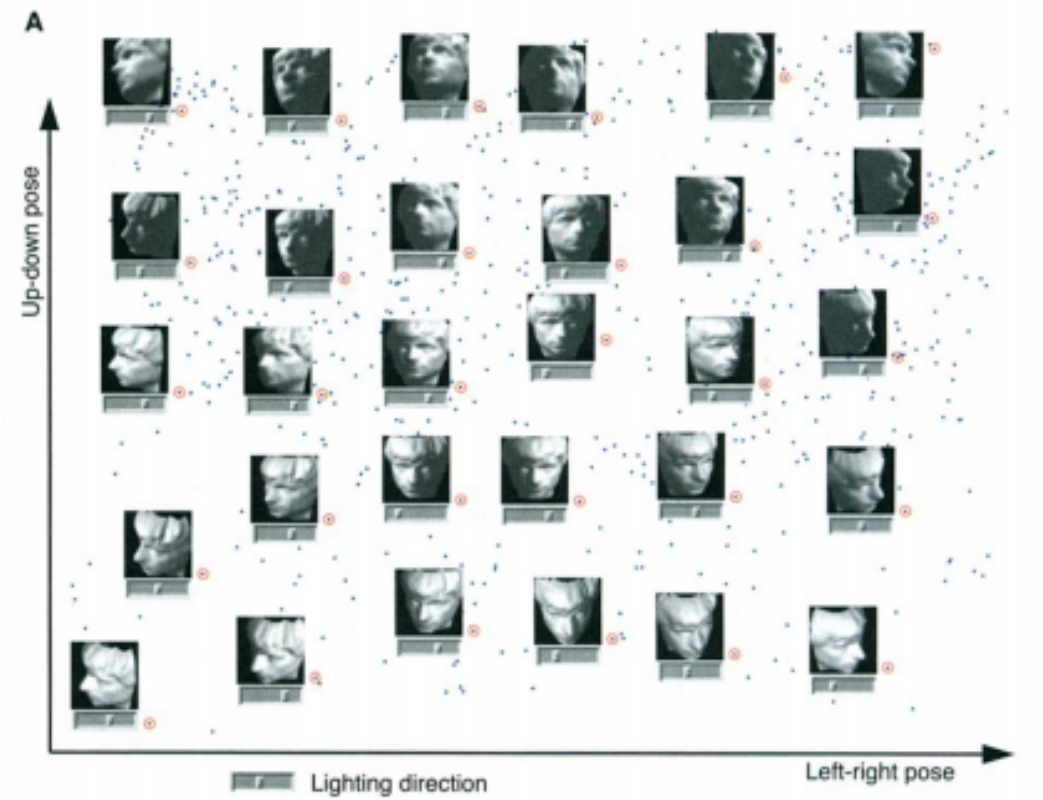
Isomap



La segunda etapa consiste en aplicar MDS a la matriz de distancias geodsicas. Del mismo modo que en MDS, si se toman los k vectores propios asociados a los k valores propios ms significativos, se obtienen las representaciones de los puntos originales en el nuevo espacio de dimensión d.

