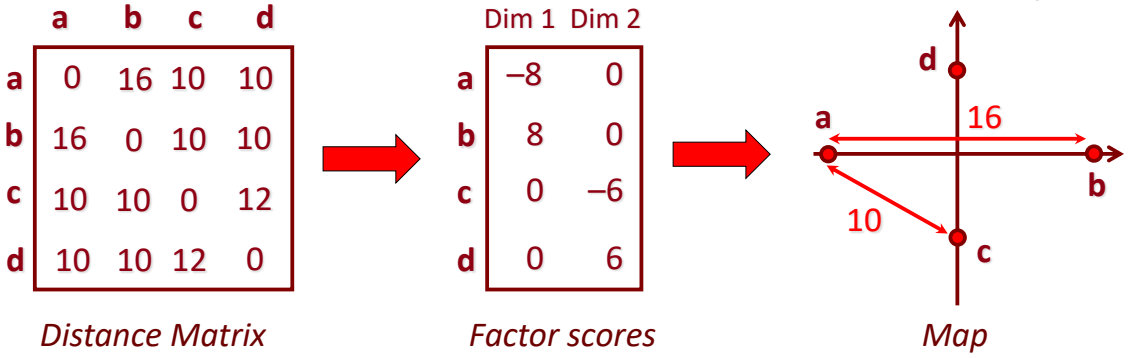
Multidimensional Scaling*

Hervé Abdi†

2022

MultiDimensional scaling (MDS)—also called Principal Coordinates Analysis—transforms a data table of distances (or dissimilarities) measured between a set of observations into a set of (Euclidean) coordinates for these observations that can, in turn, be used to plot the observations on a map, so that the (Euclidean) distances on the map between the observations best approximate their distances in the original distance matrix (see Figure 1 for a sketch of the method).

Figure 1: The Main Steps of Multidimensional Scaling: 1) Start with a distance matrix, 2) Transforms the distance matrix into a set of factor scores, and 3) Plot the observations using their factor scores. Here the distances on the map exactly recover the distances in the original matrix.



Definitions and Notations

Multidimensional scaling exists in two variations: *metric multidimensional scaling (MMDS)* to be used when the data are real distances (preferably Euclidean) and *non-metric multidimensional scaling (NMDS)* when the data are simply dissimilarities. But first, some definitions are needed to describe the different varieties of MDS and their properties.

*(2022) in B. Frey (Ed.) *The Sage encyclopedia of research design* (pp. 950–958). Sage.
†The University of Texas at Dallas. E-mail: herve@utdallas.edu.

Distances

A distance noted d associates to a pair of objects from a given set (which could be denoted $\{a, b, \dots\}$) a number such that

- $d(a, b) \geq 0$ (positivity)
- $d(a, a) = 0$
- $d(a, b) = d(b, a)$ (symmetry)
- $d(a, b) \leq d(a, c) + d(c, b)$ (triangle inequality).

When only the first three axioms hold (i.e., the triangle inequality does not hold) d is called a *dissimilarity*. Note that, these axioms can easily be fulfilled. For example, if two objects are the same, their distance can be defined as $d(a, a) = 0$, if they are different, their distances can be set to $d(a, b) = 1$: Using only these two numbers satisfies all four axioms of a distance.

Of course, we are mostly familiar with Euclidean distances: In addition to the four axioms of a distance, Euclidean distances exist in spaces for which the Pythagorean theorem holds. In these spaces, the objects to be considered can be seen as points in a space whose position is specified by their coordinates. These objects—called *vectors*—are denoted by bold face lower case letters (e.g., $\mathbf{a}$). For example, in Figure 1 the position of vector $\mathbf{a}$ is specified by its coordinates of -8 on Dimension 1 and 0 on Dimension 2. Formally, this vector $\mathbf{a}$ is written as a column of numbers as

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -8 \\ 0 \end{bmatrix} . \tag{1}$$

When transposed, the vector $\mathbf{a}$—now denoted $\mathbf{a}^\mathsf{T}$—becomes a row vector. For example

$$\mathbf{a}^\mathsf{T} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}^\mathsf{T} = \begin{bmatrix} a_1, & a_2 \end{bmatrix} = \begin{bmatrix} -8 \\ 0 \end{bmatrix}^\mathsf{T} = \begin{bmatrix} -8, & 0 \end{bmatrix} . \tag{2}$$

Euclidean distance

The squared Euclidean distance between two vectors is computed from the Pythagorean theorem applied to the coordinates of the vectors. For example, with $\mathbf{a}$ and $\mathbf{c}$ (see Figure 1) having coordinates:

$$\mathbf{a} = \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} -4 \\ 0 \end{bmatrix} \text{ and } \mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \begin{bmatrix} 0 \\ -3 \end{bmatrix} \tag{3}$$

the *squared* Euclidean distance $d(\mathbf{a}, \mathbf{c})$ is computed as

$$d^2(\mathbf{a}, \mathbf{c}) = (a_1 - c_1)^2 + (a_2 - c_2)^2 = (8 - 0)^2 + (0 - 6)^2 = 64 + 36 = 100 \tag{4}$$

\end{equation}

The Euclidean distance is defined for vectors with any number of dimensions. Suppose that $\mathbf{a}$ and $\mathbf{b}$ are two vectors with I elements, the Euclidean distance between these two vectors is computed as

$$d^2(\mathbf{a}, \mathbf{b}) = (\mathbf{a} - \mathbf{b})^\mathsf{T}(\mathbf{a} - \mathbf{b}) . \tag{5}$$

