# Classical Metric Multidimensional Scaling

Dave Dubin

September 2001

# Overview

These are slides based on a presentation at the University of North Texas, in August of 2001. The aim was to provide an overview of classical metric MDS, opening a discussion on how the process could be adapted for large data sets. I'm grateful to J. Douglas Carroll who provided the initial suggestions and pointers, and to Michael Trosset who provided code employing a better algorithm than the one I had proposed to adopt. Any errors or mischaracterizations are my responsibility.

# What is MDS?

Given a set of proximities $p_{ij}$ for $m$ objects, compute coordinates in $n$ dimensions for the objects, such that the Euclidean distances in the new coordinate space are a monotonic or linear function of the proximities.

A small example serves to illustrate: reconstruct a two-dimensional map of the US from a set of distances between various cities.

- Boston to Phila 268 miles
- Boston to Chicago 856
- Phila to Chicago 668
- Boston to Tampa 1182

- Chicago to Tampa 1001
- Phila to Tampa 931
- Boston to Dallas 1551
- Chicago to Dallas 798
- Phila to Dallas 1300
- Tampa to Dallas 915
- Boston to San Francisco 2708
- Phila to SF 2530
- Chicago to SF 1863

- Tampa to SF 2407
- Dallas to SF 1493

| | B | C | D | P | S | T |
|---|---|---|---|---|---|---|
| B | 0 | 856 | 1551 | 268 | 2708 | 1182 |
| C | 856 | 0 | 798 | 668 | 1863 | 1001 |
| D | 1551 | 798 | 0 | 1300 | 1493 | 915 |
| P | 268 | 668 | 1300 | 0 | 2530 | 931 |
| S | 2708 | 1863 | 1493 | 2530 | 0 | 2407 |
| T | 1182 | 1001 | 915 | 931 | 2407 | 0 |

# Scalar product vs. Euclidean distance

Recall that $L_2$ (conventional, straight line distance) generalizes the Pythagorean theorem:

$$d_{ij} = \left[ \sum_{k=1}^{n} \left( x_{ik} - x_{jk} \right)^2 \right]^{\frac{1}{2}}$$

$L_2$ is a relation between two vectors independent of any origin – you can translate a set of vectors in space without changing the distance between any two of them.

# Scalar product vs. Euclidean distance

The scalar product is a relation between three vectors (or between two with respect to some origin — it contains more information).

$$b_{ij} = \sum_{k=1}^{n} \left( x_{ik} \times x_{jk} \right)$$

The scalar product can be understood as a similarity measure between two object vectors: the more features they have in common (and the higher the scores on those common features) the higher the similarity value will be. But those inner product scores are not invariant under translation of the origin. That's why we say its a relation between three vectors.

# Application

Imagine an object by feature matrix. Multiply the matrix by its transpose and you'd get an object-object scalar product matrix. Classical MDS proceeds as follows:

1. Take our proximities as the target distances in the new space (after transformations).

2. Convert the object by object distances into scalar products, by selecting an origin and double-centering.

3. Factor the inner product matrix into two new object by dimension matrices.

# Step 1: transform proximities into distances

- If the proximities are to be distances, then they should be dissimilarities.
- Distances in a metric space respect metric axioms, and we'll have trouble later if these axioms are violated:

1. $d_{ij} > 0$
2. $d_{ii} = 0$
3. $d_{ij} = d_{ji}$
4. $d_{ik} \leq d_{ij} + d_{jk}$

Various transformations are possible (such as adding a constant).

Running Example: divide our city distances by ten, then square the results:

| | B | C | D | P | S | T |
|---|---|---|---|---|---|---|
| B | 0 | 7327 | 24056 | 718 | 73332 | 13971 |
| C | 7327 | 0 | 6368 | 4462 | 34707 | 10020 |
| D | 24056 | 6368 | 0 | 16900 | 22290 | 8372 |
| P | 718 | 4462 | 16900 | 0 | 64009 | 8667 |
| S | 73332 | 34707 | 22290 | 64009 | 0 | 57936 |
| T | 13971 | 10020 | 8372 | 8667 | 57936 | 0 |

