Weighted Euclidean Biplots

Michael Greenacre
Patrick J.F. Groenen

July 2013

Barcelona GSE Working Paper Series

Working Paper nº 708

Weighted Euclidean Biplots

Michael Greenacre
Department of Economics and Business
Universitat Pompeu Fabra
Ramon Trias Fargas, 25-27
08005 Barcelona
SPAIN
E-mail: michael.greenacre@upf.edu

Patrick J.F. Groenen
Econometric Institute
Erasmus School of Economics
Erasmus University
Rotterdam
THE NETHERLANDS
groenen@ese.eur.nl

Abstract: We construct a weighted Euclidean distance that approximates any distance or dissimilarity measure between individuals that is based on a rectangular cases-by-variables data matrix. In contrast to regular multidimensional scaling methods for dissimilarity data, the method leads to biplots of individuals and variables while preserving all the good properties of dimension-reduction methods that are based on the singular-value decomposition. The main benefits are the decomposition of variance into components along principal axes, which provide the numerical diagnostics known as contributions, and the estimation of nonnegative weights for each variable. The idea is inspired by the distance functions used in correspondence analysis and in principal component analysis of standardized data, where the normalizations inherent in the distances can be considered as differential weighting of the variables. In weighted Euclidean biplots we allow these weights to be unknown parameters, which are estimated from the data to maximize the fit to the chosen distances or dissimilarities. These weights are estimated using a majorization algorithm. Once this extra weight-estimation step is accomplished, the procedure follows the classical path in decomposing the matrix and displaying its rows and columns in biplots.

Keywords: biplot, correspondence analysis, distance, majorization, multidimensional scaling, singular-value decomposition, weighted least squares.

JEL codes: C19, C88

1. Introduction

We are concerned here with biplots of rectangular data matrices (Gabriel 1971, Gower and Hand 1996, Greenacre 2010, Gower, Lubbe and Le Roux 2011). A biplot is a graphical representation of the rows (usually cases) and columns (usually variables) of a matrix, where typically a distance approximation is achieved with respect to the cases, depicted by points, while the variables are represented by arrows defining biplot axes onto which cases are projected, yielding approximations of the original data values. In standard applications of the biplot, a Euclidean distance is assumed between the cases, usually incorporating some form of pre-standardization of the variables. This biplot, which we could call the "regular biplot", is easy to understand and to interpret, is an optimal least-squares representation of the data in a low-dimensional space and has convenient properties such as the decomposition of the total variance of the matrix into contributions by all the elements of the matrix along each dimension of the solution and thus by each row and each column as well.

Our particular interest here is in a more general class of distance or dissimilarity measures defined on the cases, which does not fit into the regular biplot approach. At present there are two approaches, one linear and the other nonlinear, to deal with this situation where proximities between cases are preferred to be defined in a non-standard way. Both are two-step approaches where an initial configuration of the cases is obtained by multidimensional scaling, or some other nonlinear mapping, of the proximity matrix, and the second step is conditional on the first one. The linear approach imitates the regular biplot by simply adding the variables as vectors to this configuration, using as coordinates the estimated coefficients of a linear regression of each variable on the dimensions of the configuration (see, for example, Groenen and Borg 2005: Chapter 4; Greenacre 2010: Chapter 4). The more complicated nonlinear approach adds curved trajectories by circle projection or using differentials to achieve so-called nonlinear biplots (Gower and Hand 1996, Gower et al. 2011). We propose a simple but elegant alternative approach, which stays within the regular biplot framework, while allowing any proximity measure to be used between the cases. This is achieved by estimating weights for the variables and using a weighted Euclidean distance function between the cases, an idea inspired by the normalization commonly used in principal component analysis and correspondence analysis. We thus call this approach a weighted Euclidean biplot.

In principal component analysis (PCA) of a cases-by-variables data matrix **X**, where variables are standardized, the distances between rows are given by standardized Euclidean distances according to the following definition in squared distance form:

$$d^2(\mathbf{x}_i,\mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\mathsf{T} \mathbf{D}_s^{-2} (\mathbf{x}_i - \mathbf{x}_j) = \sum_k (x_{ik} - x_{jk})^2 / s_k^2 \qquad (1)$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are vectors denoting the i-th and j-th rows of **X** and $\mathbf{D}_s$ is the diagonal matrix of standard deviations s_k. The standardizing factors $1/s_k^2$ can be considered as squared weights assigned to the respective variables in the calculation of the distances between rows. Similarly, in correspondence analysis (CA) of a table of frequencies, the inherent chi-square distance has the same form, but the (squared) weights are proportional to the inverses of the corresponding margins of the table (see, for example, Greenacre 2007).

These distance functions can be subsumed in the general case of a weighted Euclidean (squared) distance:

$$d^2(\mathbf{x}_i,\mathbf{x}_j) = (\mathbf{x}_i - \mathbf{x}_j)^\mathsf{T} \mathbf{D}_w (\mathbf{x}_i - \mathbf{x}_j) = \sum_k w_k^2 (x_{ik} - x_{jk})^2 \qquad (2)$$

where $\mathbf{D}_w$ is a diagonal matrix of squared weights w_k^2, $k=1,\ldots,p$, for the p variables, serving to balance out, in some sense, the contributions of the variables to the distances between cases.

In several areas of research, the practitioner is more interested in distance measures which are not of the above form and often non-Euclidean, for example the Bray-Curtis dissimilarity measure in ecology – see Gower and Legendre (1986) and Legendre and Legendre (1998) for a repertory of such distances.

The present paper aims to approximate the distances or dissimilarities chosen by the user, whatever their definition, by a weighted Euclidean distance of the form (2). Weights will be estimated for the variables, and these weights can then be interpreted as those that are inherently assigned to the variables by the chosen distance function. We can then follow the regular biplot approach using weighted least-squares approximation of the matrix, which has the following advantages:

1. The framework of the singular value decomposition, visualizing the cases (rows) and variables (columns) in a joint plot, with a straightforward interpretation in terms of distances and scalar products;
2. The convenient breakdown of variance across principal axes of both the rows and columns, which provides useful numerical diagnostics in the interpretation and evaluation of the results.