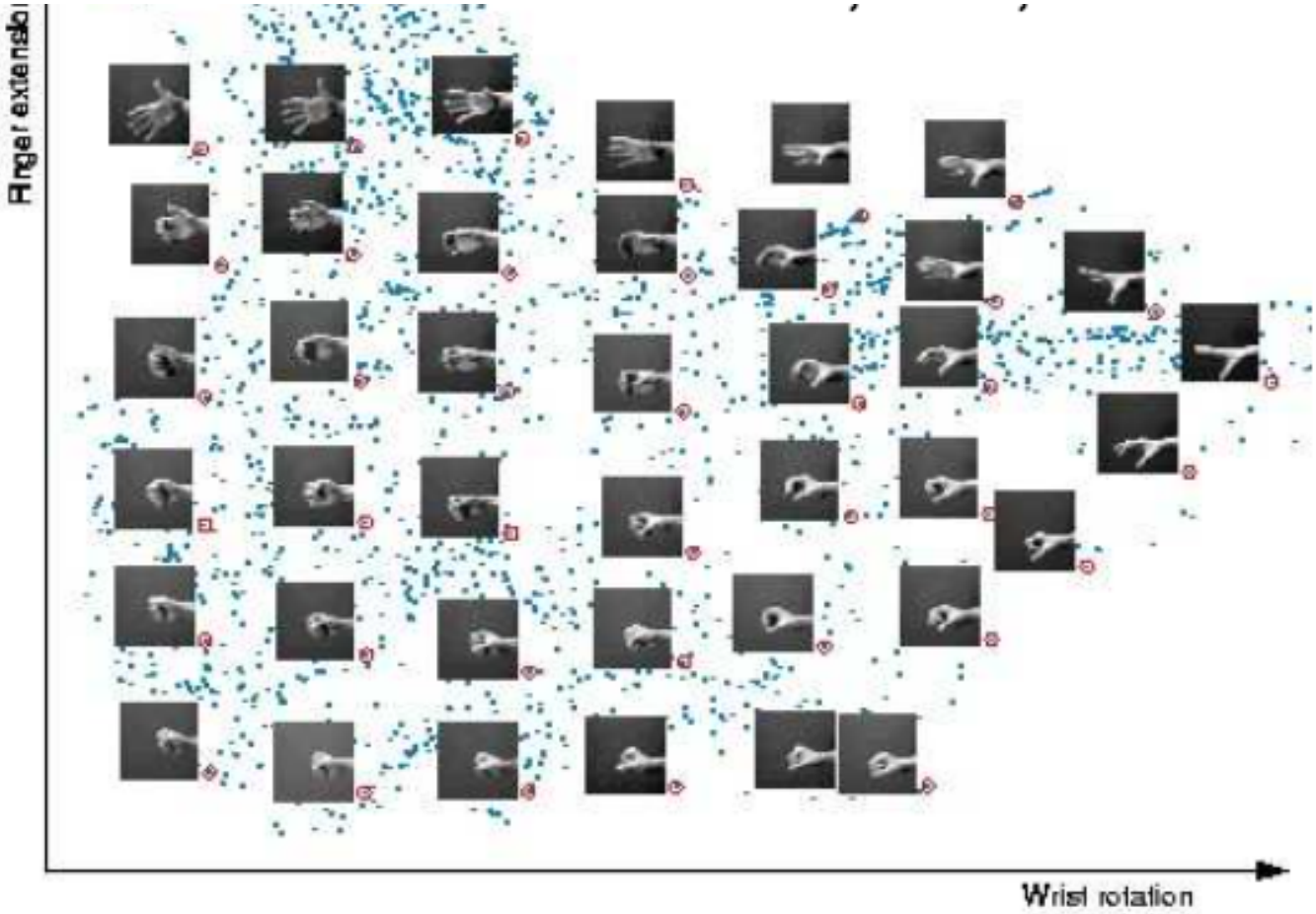
Lectura imprescindible: *A Global Geometric Framework for Nonlinear Dimensionality Reduction* de Joshua B. Tenenbaum, Vin de Silva, John C. Langford.

A Global Geometric Framework for Nonlinear Dimensionality Reduction de Joshua B. Tenenbaum, Vin de Silva, John C. Langford (2000).
- Dimensión: 4096-dimensional vectors, rep- resenting the brightness values of 64 pixel by 64 pixel images of a face rendered with different poses and lighting directions.
- $N = 698$,
- Isomap ($K = 6$) learns a three-dimensional embedding of the data's intrinsic geometric structure. A two-dimensional projection is shown, with a sample of the original input images (red circles) superimposed on all the data points (blue) and horizontal sliders (under the images) repre-senting the third dimension. The input-space distances given to Isomap were Euclidean distances between the 4096-dimensional image vectors.