# Step 2: Create a scalar product matrix

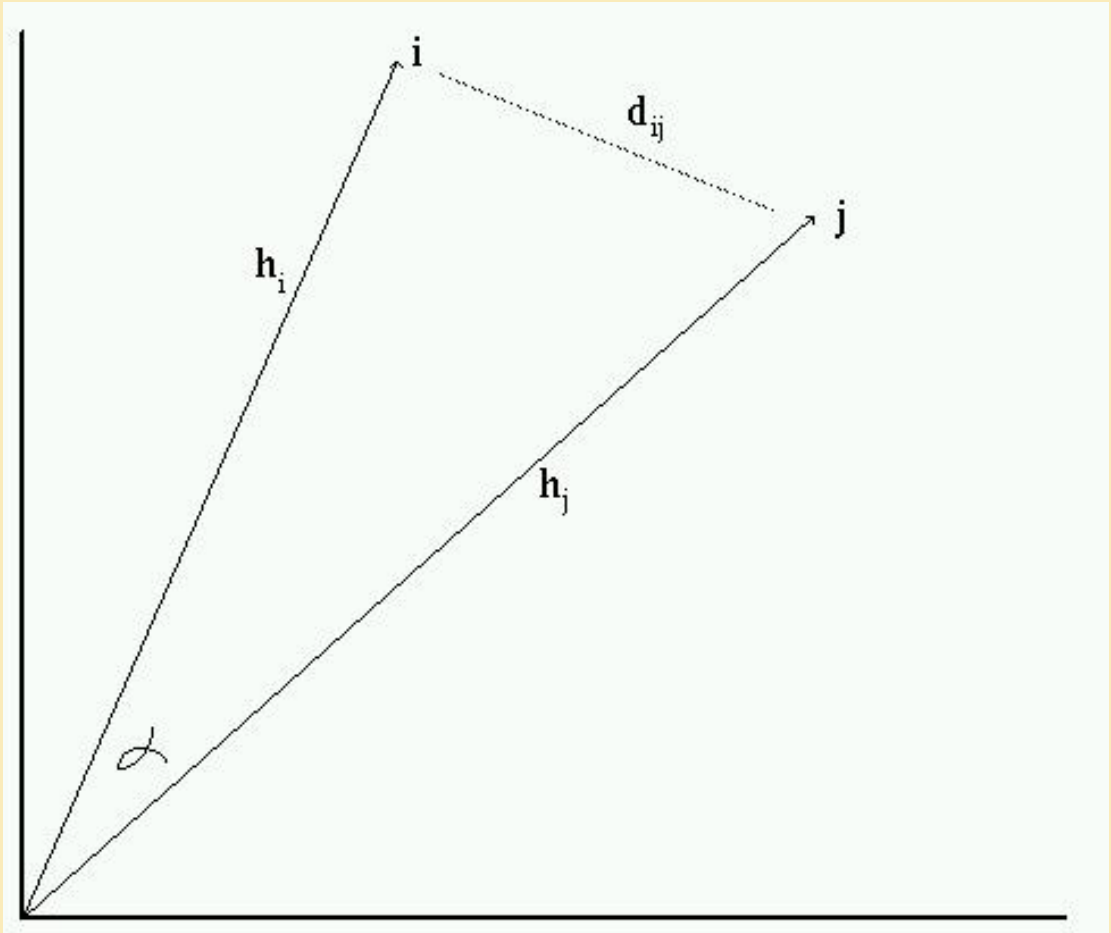
Any point might be selected as the origin of our space, but Torgerson recommends the centroid of our objects [4]. If we conjecture an underlying "true" metric structure, but suspect that our proximities will have random errors, then choosing the centroid of the space will tend to minimize the errors as they cancel each other out.

But deciding to make the centroid our origin doesn't tell us where the origin actually is, just where it is with respect to the other objects. Luckily, that's all the information we need. We're going to obtain our scalar product similarities by using the law of cosines that relates the distance between two vectors to their lengths and the cosine of the angle between them. In a

triangle with sides $a$, $b$, and $c$:

$$a^2 = b^2 + c^2 - 2bc\cos\alpha$$

where $\alpha$ is the angle between sides $b$ and $c$.



Recall that the cosine of the angle between two vectors is equal to the scalar product divided by the product of the vectors' $L_2$ lengths. We have $d_{ij}$ from our transformed proximities. Therefore we have all the information we need to compute the cosines of the angles between points in our solution (and thus the scalar products, too).