In Section 2 we shall summarize the classical biplot framework with weights on the variables. Then in Section 3 we describe an algorithmic approach to estimate the weights, with specific details given in Appendix 1. In Section 4 we give an example of this approach and conclude with a discussion in Section 5.

2. Weighted Euclidean biplots

Our main interest is in weighting the variables in the definition of distances between the individuals, or cases, usually the rows of the data matrix. Since cases themselves can also be weighted to differentiate their influence on the solution, which serves a different purpose, we shall use the terms *mass* for a case weight and *weight* for a variable weight. Notice that in correspondence analysis the term “mass” is used for both rows and columns, where they play the dual roles of masses and weights in the present sense.

Suppose that we have a data matrix $\mathbf{Y}$ ($n \times m$), usually pre-centered with respect to rows or columns or both. Let $\mathbf{D}_r$ ($n \times n$) and $\mathbf{D}_w$ ($m \times m$) be diagonal matrices of row (case) masses and column (variable) weights respectively. The masses and weights are all non-nonegative and, without loss of generality, the row masses have a sum of 1. The rows of $\mathbf{Y}$ are presumed to be points in an m-dimensional Euclidean space, structured by the scalar product and metric defined by the weight matrix $\mathbf{D}_w$. The solution, a low-dimensional subspace that fits the points as closely as possible, is established by weighted least-squares, where each point is weighted by its mass. The following function is thus minimized:

$$\mathrm{In}(\mathbf{Y} - \hat{\mathbf{Y}}) = \sum_i r_i (\mathbf{y}_i - \hat{\mathbf{y}}_i)^{\mathsf{T}} \mathbf{D}_w (\mathbf{y}_i - \hat{\mathbf{y}}_i) = \mathrm{trace}[\mathbf{D}_r (\mathbf{Y} - \hat{\mathbf{Y}}) \mathbf{D}_w (\mathbf{Y} - \hat{\mathbf{Y}})^{\mathsf{T}}] \qquad (3)$$

where $\hat{\mathbf{y}}_i$, the i-th row of $\hat{\mathbf{Y}}$, is the closest low-dimensional approximation of $\mathbf{y}_i$. The function In(…) stands for the *inertia*, in this case the inertia of the difference between the original and approximated matrices. The *total inertia*, which is being decomposed or “explained” by the solution, is equal to In($\mathbf{Y}$).

As is well-known (see, for example, Greenacre, 1984, Appendix), the solution can be obtained neatly using the generalized singular value decomposition (SVD) of the matrix $\mathbf{Y}$. Computationally, using the regular SVD, the steps in finding the solution are to first pre-process the matrix $\mathbf{Y}$ by pre- and post-multiplying by the square roots of the weighting matrices, then to calculate the SVD and then post-process the solution using the inverse transformation, leading to so-called principal coordinates, principal axes, standard coordinates and contribution coordinates. The steps are summarized as follows:

1. $\mathbf{S} = \mathbf{D}_r^{1/2} \mathbf{Y} \mathbf{D}_w^{1/2}$ (4)
2. $\mathbf{S} = \mathbf{U} \mathbf{D}_\alpha \mathbf{V}^{\mathsf{T}}$ (the SVD), (5)

where the left and right singular vectors in the columns **U** and **V** satisfy $\mathbf{U}^\mathsf{T}\mathbf{U} = \mathbf{V}^\mathsf{T}\mathbf{V} = \mathbf{I}$, and $\mathbf{D}_\alpha$ is the diagonal matrix of positive singular values in descending order: $\alpha_1 \geq \alpha_2 \geq \cdots > 0$.