A Global Geometric Framework for Nonlinear Dimensionality Reduction de Joshua B. Tenenbaum, Vin de Silva, John C. Langford (2000).
400-dimensional vectors Isomap applied to $N = 1000$ handwritten 2's from the MNIST database. The two most significant dimensions in the Isomap embedding, shown here, articulate the major features of the "2": bottom loop (x axis) and top arch (y axis). Input-space distances were measured by tangent distance, a metric designed to capture the invariances relevant in handwriting recognition (41). Here we used ϵ-Isomap (with $\epsilon = 4.2$)

A Global Geometric Framework for Nonlinear Dimensionality Reduction de Joshua B. Tenenbaum, Vin de Silva, John C. Langford (2000).



A Global Geometric Framework for Nonlinear Dimensionality Reduction de Joshua B. Tenenbaum, Vin de Silva, John C. Langford (2000).

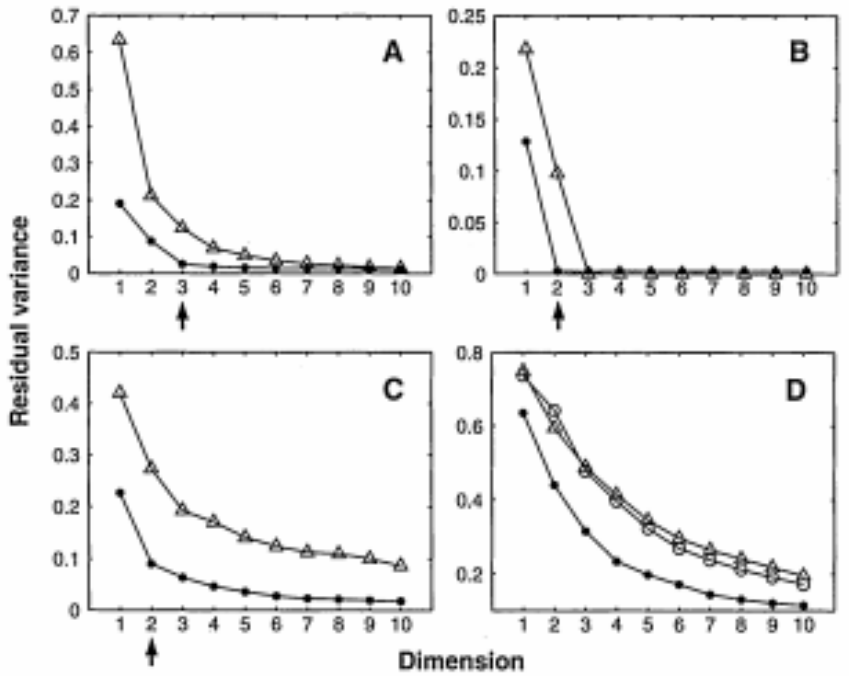


Fig. 2. The residual variance of PCA (open triangles), MDS [open triangles in (A) through (C); open circles in (D)], and Isomap (filled circles) on four data sets (*42*). (**A**) Face images varying in pose and illumination (Fig. 1A). (**B**) Swiss roll data (Fig. 3). (**C**) Hand images varying in finger extension and wrist rotation (*20*). (**D**) Handwritten "2"s (Fig. 1B). In all cases, residual variance decreases as the dimensionality *d* is increased. The intrinsic dimensionality of the data can be estimated by looking for the "elbow" at which this curve ceases to decrease significantly with added dimensions. Arrows mark the true or approximate dimensionality, when known. Note the tendency of PCA and MDS to overestimate the dimensionality, in contrast to Isomap.

Métodos basados en grafos

- Isomap (global approach): *A Global Geometric Framework for Nonlinear Dimensionality Reduction* de Joshua B. Tenenbaum, Vin de Silva, John C. Langford (2000).
- Existen otros:
  1. Locally Linear Embedding (LLE) (local approach): *Nonlinear Dimensionality Reduction by Locally Linear Embedding* de Sam T. Roweis and Lawrence K. Saul, 2000 `https://cs.nyu.edu/~roweis/lle/algorithm.html`
  2. Laplacian Eigenmaps : *Laplacian Eigenmaps for Dimensionality Reduction and Data Representation* de Belkin y Niyogi, 2003