From Borg and Lingoes [2], chapters 16 and 18:

$$
\begin{aligned}
b_{ij} &= h_i h_j \cos\alpha \\
d_{ij}^2 &= d_{kj}^2 + d_{ki}^2 - 2 d_{kj} d_{ki} \cos\alpha \\
d_{kj} d_{ki} \cos\alpha &= \tfrac{1}{2}\left( d_{kj}^2 + d_{ki}^2 - d_{ij}^2 \right) \\
b_{ij} &= \tfrac{1}{2}\left( d_{kj}^2 + d_{ki}^2 - d_{ij}^2 \right) \\
(z_1, \ldots, z_m) &= \left( \tfrac{1}{n}\textstyle\sum_i^n x_{i1}, \ldots, \tfrac{1}{n}\sum_i^n x_{im} \right) \\
b_{ij} &= \textstyle\sum_a \left( x_{ia} - z_a \right)\left( x_{ja} - z_a \right) \\
b_{ij} &= \textstyle\sum_a x_{ia} x_{ja} - \sum_a x_{ia} z_a - \sum_a x_{ja} z_a + \sum_a z_a^2 \\
b_{ij} &= -\tfrac{1}{2}\left[ d_{ij}^2 - d_{\cdot j}^2 - d_{i\cdot}^2 + d_{\cdot\cdot}^2 \right]
\end{aligned}
$$

Where the point subscript mans that distances are averaged over their respective indices.

The last equation can be expressed in matrix form as:

$$B = -\frac{1}{2}\left(I - \frac{1}{n}U\right)D^{(2)}\left(I - \frac{1}{n}U\right) = -\frac{1}{2}ZD^{(2)}Z$$

Here $Z$ is just a shortcut for $I - \frac{1}{n}U$, $I$ is an identity matrix, $U$ is an $n$ by $n$ matrix consisting entirely of 1s only and $D^{(2)}$ is the matrix with elements $d_{ij}^2$. This is what is known as a "double centered" matrix.

From $I$ and $U$ we get the matrix $Z = I - \frac{1}{n}U$:

| | | | | | |
|---|---|---|---|---|---|
| 0.833 | -0.167 | -0.167 | -0.167 | -0.167 | -0.167 |
| -0.167 | 0.833 | -0.167 | -0.167 | -0.167 | -0.167 |
| -0.167 | -0.167 | 0.833 | -0.167 | -0.167 | -0.167 |
| -0.167 | -0.167 | -0.167 | 0.833 | -0.167 | -0.167 |
| -0.167 | -0.167 | -0.167 | -0.167 | 0.833 | -0.167 |
| -0.167 | -0.167 | -0.167 | -0.167 | -0.167 | 0.833 |

From $Z$ and the matrix of squared distances, we get the matrix of scalar products: $B = -\frac{1}{2} Z D^{(2)} Z$. But for rounding error, the columns and the rows should sum to zero:

| | | | | | |
|---|---|---|---|---|---|
| 10091 | 1718 | -5388 | 7678 | -15502 | 1403 |
| 1718 | 671 | -1254 | 1096 | -900 | -1332 |
| -5388 | -1254 | 3188 | -3864 | 6567 | 751 |
| 7678 | 1096 | -3864 | 5983 | -12894 | 2001 |
| -15502 | -900 | 6567 | -12894 | 32237 | -9507 |
| 1403 | -1332 | 751 | 2001 | -9507 | 6685 |

# Factoring $B$ into $XX'$

(Borg and Lingoes [2], chapter 17) From matrix algebra it can be shown that the decomposition of of a quadratic matrix A into the product $LDU$ is unique — there is exactly one lower triangular ($L$) and one upper triangular ($U$) matrix which, in combination with the diagonal matrix $D$, generate $A$.

Scalar products are symmetric. Thus $B = B'$.

$$B = LDU = B' = (LDU)'$$
$$(LDU)' = U'D'L'$$
$$LDU = U'D'L', L = U', U = L', D = D'$$

So for a symmetric matrix $S$, $S = LDL' = U'DU$. If we split $D$

into $D^{\frac{1}{2}}D^{\frac{1}{2}}$ then

$$\begin{aligned} B &= LDL' \\ &= LD^{\frac{1}{2}}D^{\frac{1}{2}}L' \\ &= \left(LD^{\frac{1}{2}}\right)\left(LD^{\frac{1}{2}}\right)' \\ &= XX' \end{aligned}$$

Note that if any negative values show up in $D$ then $D^{\frac{1}{2}}$ and $X$ will contain imaginary values.

Our goal of factoring $B$ into $LDL'$ is also called "orthogonal diagonalization" [3], and is always achievable if we're working with a square symmetric matrix. One factoring technique we can use[1] is called singular value decomposition (SVD). From

[1] Michael Trosset has shown we can take a more scalable approach using the spectral decomposition instead.