3. Principal coordinates of rows: $\mathbf{F} = \mathbf{D}_r^{-1/2}\mathbf{U}\mathbf{D}_\alpha$ (6)

4. Principal axes: $\mathbf{A} = \mathbf{D}_w^{-1/2}\mathbf{V}$ (7)

5. Standard coordinates of columns: $\mathbf{\Gamma} = \mathbf{D}_w^{1/2}\mathbf{V}$ (8)

6. Contribution coordinates of columns: $\mathbf{\Gamma}^* = \mathbf{D}_w^{-1/2}\mathbf{\Gamma} = \mathbf{V}$ (9)

From (4) and (5) **Y** can be written as: $\mathbf{Y} = (\mathbf{D}_r^{-1/2}\mathbf{U}\mathbf{D}_\alpha)(\mathbf{V}^\mathsf{T}\mathbf{D}_w^{-1/2}) = \mathbf{F}\mathbf{A}^\mathsf{T}$, where **F** is the matrix of row principal coordinates (6) and the columns of $\mathbf{A} = \mathbf{D}_w^{-1/2}\mathbf{V}$ are the principal axes (7): each row of **Y** is thus a linear combination of the rows of $\mathbf{A}^\mathsf{T}$, i.e. the *i*-th row of **Y**, written as a column vector, is a linear combination of the principal axes, where the coefficients of the linear combination are the principal coordinates in the *i*-th row of **F**. Notice that the principal axes are orthornormal in the metric $\mathbf{D}_w$, forming a new set of basis vectors for the rows of **Y**: $\mathbf{A}^\mathsf{T}\mathbf{D}_w\mathbf{A} = (\mathbf{V}^\mathsf{T}\mathbf{D}_w^{-1/2})\mathbf{D}_w(\mathbf{D}_w^{-1/2}\mathbf{V}) = \mathbf{I}$. Rows (cases) are conventionally depicted by points, almost always in principal coordinates, which are the projections of the case vectors $\mathbf{y}_i$ onto the principal axes (projections are always in the metric defined by $\mathbf{D}_w$): $\mathbf{Y}\mathbf{D}_w\mathbf{A} = (\mathbf{D}_r^{-1/2}\mathbf{U}\mathbf{D}_\alpha\mathbf{V}^\mathsf{T}\mathbf{D}_w^{-1/2})\mathbf{D}_w(\mathbf{D}_w^{-1/2}\mathbf{V}) = \mathbf{D}_r^{-1/2}\mathbf{U}\mathbf{D}_\alpha = \mathbf{F}$ (see (6)). The columns (variables) are conventionally depicted by arrows and in one of two scalings, either standard coordinates or contribution coordinates. The standard coordinates of the variables are projections onto the principal axes of unit vectors in the full space of the variables (e.g., $[1\ \ 0\ \ 0\ \ \cdots\ \ 0]$ for the first variable, $[0\ \ 1\ \ 0\ \ \cdots\ \ 0]$ for the second variable, etc…, constituting an identity matrix **I**): $\mathbf{I}\mathbf{D}_w\mathbf{A} = \mathbf{I}\mathbf{D}_w(\mathbf{D}_w^{-1/2}\mathbf{V}) = \mathbf{D}_w^{-1/2}\mathbf{V} = \mathbf{\Gamma}$ (see (8)). The contribution coordinates in $\mathbf{\Gamma}^*$, simply equal to the singular vectors **V** (i.e., the standard coordinates **Γ** multiplied by the inverse square roots of the variable weights – see (8) and (9)) form a useful alternative scaling for the variables, especially when the variables have different weights. These coordinates maintain the directions of the standard coordinates but rescale their lengths so that their squared values along principal axes are the variables' contributions to the respective axes (Greenacre 2013). A biplot of the cases and variables in a two-dimensional solution, say, would use the first two columns of $\mathbf{F}$ for the cases and either $\mathbf{\Gamma}$ or $\mathbf{\Gamma}^*$ for the variables. The total inertia is the sum of squares of the singular values $\alpha_1^2+\alpha_2^2+\ldots$; the inertia accounted for in a two-dimensional solution, say, is the sum of the first two terms $\alpha_1^2+\alpha_2^2$; while the inertia not accounted for (i.e., the residual inertia (3)) is the sum of the remaining ones: $\alpha_3^2+\alpha_4^2+\ldots$.

Apart from this simple decomposition of the inertia in the data matrix, there is another benefit of the weighted least-squares approach via the SVD, namely a further breakdown of the inertia for each point along each principal axis. For example, since $\mathbf{F}^\mathsf{T}\mathbf{D}_r\mathbf{F} = \mathbf{D}_\alpha^2$ (from (6)), $\sum_i r_i f_{ik}^2 = \alpha_k^2$, so each $r_i f_{ik}^2$ is a contribution of the i-th point to the k-th axis's inertia of α_k^2, and at the same time $r_i f_{ik}^2$ is a contribution of the k-th axis to the i-th point's inertia of $\sum_k r_i f_{ik}^2$. These contributions give very useful diagnostics for quantifying the quality of representation of the points and are routinely computed in correspondence analysis. When applied to the columns (the variables) they form the basis of the contribution coordinates, showing explicitly which variables contribute to each principal axis of the solution – see Greenacre (2013) for more details.

3. Computing the variable weights

We now consider the case when any measure of distance or dissimilarity measure is used between cases, not necessarily Euclidean-embeddable. Using conventional MDS notation (Borg and Groenen, 2005) let us suppose that δ_{ij} is the observed distance/dissimilarity between individuals i and j based on their description vectors $\mathbf{x}_i$ and $\mathbf{x}_j$. We use $d_{ij} = d_{ij}(\mathbf{w})$ to indicate the weighted Euclidean distance of the form (2) based on (unknown) weights in the vector **w**. The problem is then to find the weights that give the best fit to the observed dissimilarities by optimizing the fit to distances through least-squares scaling (LSS) by stress, that is, minimizing

$$\sigma^2(\mathbf{w}) = \frac{\sum\sum_{i<j}\left(\delta_{ij} - d_{ij}(\mathbf{w})\right)^2}{\sum\sum_{i<j}\delta_{ij}^2} \tag{10}$$

over **w**. The notation $\sum\sum_{i<j}$ denotes summation $\sum_{j=2}^{n}\sum_{i=1}^{j-1}$ over the half triangle of the corresponding $n \times n$ square matrices. We follow the algorithmic approach for minimizing $\sigma^2(\mathbf{w})$ by the method of *majorization* (De Leeuw, 1977, 1988; Borg and Groenen, 2005). The extension of the method by De Leeuw and Heiser (1980) allows a variety of restrictions to be incorporated. Commandeur and Heiser (1993) worked out the theory in detail for a variety of dimension-weighting models, including for the weighted Euclidean distance. The approach taken here is the same as Commandeur and Heiser (1993), except that it is focused only on updating the weights **w** in the weighted Euclidean distance. Full details are given in Appendix 1. Note that the masses r_i assigned to the cases can be taken into account in the fitting, in which case the (i,j)-th squared terms in the numerator and denominator of (10) are multiplied by $r_i r_j$ – this is a simple generalization of the algorithm in Appendix 1.

The goodness of fit of the weighted Euclidean distances to the original distances at an optimum can be measured by the squared Tucker's congruence coefficient

$$\frac{\left(\sum\sum_{i<j}\delta_{ij}d_{ij}(\mathbf{w})\right)^2}{\sum\sum_{i<j}\delta_{ij}^2\sum\sum_{i<j}d_{ij}^2(\mathbf{w})} \tag{11}$$

(Tucker, 1951) or, equivalently, by 1 minus the stress. Our biplot procedure thus passes through two stages of approximation, first the fitting of the distances by estimating the variable weights, and second the matrix approximation of the generalized SVD, defined in (4)–(9), to give the graphical display of the weighted Euclidean distances and the associated biplot vectors for the variables.