Matrices

Matrices generalize vectors and are defined as tables of numbers. Matrices are often named from their number of rows and columns and are denoted with uppercase bold face letters. For example, the 4×2 coordinates matrix $\mathbf{F}$ from Figure 1 is written as

$$\underset{4 \times 2}{\mathbf{F}} = \mathbf{F} = \begin{bmatrix} f_{1,1} & f_{1,2} \\ f_{2,1} & f_{2,2} \\ f_{3,1} & f_{3,2} \\ f_{4,1} & f_{4,2} \end{bmatrix} \begin{bmatrix} -8 & 0 \\ 8 & 0 \\ 0 & -6 \\ 0 & -6 \end{bmatrix}. \tag{6}$$

for more information on matrices see entry *Matrix Algebra.*

Metric multidimensional scaling (MDS)

Metric multidimensional scaling (MDS) analyzes data tables that store the distances between a set of observations. MDS represents these observations as points on a map that are positioned to best approximate their distances in the original data table. To do so, MDS transforms the original distance matrix into coordinates—akin to the factor scores of principal component analysis—which are then used to create these maps.

The basic idea of MMDS is to transform the original distance matrix into a cross-product matrix (i.e., a matrix akind to a covariance matrix) which is then decomposed with the eigen-decomposition method—-a process equivalent to principal component analysis (PCA). Just like PCA, MDS can be used with supplementary elements (also called illustrative or "out of sample" elements) which are elements that are not used in the original distance matrix but are projected onto the MMDS dimensions after these dimensions have been computed.

Metric Multidimensional Scaling: Eigen-analysis of a distance matrix

PCA is obtained by performing the eigen-decomposition of a matrix which can be 1) a correlation matrix (i.e., the variables to be analyzed are centered and normalized), 2) a covariance matrix (i.e., the variables are centered but not normalized), or 3) a cross-product matrix (i.e., the variables are neither centered nor normalized). A distance matrix cannot be analyzed directly using the eigen-decomposition (because distance matrices are not positive semi-definite matrices which are the only type of matrices that can be analyzed with PCA), but it can be transformed into an equivalent cross-product matrix which can then be analyzed with an eigen-decomposition.

Theory of MMDS: Transforming a distance matrix into a cross-product matrix

In order to transform a distance matrix into a cross-product matrix, we start from the fact that the scalar product between two vectors can easily be transformed into a distance (recall that the scalar product between vectors corresponds to a cross-product).

This distance can be rewritten in order to show the scalar product between vectors $\mathbf{a}$ and $\mathbf{b}$:

$$d^2(\mathbf{a},\mathbf{b}) = (\mathbf{a}-\mathbf{b})^\mathsf{T}(\mathbf{a}-\mathbf{b}) = \mathbf{a}^\mathsf{T}\mathbf{a} + \mathbf{b}^\mathsf{T}\mathbf{b} - 2 \times (\mathbf{a}^\mathsf{T}\mathbf{b}) \,, \tag{7}$$

where $\mathbf{a}^\mathsf{T}\mathbf{b}$ is the scalar product between $\mathbf{a}$ and $\mathbf{b}$.

Suppose now that we have a set of I observations (i.e., vectors) each described by J variables; and that these data are stored into an I by J data matrix denoted $\mathbf{X}$, the between observations cross product matrix (denoted $\mathbf{S}$) is then obtained as

$$\underset{I\times I}{\mathbf{S}} = \underset{I\times J}{\mathbf{X}}\underset{J\times I}{\mathbf{X}}^\mathsf{T} \,. \tag{8}$$

If we denote $\mathbf{s}$ the I by 1 vector of the diagonal elements of $\mathbf{S}$ and by $\mathbf{1}$ a vector of 1s, a between observation squared Euclidean distance matrix can be computed directly from the cross-product matrix as

$$\underset{I\times I}{\mathbf{D}} = \underset{I\times 1}{\mathbf{s}}\underset{1\times I}{\mathbf{1}}^\mathsf{T} + \underset{I\times 1}{\mathbf{1}}\underset{1\times I}{\mathbf{s}}^\mathsf{T} - 2\underset{I\times I}{\mathbf{S}} \,. \tag{9}$$

(Note that the elements of $\mathbf{D}$ gives the *squared* Euclidean distance between rows of $\mathbf{S}$)

Equation 9 shows that an Euclidean distance matrix can be computed from a cross-product matrix. To perform MMDS, Equation 9 needs to "reverted' ' in order to obtain a cross-product matrix from a distance matrix. But, there is one problem when implementing this idea, namely that *different* cross-product matrices can give the *same* distance. This can happen because distances are invariant by any change of origin. Therefore, in order to revert Equation 9, we need to impose an origin for the computation of the distance. An obvious choice is to choose the origin of the distance as the center of gravity of the dimensions. With this constraint, the cross-product matrix is obtained as follows.

First define a mass vector denoted $\mathbf{m}$ whose I elements give the mass of the I rows of matrix $\mathbf{D}$. These elements are all non-negative (i.e., positive or null) and their sum is equal to one:

$$\underset{1\times}{\mathbf{m}}^\mathsf{T}\underset{I\times 1}{\mathbf{1}} = 1 \,. \tag{10}$$

When all the rows have equal importance, each element is equal to $\frac{1}{I}$.