Berry, Dumais, and Letsche [1]:

Given an $m$ by $n$ matrix $A$ where, without loss of generality, $m \geq n$ and the rank of $A$ is $r$, the SVD of $A$ is defined as $A = U\Sigma V'$ where $U'U = V'V = I_n$ and $\Sigma$ is a diagonal matrix $(\sigma_1, \ldots \sigma_n)$, $\sigma_i > 0$ for $1 \leq i \leq r$, $\sigma_j = 0$ for $j = r + 1$. The first $r$ columns of the orthogonal matrices $U$ and $V$ define the orthonormal eigenvectors associated with the $r$ nonzero eigenvalues of $AA'$ and $A'A$, respectively.

As it turns out, the orthogonalization of a square, symmetric matrix is a special case of SVD (See Green and Carroll, chapter 5 [3]). So taking the SVD of $B$ will give us the solution we want.

Back to the running example: We factor our scalar products matrix $B$ via SVD, obtaining $B = TDT'$ Here's $T$:

| | | | | | |
|---|---|---|---|---|---|
| -0.413 | -0.419 | 0.509 | 0.378 | 0.291 | 0.408 |
| -0.036 | -0.256 | -0.546 | -0.435 | 0.529 | 0.408 |
| 0.187 | 0.399 | -0.394 | 0.694 | 0.054 | 0.408 |
| -0.336 | -0.207 | -0.244 | -0.108 | -0.779 | 0.408 |
| 0.800 | -0.228 | 0.318 | -0.132 | -0.151 | 0.408 |
| -0.202 | 0.711 | 0.356 | -0.397 | 0.056 | 0.408 |

And here's *D*, the diagonal matrix of eigenvalues:

| | | | | | |
|---|---|---|---|---|---|
| 49625 | 0 | 0 | 0 | 0 | 0 |
| 0 | 9231 | 0 | 0 | 0 | 0 |
| 0 | 0 | 46.365 | 0 | 0 | 0 |
| 0 | 0 | 0 | 36.25 | 0 | 0 |
| 0 | 0 | 0 | 0 | 9.9439 | 0 |
| 0 | 0 | 0 | 0 | 0 | 1.1749e-13 |

But we want to factor *B* into the product of two matrices (not 3) so we take the square roots of the values in *D*

| | | | | | |
|---|---|---|---|---|---|
| 222.77 | 0 | 0 | 0 | 0 | 0 |
| 0 | 96.08 | 0 | 0 | 0 | 0 |
| 0 | 0 | 6.81 | 0 | 0 | 0 |
| 0 | 0 | 0 | 6.02 | 0 | 0 |
| 0 | 0 | 0 | 0 | 3.15 | 0 |
| 0 | 0 | 0 | 0 | 0 | 3.4277e-07 |