In the cases of the aforementioned methods already based on weighted Euclidean distances, namely correspondence analysis and principal component analysis of standardized data, these are subsumed in the weighted Euclidean biplot approach. For example, if Euclidean distances are computed on standardized data, the estimated squared weights will be exactly the inverses of the variables' variances and the weighted Euclidean biplot will be equivalent to the PCA.

4. Application – the Bhattacharya (arc cos) distance

This research was originally motivated by an article in the Catalan statistical journal *Qüestiió* (now published in English under the name *SORT*) by Vives and Villaroya (1996), who applied "intrinsic data analysis" (Rios, Villaroya and Oller, 1994) to visualize a compositional data matrix, specifically the composition in each of the 41 Catalan counties (called *comarques*) of eight different professional groups. The full table, in percentage form, is given in the Appendix 2 – in what follows we use the data in the form of proportions. The analysis of Vives and Villaroya (1996) is based on the Bhattacharyya distance between the 41 counties:

$$d^2(\mathbf{p}_i, \mathbf{p}_j) = \text{arc cos}\,(\textstyle\sum_k \sqrt{p_{ik} p_{jk}}\,) \qquad (12)$$

where p_{ik} is the proportion of professional group k in county i , $\mathbf{p}_i$ is the vector of proportions for county i, and the function arc cos is the inverse cosine. The same authors report that their results are almost identical to those of correspondence analysis (CA). In CA the inherent weights are the inverses $1/c_k$ of the column averages.

Using the majorization approach to fit weighted Euclidean distances to the arc cos distances, the weights are estimated to be the following for the eight professional groups, and compared to the inherent CA weights:

Weights estimated by LSS fitting to Bhattacharyya distances

Pro&Tec	*PersDir*	*ServAdm*	*Com&Ven*	*Hot&Alt*	*Agr&Pes*	*Indust*	*ForArm*
1.62	2.10	2.23	1.52	1.47	1.31	0.90	5.37

Weights implied by correspondence analysis

Pro&Tec	*PersDir*	*ServAdm*	*Com&Ven*	*Hot&Alt*	*Agr&Pes*	*Indust*	*ForArm*
1.59	3.60	1.52	1.49	1.62	1.46	0.79	8.31

The two sets of weights are plotted against each other in Figure 1 – notice that we have rescaled the weights to be comparable, using the mean of the ratios of the two sets of distances as a scaling factor. It is interesting to see that the variable "*ForArm*" (*forces armades* in Catalan, i.e. armed forces) receives much higher weight than the others, very similar to the situation in CA where it is weighted highly because of very low relative frequency and thus low variance – see top right hand point in Figure 1. The arc cos distance inherently weights this variable highly as well, even though this is not at all obvious from its definition in (12). The fact that the estimated weights and the CA weights so closely resemble one another explains why the results obtained by Vives & Villaroya (1996) are so similar to those obtained by CA.

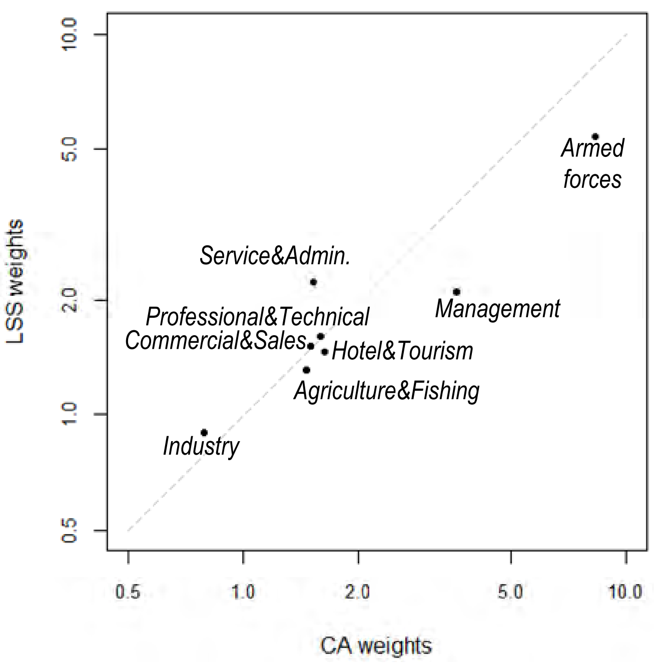

Figure 1: Comparison of estimated weights that give optimal LSS fit to the arc cos distances (vertical axis) and correspondence analysis weights (horizontal axis). Both axes are on a logarithmic scale.

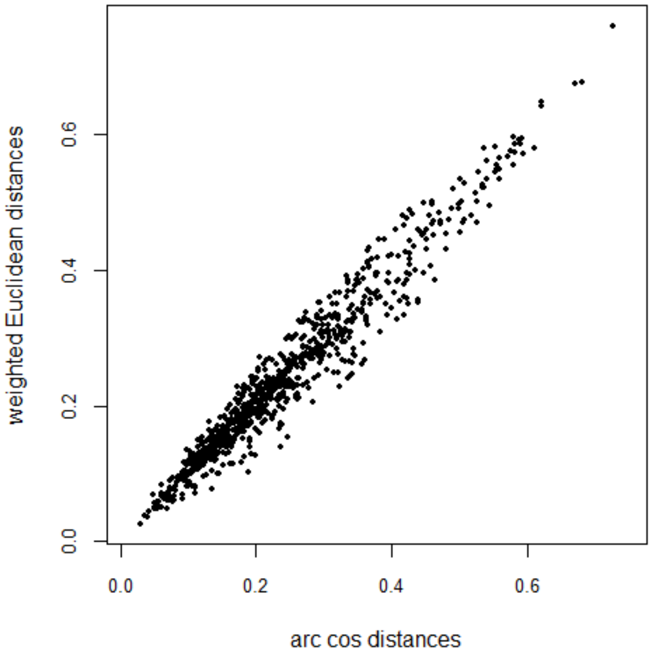

Figure 2: Scatterplot of the weighted Euclidean distances versus the arc cos distances computed between the 41 counties.