Second, define a $I \times I$ centering matrix denoted $\mathbf{\Xi}$ (read "big Xi' ') equal to

$$\underset{I\times I}{\mathbf{\Xi}} = \underset{I\times I}{\mathbf{I}} - \underset{I\times 1}{\mathbf{1}}\underset{1\times I}{\mathbf{m}}^\mathsf{T} \,. \tag{11}$$

Finally, the cross-product matrix is obtained from matrix $\mathbf{D}$ as:

$$\underset{I\times I}{\mathbf{S}} = -\tfrac{1}{2}\mathbf{\Xi}\mathbf{D}\mathbf{\Xi}^\mathsf{T} \,. \tag{12}$$

The eigen-decomposition of this matrix gives

$$\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\mathsf{T} \tag{13}$$

with

$$\mathbf{U}^{\mathsf{T}}\mathbf{U} = \mathbf{I} \text{ and } \mathbf{\Lambda} \text{ diagonal matrix of eigenvalues.} \tag{14}$$

The factor scores (i.e., the projection of the rows on the principal components of the analysis of $\mathbf{S}$) are obtained as

$$\mathbf{F} = \mathbf{U}\mathbf{\Lambda}^{\frac{1}{2}} \tag{15}$$

These scores have the properties that their cross-product (i.e., their sum of squares) is equal to the eigenvalues:

$$\mathbf{F}^{\mathsf{T}}\mathbf{F} = \mathbf{\Lambda}\,. \tag{16}$$

The map drawn from these factor scores gives now the best Euclidean approximation of the original distance matrix (or an exact Euclidean rendition, if the original data were a two dimensional Euclidean distance).

Also the (Euclidean) distance matrix can directly be build back from these factor scores because

$$\mathbf{F}\mathbf{F}^{\mathsf{T}} = \mathbf{S} \tag{17}$$

\end{equation} which in turn gives the original data matrix from Equation 9.

Note, incidentally that some implementations of MMDS prefer to compute factor scores whose *variance* (rather than their sums of squares) equals their eigenvalue. In this case, the factor scores—denoted $\widetilde{\mathbf{F}}$—are computed as

$$\widetilde{\mathbf{F}} = \mathbf{M}^{-\frac{1}{2}}\mathbf{U}\mathbf{\Lambda}^{\frac{1}{2}} \text{ (with } \mathbf{M} = \text{diag}\,\{\mathbf{m}\})\,. \tag{18}$$

The *variance* of these scores now equals the eigenvalues:

$$\widetilde{\mathbf{F}}^{\mathsf{T}}\mathbf{M}\widetilde{\mathbf{F}} = \mathbf{\Lambda}\,. \tag{19}$$

The distances read on the map are now *proportional* to the original distance.

A Toy Example

To illustrate the details of MMDS, we use the data from Figure 1). The distance data matrix is denoted $\mathbf{D}_{\text{Euclid}}$ and the matrix storing the Euclidean *squared* distance is denoted $\mathbf{D}$

$$\mathbf{D}_{\text{Euclid}} = \begin{bmatrix} 0 & 16 & 10 & 10 \\ 16 & 0 & 10 & 10 \\ 10 & 10 & 10 & 12 \\ 10 & 10 & 12 & 0 \end{bmatrix} \text{ and } \mathbf{D} = \begin{bmatrix} 0 & 256 & 100 & 100 \\ 256 & 0 & 100 & 100 \\ 100 & 100 & 100 & 144 \\ 100 & 100 & 144 & 0 \end{bmatrix}\,. \tag{20}$$

The elements of the mass vector $\mathbf{m}$ are all equal to $\frac{1}{4}$;

$$\mathbf{m}^{\mathsf{T}} = \begin{bmatrix} .25 & .25 & .25 & .25 \end{bmatrix}\,. \tag{21}$$

The centering matrix is equal to:

$$\underset{4\times 4}{\mathbf{\Xi}} = \begin{bmatrix} .75 & -.25 & -.25 & -.25 \\ -.25 & .75 & -.25 & -.25 \\ -.15 & -.25 & .75 & -.25 \\ -.25 & -.25 & -.25 & .75 \end{bmatrix}.$$

The cross product matrix is then equal to

$$\mathbf{S} = \begin{bmatrix} 64 & -64 & 0 & 0 \\ -64 & 64 & 0 & 0 \\ 0 & 0 & 36 & -36 \\ 0 & 0 & -36 & 36 \end{bmatrix}.$$

The eigen-decomposition of $\mathbf{S}$ gives

$$\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathsf{T}} \text{ with } \quad \mathbf{U} = \begin{bmatrix} -\frac{1}{\sqrt{2}} & 0 \\ \frac{1}{\sqrt{2}} & 0 \\ 0 & -\frac{1}{\sqrt{2}} \\ 0 & \frac{1}{\sqrt{2}} \end{bmatrix} \text{ and } \mathbf{\Lambda} = \begin{bmatrix} 128 & 0 \\ 0 & 72 \end{bmatrix}. \tag{22}$$

As in PCA, the eigenvalues are often transformed into percentage of explained variance (or inertia) in order to make their interpretation easier. Here, for example, we find that the first dimension "explains" 64% of the variance of the distances (i.e., $\frac{128}{128+72} = .64$).