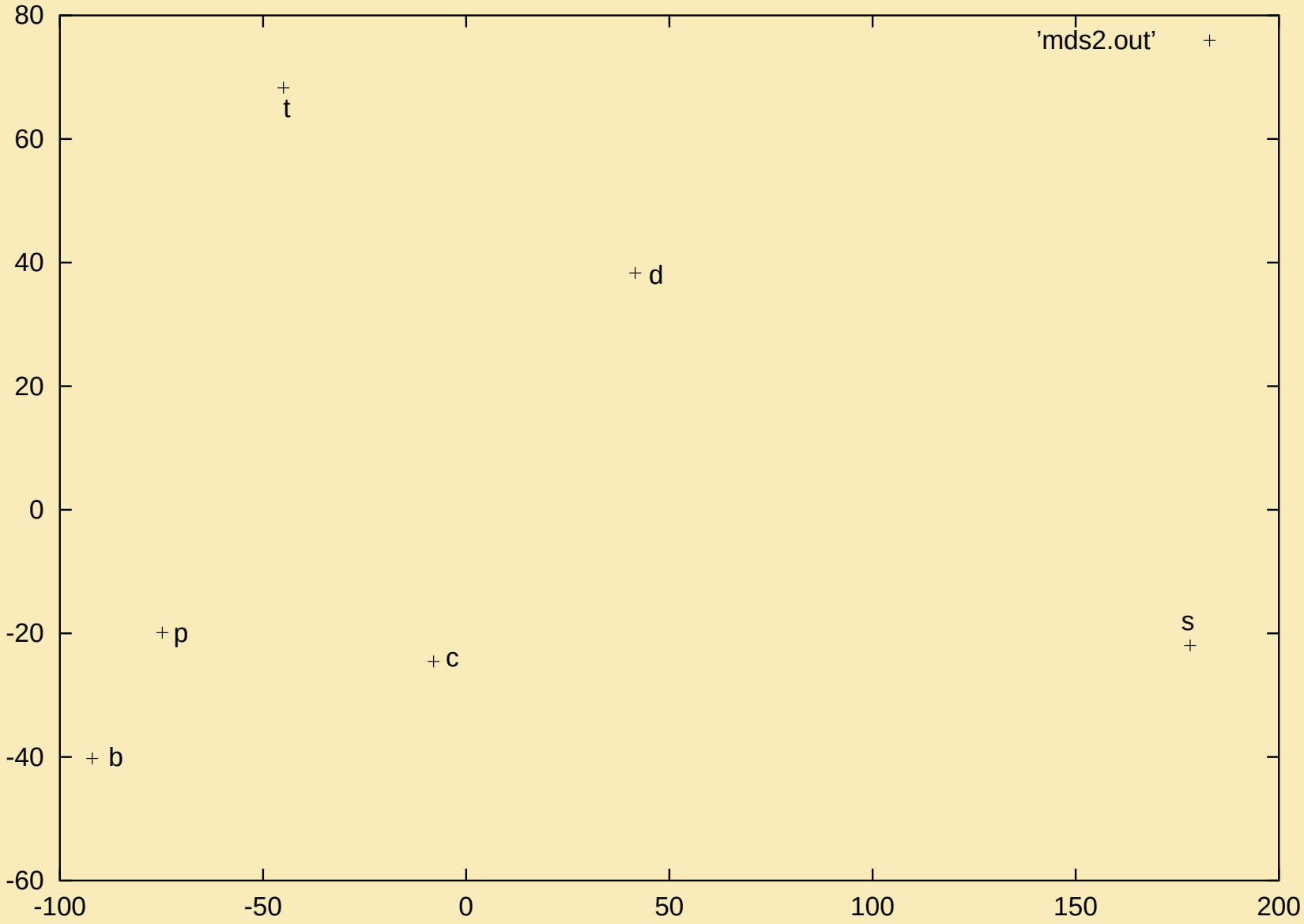
Finally, we multiply $T$ by $\sqrt{D}$ to get our solution $X$ and $B = XX'$

| | | | | | | |
|---|---|---|---|---|---|---|
| -92.009 | -40.247 | 3.4677 | 2.2772 | 0.91855 | 0 | b |
| -8.0208 | -24.555 | -3.7171 | -2.6196 | 1.6669 | 0 | c |
| 41.588 | 38.328 | -2.6827 | 4.1763 | 0.17144 | 0 | d |
| -74.781 | -19.872 | -1.6591 | -0.65215 | -2.4563 | 0 | p |
| 178.19 | -21.940 | 2.1653 | -0.79325 | -0.47701 | 0 | s |
| -44.967 | 68.286 | 2.4258 | -2.3886 | 0.17646 | 0 | t |

The last operation scales the coordinates by the "importance" of the derived dimensions as represented by the eigenvalues in *D*. Notice that there's a large amount of variability in the first two dimensions (SVD will always extract the most significant dimensions first). Coordinates and variability along the

third, fourth, and fifth dimensions are at least an order of magnitude smaller than the second dimension, and the sixth dimension is reduced to zero after rounding. This is what we'd expect, since our starting proximities are geographic distances between cities. If we introduced additional error (by using driving distances, for example) then the third and subsequent dimensions would probably be somewhat more significant.

Finally, here is a graph of the objects in the first two dimensions. They roughly correspond to east-west and north-south, respectively, although you have to rotate the figure 180 degrees to see how the cities would fall on a map of the US.


'mds2.out'
t
d
p
c
s
b
80
60
40
20
0
-20
-40
-60
-100
-50
0
50
100
150
200

# References

[1] BERRY, M., DUMAIS, S., AND LETSCHE, T. Computational methods for intelligent information access. In *Proceedings of Supercomputing'95* (San Diego, CA, December 1995), Association for Computing Machinery.

[2] BORG, I., AND LINGOES, J. *Multidimensional Similarity Structure Analysis*. Springer-Verlag, New York, 1987.

[3] GREEN, P. E., AND CARROLL, J. D. *Mathematical Tools for Applied Multivariate Analysis*. Academic Press, San Diego, CA, 1976.

[4] TORGERSON, W. S. *Theory and Methods of Scaling*. Wiley, New York, 1958.