The fit of the weighted Euclidean distances to the arc cos distances is excellent: Tucker's squared congruence coefficient equals 0.989. The $41 \times 40/2 = 820$ pairs of distances are plotted in Figure 2.

Figure 3 shows the contribution biplot (Greenacre 2013) of the result, with rows (counties) in principal coordinates so that we can interpret the inter-row distances, and the columns (professional categories) in contribution coordinates. Projecting the rows onto the biplot axes defined by the column vectors will give an approximation to the original data values, while the most outlying columns on the two principal axes are the most determinant in the solution.

Clearly, the professional categories that most distinguish the counties are "agriculture & fisheries" on the first axis and "industrial" on the second, with a closely related set of four categories towards bottom right separating out the counties such as Garraf (Ga), Barcelona (Br) and Val d'Aran (VA), which are higher on some or all of "Services/Administration",

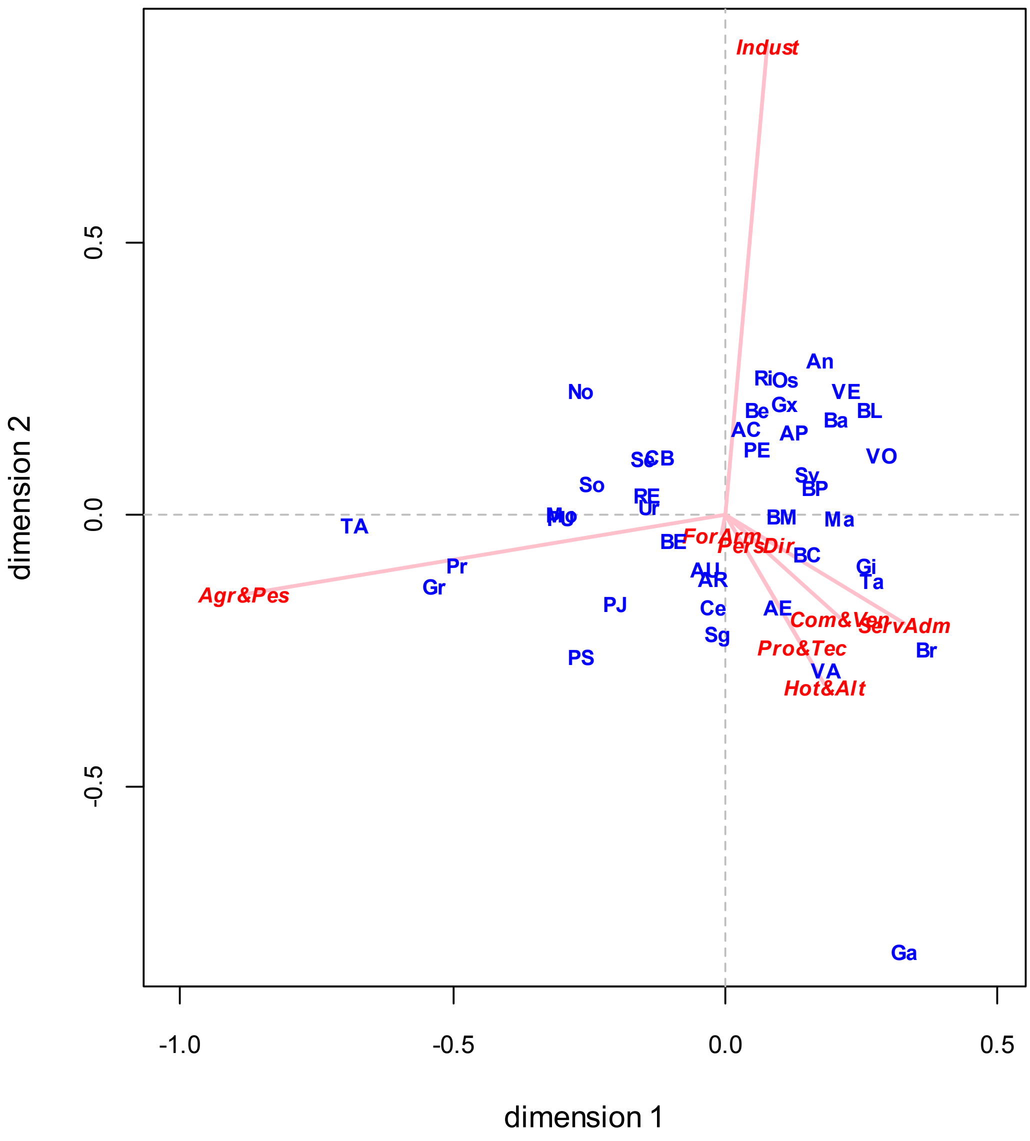

Figure 3: Biplot of 41 Catalan counties (rows, in principal coordinates) and 8 professional categories (columns, in contribution coordinates). The row coordinates have been multiplied by 2 to improve legibility. Percentages of inertia explained on the axes are 54.2% (horizontal first axis) and 37.1% (vertical second axis), totalling 91.3%.

"Hotels/Tourism", "Professional/Technical" and "Commercial/Sales", while being lower on "Agricultural/Fisheries" and "Industrial". "Armed forces" and "Management" have little relevance to the solution and can be ignored.

Table 1 shows the contributions to inertia that are another spin-off of our approach – we show the contributions for the column points, with all values given in thousandths, i.e. permills. The columns headed `CTR` show inertia components relative to the respective principal inertias, or squared singular values, for each of the two dimensions, often called the *absolute contributions* in correspondence analysis – these values are related to the contribution coordinates in the biplot. The columns headed `COR` show inertia components relative to the inertias of the respective column points. The quality of display of each column's inertia in the two-dimensional solution is given in the column headed `QLT` – these are equivalent to the inertia explained of each variable's regression on the two dimensions, for example 62.5% of the inertia of "Professional & Technical" is explained by the two dimensions.