We obtain the following matrix of scores (that can now be used to plot the Euclidean map of Figure 1)

$$\mathbf{F} = \mathbf{U}\mathbf{\Lambda}^{\frac{1}{2}} = \begin{bmatrix} -\sqrt{\frac{128}{2}} & 0 \\ \sqrt{\frac{128}{2}} & 0 \\ 0 & -\sqrt{\frac{72}{2}} \\ 0 & \sqrt{\frac{72}{2}} \end{bmatrix} . = \begin{bmatrix} -\sqrt{64} & 0 \\ \sqrt{64} & 0 \\ 0 & -\sqrt{36} \\ 0 & \sqrt{36} \end{bmatrix} . = \begin{bmatrix} -8 & 0 \\ 8 & 0 \\ 0 & -6 \\ 0 & 6 \end{bmatrix}. \tag{23}$$

A more realistic example

This realistic example is derived from research reported in an article published by Jordi Ballester and collaborators in 2009. In this work the authors asked 26 wine experts to evaluate a set of 18 wines that comprised six red, six white, and six rosé wines. These participants—who could not see the color of the wines (the wines were presented in dark glasses under red light)—were asked to smell the wines and to sort them in three groups. From this task (called the *sorting* task) a distance can be defined between two wines by counting the number of times these two wines were *not* grouped together. Table 1 shows the distance matrix obtained from the sorting task performed by the wine experts.

Table 1: Sorting distance for 26 wine experts. Numbers in the table give the number of times two wines were not sorted together. Wines R1 to R6 are red wines, P1 to P6 are rosés wines, and W1 to W6 are white wines.

	P1	P2	P3	P4	P5	P6	R1	R2	R3	R4	R5	R6	W1	W2	W3	W4	W5	W6
P1	0	17	13	17	18	16	17	19	22	20	20	19	17	18	18	20	18	17
P2	17	0	14	15	18	16	24	21	22	20	20	24	17	15	13	14	16	19
P3	13	14	0	17	18	15	24	21	25	22	22	23	13	18	11	15	15	18
P4	17	15	17	0	15	15	19	17	20	21	19	20	20	19	18	24	24	20
P5	18	18	18	15	0	11	22	21	20	23	24	22	18	17	16	19	16	15
P6	16	16	15	15	11	0	25	19	25	25	25	24	18	16	11	16	17	15
R1	17	24	24	19	22	25	0	9	6	6	5	4	24	19	24	23	22	24
R2	19	21	21	17	21	19	9	0	9	10	11	7	21	23	23	22	22	23
R3	22	22	25	20	20	25	6	9	0	5	6	6	22	21	25	22	22	22
R4	20	20	22	21	23	25	6	10	5	0	5	4	22	21	25	21	23	23
R5	20	20	22	19	24	25	5	11	6	5	0	5	23	20	24	21	21	24
R6	19	24	23	20	22	24	4	7	6	4	5	0	23	21	24	22	23	23
W1	17	17	13	20	18	18	24	21	22	22	23	23	0	18	12	12	13	11
W2	18	15	18	19	17	16	19	23	21	21	20	21	18	0	11	12	12	14
W3	18	13	11	18	16	11	24	23	25	25	24	24	12	11	0	12	12	14
W4	20	14	15	24	19	16	23	22	22	21	21	22	12	12	12	0	10	13
W5	18	16	15	24	16	17	22	22	22	23	21	23	13	12	12	10	0	16
W6	17	19	18	20	15	15	24	23	22	23	24	23	11	14	14	13	16	0