	Quality	*Principal axes* 1		*Principal axes* 2	
	QLT	CTR	COR	CTR	COR
Prof&Tec	625	20	210	57	415
PersDir	411	2	275	2	136
Serv&Admin	773	110	621	39	152
Com&Ven	777	44	501	35	276
Hotel&Altres	661	33	219	98	442
Agric&Pesc	998	784	979	22	19
Indust	999	6	12	745	987
ForArm	142	0	5	1	137
Principal inertias		0.0146		0.0100	
(% of total)		(54.2%)		(37.1%)	

Table 1*:* Contributions of the eight column points along first two principal axes. The principal inertias (eigenvalues, or squared singular values) are decomposed amongst the points as given in the columns `CTR`, given in "permills": for example the first axis is determined mostly by points *Agric&Pesc (Agriculture & Fisheries)* (78.4%). The inertia of a point is decomposed along the principal axes according to the values in the columns `COR` and can also be interpreted as squared correlations of the points with the principal axes. Thus the

point *Indust (Industrial)* is mostly explained by the second axis, while *ForArm* is not well explained by either axis and also plays hardly any role in determining the two-dimensional solution, even with the large weight assigned to it. The column `QLT` refers to quality of display of each variable in the plane, and is the sum of the `COR` columns.

5. Discussion and conclusion

The idea in this paper is to replace the user's preferred distance/dissimilarity measure by the "closest" weighted Euclidean distance, which is not only more manageable but brings with it a host of results to assist in the interpretation as well as the classical biplot displays. In the example presented here it has been possible to approximate the given measure very accurately by a weighted Euclidean distance. The weights allocated to the variables are estimated using the majorization algorithm. These weights are of interest *per se*, since they reflect the intrinsic weighting of the variables implied by the chosen dissimilarity measure, which is usually not obvious from the definition of this measure, as testified by the example of the arc cos distance.

As an alternative approach to the estimation of the weights, one could use squared distance scaling through S-stress. The advantage of that approach is that the weights can be obtained using quadratic programming. The disadvantage is that the weights tend to be dominated by the large dissimilarities. It is for this reason that we prefer the majorization solution through stress proposed in this paper.

Finally, we reiterate that the above approach subsumes regular methods such as correspondence analysis that are already weighted Euclidean. For example, if we computed the chi-squared distances between the counties in Appendix 1 and then applied our weight-estimation procedure, we would recover the exact weights used in the chi-square distance function, and the weighted Euclidean biplot would then be the same as the correspondence analysis biplot.

Acknowledgments

Michael Greenacre's research has been supported by a collaboration with the BBVA Foundation, Madrid, Spain, which has also published his book "Biplots in Practice" and the website www.multivariatestatistics.org. Partial support of the Spanish Ministry of Economics and Competitiveness grant MTM2012-37195 is also acknowledged.

References

Borg, I. and Groenen, P.J.F. (2005). *Modern Multidimensional Scaling, 2^{nd} edition*. New York: Springer.

Commandeur, J.J.F. and Heiser, W.J. (1993). Mathematical derivations in the proximity scaling (PROXSCAL) of symmetric data matrices (Tech. Rep. No. RR-93-03). Leiden, The Netherlands: Department of Data Theory, Leiden University.

De Leeuw, J. (1977). Applications of convex analysis to multidimensional scaling. In J. R. Barra, F. Brodeau, G. Romier and B. van Cutsem (eds), *Recent Developments in Statistics* (pp. 133–145). Amsterdam: North-Holland.

De Leeuw, J. (1988). Convergence of the majorization method for multidimensional scaling. *Journal of Classification*, **5**, 163–180.

De Leeuw, J. and Heiser, W. J. (1980). Multidimensional scaling with restrictions on the configuration. In P. R. Krishnaiah (Ed.), *Multivariate Aalysis* (Vol. V, pp. 501–522). Amsterdam: North-Holland.

De Leeuw, J. and Mair, P. (2009). Multidimensional scaling using majorization : SMACOF in R. *Journal of Statistical Software*, **31**(3).

Gabriel, K. R. (1971) The biplot-graphic display of matrices with applications to principal component analysis. *Biometrika*, **58**, 453-467.

Gower, J.C. and Hand, D.J. (1996) *Biplots*. London: Chapman and Hall.

Gower, J.C. & Legendre, P. (1986) Metric and Euclidean properties of dissimilarity coefficients. *Journal of Classification*, **3**, 5-48.

Gower, J.C., Lubbe, S. and Le Roux, N. (2011). *Understanding Biplots*. Chichester, UK: Wiley.

Greenacre, M.J. (1984) *Theory and Applications of Correspondence Analysis*. London: Academic Press.

Greenacre, M.J. (2007) *Correspondence Analysis in Practice, 2^{nd} edition*. London: Chapman & Hall / CRC. Free download of the Spanish translation of this book from www.multivariatestatistics.org.

Greenacre, M.J. (2010) *Biplots in Practice*. Madrid: BBVA Foundation. Free download from www.multivariatestatistics.org.

Greenacre, M.J. (2013) Contribution biplots. *Journal of Computational and Graphical Statistics*, **22**, 107–122.

Legendre, P. and Legendre, L. (1998) *Numerical Ecology*. Amsterdam: North Holland.

Rios, M., Villaroya, A. and Oller, J.M. (1994) Intrinsic data analysis : a method for the simultaneous representation of populations and variables. Research report 160, Department of Statistics, University of Barcelona.

Tucker, L.R. (1951). A method for the synthesis of factor analysis studies (Tech. Rep. No. 984). Washington, DC: Department of the Army.

Vives, S. and Villaroya, A. (1996) La combinació de tècniques de geometria diferencial amb anàlisi multivariant clàssica: una aplicació a la caracterització de les comarques catalanes. *Qüestiió*, **20**, 449-482.

R development core team (2011). R: a language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0. URL http://www.R-project.org/.

Appendix 1: Computing the variable weights by majorization

The objective for least-squares scaling is:

$$\text{minimize} \quad \sigma^2(\mathbf{w}) = \sum\sum_{i<j} \left(\delta_{ij} - d_{ij}(\mathbf{w})\right)^2$$

where $d_{ij}(\mathbf{w}) = \sqrt{\sum_k w_k^2 (x_{ik} - x_{jk})^2}$. For notational simplicity we assume without loss of generality that $\sum\sum_{i<j} \delta_{ij}^2 = 1$. Expanding $\sigma^2(\mathbf{w})$:

$$\begin{aligned} \sigma^2(\mathbf{w}) &= \sum\sum_{i<j} \delta_{ij}^2 + \sum\sum_{i<j} d_{ij}^2(\mathbf{w}) - 2\sum\sum_{i<j} \delta_{ij} d_{ij}(\mathbf{w}) \\ &= \eta_\delta^2 + \eta^2(\mathbf{w}) - 2\rho(\mathbf{w}). \end{aligned}$$

The term $\eta^2(\mathbf{w})$ can be conveniently written as

$$\eta^2(\mathbf{w}) = \sum_k w_k^2 \sum\sum_{i<j} (x_{ik} - x_{jk})^2 = \sum_k w_k^2 a_k = \mathbf{w}^\mathrm{T}\mathbf{D}_a\mathbf{w}$$

where $a_k = \sum\sum_{i<j} (x_{ik} - x_{jk})^2 = n\sum_k (x_{ik} - \bar{x}_k)^2 = n(n-1)\,\mathrm{var}(\mathbf{x}_k)$ and $\mathbf{D}_a$ is a diagonal matrix with values a_k on the diagonal. The difficult part lies in $\rho(\mathbf{w})$. The core of the majorization method for multidimensional scaling lies in replacing in each iteration $-2\rho(\mathbf{w})$ by a linear function $-2\hat{\rho}(\mathbf{w},\mathbf{s}) = -2\mathbf{w}^\mathrm{T}\mathbf{b}(\mathbf{s})$ such that $-2\rho(\mathbf{w}) \le -2\hat{\rho}(\mathbf{w},\mathbf{s})$ and $-2\rho(\mathbf{w}) = -2\hat{\rho}(\mathbf{w},\mathbf{w})$. Here, $\mathbf{s}$ is the previous estimate of $\mathbf{w}$. Then, in each iteration the so called majorizing function

$$\hat{\sigma}^2(\mathbf{w},\mathbf{s}) = \eta_\delta^2 + \eta^2(\mathbf{w}) - 2\hat{\rho}(\mathbf{w},\mathbf{s}) = \eta_\delta^2 + \mathbf{w}^\mathrm{T}\mathbf{D}_a\mathbf{w} - 2\mathbf{w}^\mathrm{T}\mathbf{b}(\mathbf{s})$$

needs to be minimized. As $\hat{\sigma}^2(\mathbf{w},\mathbf{s})$ is quadratic in $\mathbf{w}$, this is an easy task through the update

$$\mathbf{w}^+ = \mathbf{D}_\mathbf{a}^{-1}\mathbf{b}(\mathbf{s}) \qquad (12)$$

having elements $w_k^+ = a_k^{-1} b_k(\mathbf{s})$. To find a $\mathbf{b}(\mathbf{s})$ such that the two conditions $-2\rho(\mathbf{w}) \le -2\hat{\rho}(\mathbf{w},\mathbf{s})$ and $-2\rho(\mathbf{w}) = -2\hat{\rho}(\mathbf{w},\mathbf{w})$ are satisfied, we consider the Cauchy-Schwartz inequality

$$\sum_k w_k s_k (x_{ik} - x_{jk})^2 \le \sqrt{\sum_k w_k^2 (x_{ik} - x_{jk})^2 \sum_k s_k^2 (x_{ik} - x_{jk})^2}$$

that becomes an equality whenever $\mathbf{w} = \mathbf{s}$. Multiplying both sides by $-1\Big/\sqrt{\sum_k s_k^2 (x_{ik} - x_{jk})^2}$ yields

$$-d_{ij}(\mathbf{w}) = -\sqrt{\sum_k w_k^2 (x_{ik} - x_{jk})^2} \le -\frac{\sum_k w_k s_k (x_{ik} - x_{jk})^2}{\sqrt{\sum_k s_k^2 (x_{ik} - x_{jk})^2}} = -\frac{\sum_k w_k s_k (x_{ik} - x_{jk})^2}{d_{ij}(\mathbf{s})}.$$

Multiplying both sides by δ_{ij} gives

$$-\delta_{ij} d_{ij}(\mathbf{w}) \le -\sum_k w_k s_k \frac{\delta_{ij}}{d_{ij}(\mathbf{s})} (x_{ik} - x_{jk})^2 .$$

The inequalities above assume that $d_{ij}(\mathbf{s}) > 0$. If $d_{ij}(\mathbf{s}) = 0$, then the right part of the inequality is replaced by 0 so that $-d_{ij}(\mathbf{s}) \le 0$ that is trivially true due to the nonnegativity of the Euclidean distance. Thus, let $c_{ij} = \delta_{ij} / d_{ij}(\mathbf{s})$ if $d_{ij}(\mathbf{s}) > 0$ and $c_{ij} = 0$ if $d_{ij}(\mathbf{s}) = 0$. Then

$$-\delta_{ij} d_{ij}(\mathbf{w}) \le -\sum_k w_k s_k c_{ij} (x_{ik} - x_{jk})^2 .$$

Summing over i, j gives

$$-2\rho(\mathbf{w}) = -2\sum\sum_{i<j} \delta_{ij} d_{ij}(\mathbf{w}) \le -2\sum_k w_k s_k \sum\sum_{i<j} c_{ij} (x_{ik} - x_{jk})^2 = -2\mathbf{w}^{\mathsf{T}}\mathbf{b}(\mathbf{s}) = -2\hat{\rho}(\mathbf{w},\mathbf{s})$$

with

$$b_k(\mathbf{s}) = s_k \sum\sum_{i<j} c_{ij} (x_{ik} - x_{jk})^2 . \quad (13)$$

The majorization algorithm thus proceeds as follows:

1. Choose a starting value of $\mathbf{s}$, for example, $\mathbf{s} = \mathbf{1}$.
2. For $k = 1,...,m$, $w_k^+ = b_k(\mathbf{s})/a_k = \left(s_k \sum\sum_{i<j} c_{ij} (x_{ik} - x_{jk})^2\right) / \left(n(n-1)\,\mathrm{var}(\mathbf{x}_k)\right)$.
3. Set $\mathbf{s} = \mathbf{w}^+$ and repeat 2 and 3 until convergence.