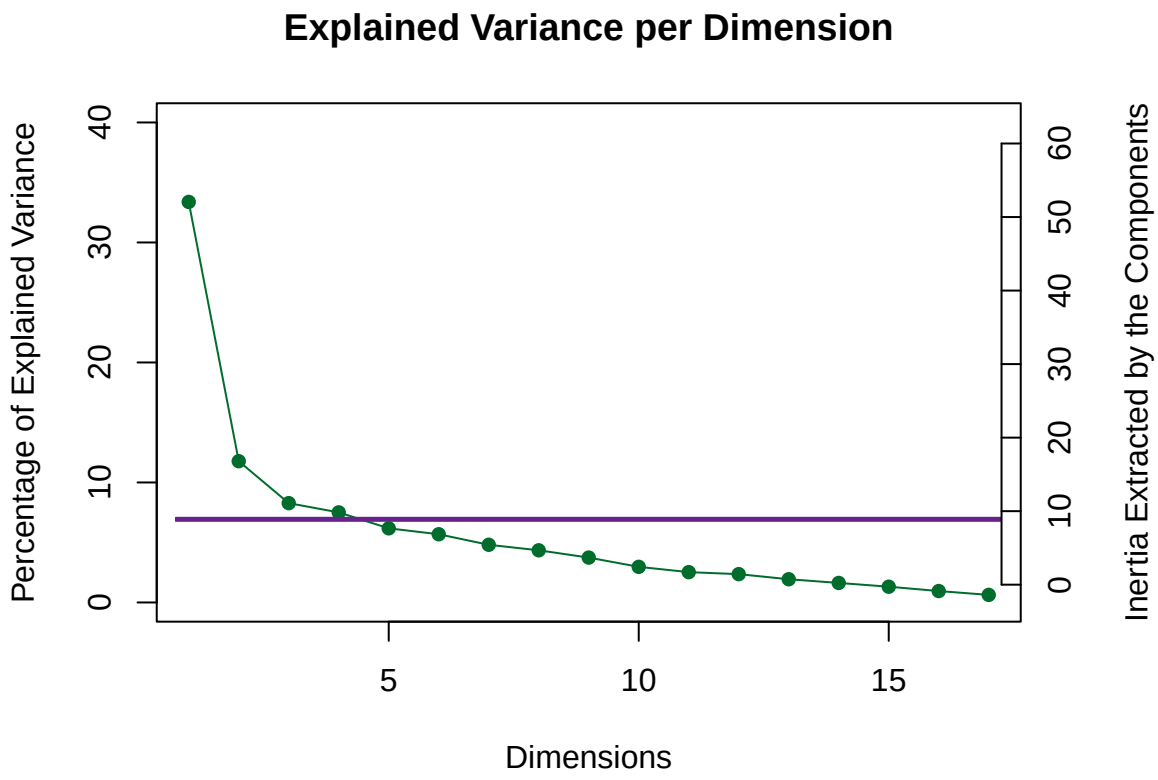


Figure 2: Metric MDS. Experts. Eigenvalues Scree Plot with Kaiser Line.

Figure 2 shows the scree plot of the eigenvalues. The horizontal line in the graph is the *Kaiser* line that separates the eigenvalues larger than average (considered as reliable and warranting further investigation) from the eigenvalues smaller than average (that could be ignored at first approximation). The Kaiser line suggests that up to four dimensions could be relevant but also that the first two dimensions explain most of the variance and therefore deserves close inspection. Figure 3 shows the map obtained from these first two dimensions that together explained 45% (i.e., 33 + 12) of the variance of the data. Dimension 1 isolates the red from the rosé and white wines. Dimension 2 separates white from rosés wines. This pattern of results strongly suggests that Experts do perceive wine of different colors as being different wines even when the wines are tested blindly.

Comparing two distance matrices: supplementary projections

The authors also asked a group of 19 novices to sort the same wines. These novices were asked to smell the wines and to sort them in as many groups they wanted (with the restriction that they needed to make more than one group but less than 18 groups). The results of this task are given in Table 2.

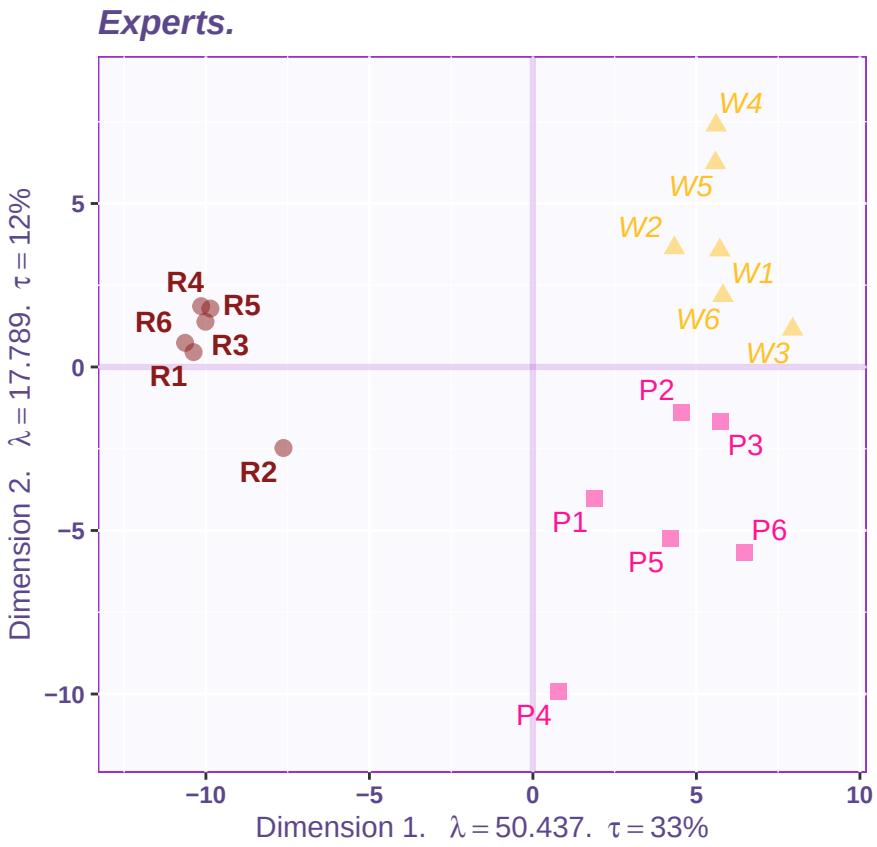


Figure 3: Metric MDS. Experts. Wines R1 to R6 are red wines, P1 to P6 are rosés wines, and W1 to W6 are white wines.

Table 2: Sorting distance for 19 wine novices. Numbers in this table give the number of times two wines were not sorted together. Wines R1 to R6 are red wines, P1 to P6 are rosés wines, and W1 to W6 are white wines.

	P1	P2	P3	P4	P5	P6	R1	R2	R3	R4	R5	R6	W1	W2	W3	W4	W5	W6
P1	0	13	14	16	15	14	16	18	16	17	16	19	16	12	15	16	17	15
P2	13	0	7	10	12	10	16	17	18	19	19	18	10	13	9	12	13	13
P3	14	7	0	12	11	15	15	17	17	19	19	17	13	15	12	12	13	13
P4	16	10	12	0	5	7	19	19	19	19	19	15	10	14	7	12	11	10
P5	15	12	11	5	0	9	18	16	19	18	19	15	13	13	9	13	11	11
P6	14	10	15	7	9	0	18	18	19	18	19	17	11	13	11	13	10	9
R1	16	16	15	19	18	18	0	12	12	14	13	14	19	19	18	16	16	18
R2	18	17	17	19	16	18	12	0	15	14	12	11	19	17	16	16	14	19
R3	16	18	17	19	19	19	12	15	0	14	12	13	17	16	18	17	17	18
R4	17	19	19	19	18	18	14	14	14	0	12	13	18	17	19	16	18	16
R5	16	19	19	19	19	19	13	12	12	12	0	7	17	17	19	17	18	18
R6	19	18	17	15	15	17	14	11	13	13	7	0	17	17	19	16	17	19
W1	16	10	13	10	13	11	19	19	17	18	17	17	0	11	10	10	10	9
W2	12	13	15	14	13	13	19	17	16	17	17	17	11	0	13	15	13	13
W3	15	9	12	7	9	11	18	16	18	19	19	19	10	13	0	10	8	10
W4	16	12	12	12	13	13	16	16	17	16	17	16	10	15	10	0	12	11
W5	17	13	13	11	11	10	16	14	17	18	18	17	10	13	8	12	0	10
W6	15	13	13	10	11	9	18	19	18	16	18	19	9	13	10	11	10	0

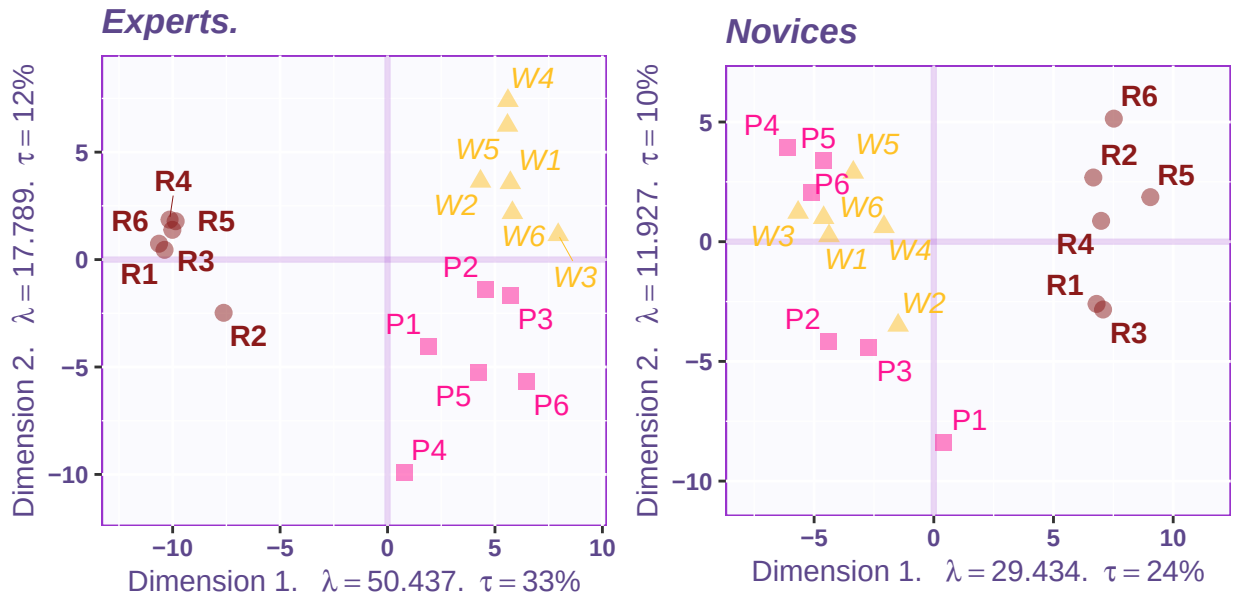


Figure 4: Experts (left) and Novices (right).

An MMDS performed on these results (see Figure 4, right panel) suggests that the novices perform in a way roughly similar to the experts, but a precise comparison is difficult to implement first because the scales of the two graphs are not the same but also because the directions of the dimensions seem reversed (e.g., experts have their red wines on the left side of Dimension 1, novices have them on the right side). Because the experts here play the role of a reference, we need to find an optimal transformation that can fit the space of the novices into the space of the experts. There are,in fact, several methods—collectively called *Procrustes* or *Procrustean*—that can perform this task. Here we present one of these methods called *projection as supplementary elements*.

Multidimensional scaling: Supplementary elements

After we have computed the MDS solution, it is possible to project supplementary or illustrative elements (also called out of sample elements) onto this solution. To illustrate this procedure, we will project the distance matrix from Table 2 obtained from the sorting task performed by the novice wine tasters onto the space defined by the analysis of the sorting task by the expert wine tasters.