These computations can be done through the `SMACOF` package (de Leeuw and Mair, 2009) in R (R development core team, 2011). R code for the application reported in this article can be obtained from the authors.

Appendix 2: Percentages of professional groups in Catalan counties

County	Abbrevn	Pro&Tec	PersDir	ServAdm	Com&Ven	Hot&Alt	Agr&Pes	Indust	ForArm
Alt Camp	AC	9.62	1.90	11.30	11.10	6.84	9.89	49.14	0.20
Alt Empordà	AE	8.42	2.26	14.39	15.73	13.77	10.02	34.50	0.91
Alt Penedés	AP	9.08	1.88	13.76	11.55	7.51	6.86	49.23	0.14
Alt Urgell	AU	10.39	1.80	11.15	13.62	10.65	14.26	37.08	1.05
Alta Ribagorça	AR	13.90	1.83	7.78	10.41	15.81	12.95	37.25	0.08
Anoia	An	8.79	1.95	11.01	11.31	7.66	3.57	55.57	0.14
Bages	Ba	11.28	1.84	11.66	12.75	8.22	3.15	50.79	0.31
Baix Camp	BC	12.15	2.11	13.14	14.98	11.13	6.97	39.29	0.23
Baix Ebre	BE	10.85	1.70	10.26	12.46	8.85	16.34	39.25	0.29
Baix Empordà	BM	8.22	2.16	10.87	14.33	13.56	8.03	42.46	0.37
Baix Llobregat	BL	5.80	1.88	14.68	12.59	11.71	1.22	51.99	0.13
Baix Penedés	BP	7.95	2.28	12.14	14.22	12.55	5.59	44.91	0.35
Barcelona	Br	17.13	2.90	21.37	14.81	11.16	0.40	32.07	0.15
Berguedà	Be	10.14	1.21	8.91	11.48	8.35	8.33	51.01	0.58
Cerdanya	Ce	9.96	2.35	9.36	13.75	15.92	13.57	34.33	0.77
Conca de Barbera	CB	8.62	1.90	9.73	9.66	7.47	16.34	46.18	0.11
Garraf	Ga	20.60	3.25	20.22	22.91	21.04	4.94	6.79	0.25
Garrigues	Gr	7.90	1.16	7.68	9.07	6.22	34.27	33.51	0.19
Garrotxa	Gx	10.14	2.07	10.96	10.82	7.54	6.71	51.58	0.17
Gironés	Gi	14.18	2.30	17.22	13.90	9.94	3.35	38.60	0.52
Maresme	Ma	11.85	3.21	13.90	14.37	10.03	4.16	42.30	0.17
Montsía	Mo	6.98	1.48	8.41	10.75	7.32	24.11	40.54	0.40
Noguera	No	7.32	1.20	6.02	7.93	5.33	20.80	51.18	0.23
Osona	Os	9.94	1.83	10.70	11.00	6.57	6.24	53.62	0.10
Pallars Jussà	PJ	12.36	1.72	10.44	10.14	8.94	20.82	33.36	2.20
Pallars Sobirà	PS	13.43	1.29	9.59	7.10	14.72	23.84	29.74	0.29
Pla d'Urgell	PU	8.25	1.62	9.74	9.75	5.71	24.57	40.15	0.23
Pla de l'Estany	PE	10.95	2.22	12.29	10.45	6.96	9.54	47.50	0.09
Priorat	Pr	8.68	1.03	7.41	7.72	7.02	32.16	35.67	0.30
Ribera d'Ebre	RE	12.39	0.99	9.06	8.70	7.84	17.45	43.21	0.36
Ripollés	Ri	9.24	1.76	8.26	10.09	9.18	7.31	53.91	0.25
Segarra	Se	9.93	1.90	9.91	8.50	6.30	17.49	45.89	0.09
Segría	Sg	13.03	2.13	13.76	13.78	10.39	14.42	31.53	0.96
Selva	Sv	7.33	1.96	10.84	12.46	15.20	5.67	46.36	0.17
Solsonés	So	10.15	1.44	7.77	7.42	8.20	21.20	43.67	0.14
Tarragonés	Ta	14.22	2.12	16.61	12.89	12.91	2.90	37.73	0.61
Terra Alta	TA	4.83	0.91	4.90	7.21	4.65	39.10	38.05	0.36
Urgell	Ur	9.06	2.09	9.76	12.70	6.73	17.68	41.72	0.28
Val d'Aran	VA	11.18	6.90	10.84	13.64	21.30	5.42	29.52	1.21
Vallés Occidental	VO	12.05	2.27	14.64	13.20	8.97	0.68	48.09	0.10
Vallés Oriental	VE	9.32	2.19	13.22	11.33	8.19	2.44	53.19	0.12
Average		10.43	2.02	11.36	11.77	9.96	12.31	41.77	0.38

Pro&Tec: professional/technical; *PersDir*: management; *ServAdm*: administration/services; *Com&Ven*: commercial/sales; *Hot&Alt*: hotel/tourism; *Agr&Pes*: agriculture/fisheries; *Indust*: industry; *ForArm*: armed forces