The number of supplementary elements is denoted by I_{sup}. For each supplementary elements, we need the values of its distances to all the I active elements. We can store these distance in an $I \times I_{\text{sup}}$ supplementary distance matrix denoted $\widetilde{\mathbf{D}}_{\text{sup}}$. The first step is to norm matrix $\widetilde{\mathbf{D}}_{\text{sup}}$ in order to make commensurable with the active distance matrix $\mathbf{D}$. This step is necessary here, because the maximum distance for a sorting distance matrix is equal to the number of participants: Here for $\mathbf{D}$ the largest possible distance is 26 (i.e., the number of Experts) whereas for $\widetilde{\mathbf{D}}_{\text{sup}}$ it is equal to 19. So, $\widetilde{\mathbf{D}}_{\text{sup}}$ is rescaled to get

$$\mathbf{D}_{\text{sup}} = \widetilde{\mathbf{D}}_{\text{sup}} \frac{26}{19} . \tag{24}$$

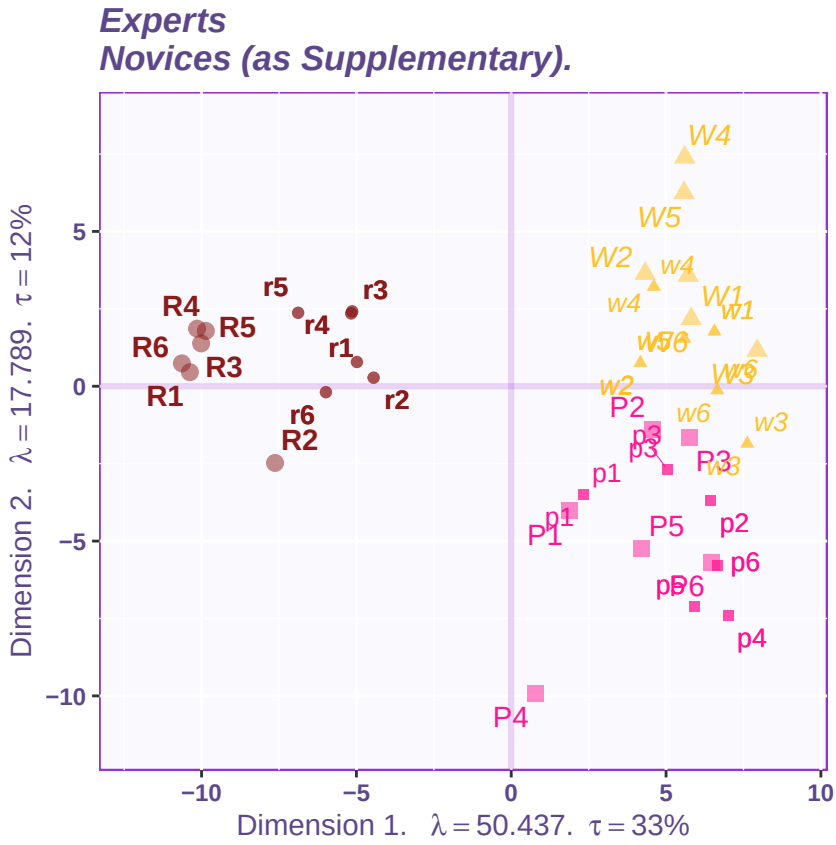


Figure 5: Metric MDS. Experts MMDS solution with Novices Projected as Supplementary Elements. Wines R1 to R6 are red wines, P1 to P6 are rosés wines, and W1 to W6 are white wines. Novices Wines are Labeled with Lower Case Letter

After normalization of the supplementary distance matrix, the next step is to transform $\mathbf{D}_{\text{sup}}$ into a cross-product matrix denoted $\mathbf{S}_{\text{sup}}$ This is done by first computing the difference for each supplementary column and the average distance vector and then centering the rows with the same centering matrix that was used previously to transform the distance of the active elements. Specifically, the supplementary cross-product matrix is obtained as:

$$\mathbf{S}_{\text{sup}} = -\tfrac{1}{2}\mathbf{\Xi}\left(\mathbf{D}_{\text{sup}} - \mathbf{D}\mathbf{m}\mathbf{1}\right)^{\mathsf{T}} \ . \tag{25}$$

(where $\mathbf{1}$ is an I_{sup} by 1 vector of ones; note, also, that when $\mathbf{D}_{\text{sup}} = \mathbf{D}$, Equation 25 reduces to Equation 11).

The next step is to project the matrix $\mathbf{S}_{\text{sup}}$ onto the space defined by the analysis of the active distance matrix. We denote by $\mathbf{F}_{\text{sup}}$ the matrix of projection of the supplementary elements. Its computational formula is obtained by first combining Equations 15 and 13 in order to get

$$\mathbf{F} = \mathbf{S}^{\mathsf{T}}\mathbf{M}^{-\frac{1}{2}}\mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}} \ , \tag{26}$$

and then substituting $\mathbf{S}_{\text{sup}}$ for $\mathbf{S}$ and simplifying gives

$$\mathbf{F}_{\text{sup}} = \mathbf{S}_{\text{sup}}^{\mathsf{T}}\mathbf{M}^{-\frac{1}{2}}\mathbf{U}\mathbf{\Lambda}^{-\frac{1}{2}} = \mathbf{S}_{\text{sup}}^{\mathsf{T}}\mathbf{F}\mathbf{\Lambda}^{-1} \ . \tag{27}$$

Figure 5 displays the projection of the novices as supplementary elements in the space defined by the experts.

Analyzing non-metric data

Metric MDS is adequate only when dealing with with real (preferably Euclidean) distances (see Togerson, 1958). In order to accommodate weaker measurements such as dissimilarities non-metric MDS is adequate. Non-Metric MDS seeks to approximate the original data by an Euclidean distance matrix that preserves the order of the original data (Shepard, 1966; for a recent thorough review, see Borg & Groenen, 2018). In this case, the *order* of the Euclidean distances on the map best approximates the *order* of the distances in the original data.

Non-metric MDS (developped by Shepard, 1966, and Kruskal, 1964) uses iterative computational approximations (mostly gradient descent) that, however, are not always guaranteed to converge. In addition, these methods need to fix a priori the dimensionality of the solution and could give different solutions for different dimensionality. Despite these potential problems, these methods provide a useful tool to graphically represent dissimilarity data. As an illustration, the sorting data of the Experts were used to perform NNMDS. The results, shown in Figure 6, give a configuration very similar to the metric-MDS (which is optimal for Euclidean distances such as the sorting task distances) even though the original distances are not exactly recovered.

See Also

Barycentric discriminant analysis, Matrix algebra, Principal component analysis.

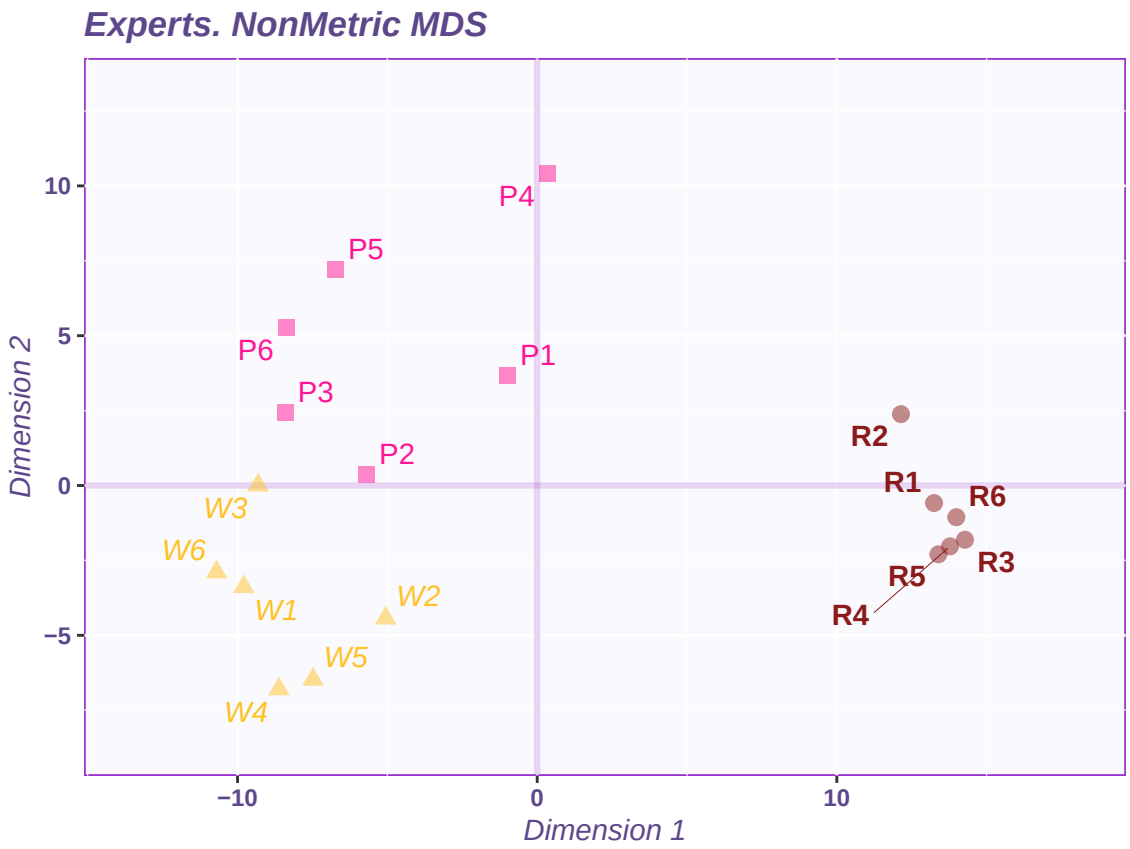


Figure 6: Non-Metric MDS: Experts. Wines R1 to R6 are red wines, P1 to P6 are rosés wines, and W1 to W6 are white wines.

Further Readings

[1] Abdi, H., & Beaton, D. (2023) *Principal component and correspondence analyses using R*. New York: Springer Verlag.

[2] Borg I., Groenen, P., & Mair, P. (2018). *Modern multidimensional scaling.* New York: Springer Verlag.

[3] Gower, J.C., & Dijksterhuis, G.B. (2004). *Procrustes problems*. Oxford: Oxford University Press.

[4] Kruskal, J.B. (1964). Nonmetric multidimensional scaling: a numerical method. *Psychometrika, 29*, 115–129.

[5] Pelé, J., Bécu, J.M., Abdi, H., & Chabbert, M. (2012). Bios2mds: an R package for comparing orthologous protein families by metric multidimensional scaling. *BMC Bioinformatics, 13,* 133–140. doi:10.1186/1471-2105-13-133.

[6] Pelé, J., Abdi, H., Moreau, M., Thybert, D., & Chabbert, M. (2011). Multidimensional scaling reveals the main evolutionary pathways of class A G-protein-coupled receptors. *PLoS One , 6*, 1–10, e19094. doi:10.1371/journal.pone.0019094.

[7] Shepard, R.N. (1966). Metric structure in ordinal data. *Journal of Mathematical Psychology, 3*, 287–315.

[8] Togerson, W.S. (1958). *Theory and methods of scaling*. New York: Wiley.