WEIGHTED EUCLIDEAN BIPLOTS

Michael Greenacre & Patrick Groenen

(last available version, December 2015, accepted by *Journal of Classification*)

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.

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 the columns (usually variables) of a matrix in a Euclidean solution space (usually two- or three-dimensional). Typically, a distance approximation is achieved between the cases, depicted by points, whereas the variables are represented by arrows defining directions onto which cases are projected. "Calibrated biplots", first shown by Gabriel and Odoroff (1990), extend these arrows to biplot axes, and the projections of the cases onto these axes yield optimal approximations of the original data values (Gower et al. 2011). In nonlinear biplots, straight-line biplot axes are replaced by curved trajectories. Well-known variants of the biplot such as principal component analysis (PCA) and correspondence analysis (CA) assume a Euclidean-type distance measure between the cases, incorporating some form of normalization of the variables. Biplot solutions, classically estimated by the singular value decomposition (SVD), are an optimal least-squares representation of the data in a low-dimensional Euclidean space, and are easy to understand and interpret. They have convenient properties such as the decomposition of the total variance of the data matrix into contributions by all the elements of the matrix along each dimension of the solution – these elementwise contributions can then be aggregated for each dimension, for each case and for each variable, providing very useful diagnostics for the interpretation.

Our particular interest here is in the linear biplot, but using a more general class of proximity measures defined on the cases (we use the term "proximity" to include both distance and dissimilarity). At present, both linear and nonlinear biplots handle this situation in a two-step approach. In the first step a configuration of the cases is obtained by multidimensional scaling, or some other nonlinear mapping, of the proximity matrix. In the second step, 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, Borg and Groenen 2005: Chapter 4; Greenacre 2010: Chapter 4). The second step of the nonlinear approach adds curved trajectories by circle projection or using differentials (Gower and Harding 1988, Gower and Hand 1996, Gower et al. 2011). We propose a simple but elegant alternative approach, which stays within the linear biplot framework, while allowing any proximity measure to be used between the cases. This is achieved by finding the weighted Euclidean distances that optimally approximate the given proximities, obtained by estimating weights for the respective variables. These estimated weights constitute an interesting result in themselves, since they show how the variables are combined in a familiar Euclidean distance function to produce the given proximities. Furthermore, the weights can be applied to the variables in a straightforward manner for producing the final biplot, with all the good properties of the linear biplot based on Euclidean distances. We thus call this approach a *weighted Euclidean biplot*.

In Sections 2 and 3 we define weighted Euclidean distance and summarize the classical biplot framework with weights on the variables. Then in Section 4 we describe an algorithmic approach to estimate the weights, with specific details given in Appendix 1. In Section 5 we give an example of this approach and conclude with a discussion in Section 6.

2. Weighted Euclidean distance

In principal component analysis (PCA) of a cases-by-variables data matrix $\mathbf{X}$, 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 $\mathbf{X}$ 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,…,m, for the m 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 proximity measures that 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 proximity measures.

The present paper aims to approximate the proximities 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 decomposition of variance across principal axes of both the rows and columns, which provides useful numerical diagnostics in the interpretation and evaluation of the results.

3. Weighted linear biplots

Our main interest is in weighting the variables in the definition of distances between the individuals, or cases, but the cases themselves can also be weighted to differentiate their influence on the solution. To distinguish the two weighting systems, we shall use the terms *mass* for a case weight (usually pre-specified) and *weight* for a variable weight. In this section both row masses and columns weights are assumed given, whereas from Section 4 onwards the variable weights will be estimated in the initial step of our proposed procedure.

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 r_i sum to 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 shown by, for example, Greenacre (1984, Appendix), the solution can be obtained neatly using the generalized singular value decomposition (SVD) of the matrix **Y**. Computationally, using the regular SVD, the steps in finding the solution are to first pre-process the matrix **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)

In (6)-(9) the number of columns is the rank of the matrix **S** in (4). For plotting in p-dimensional space, the first p columns provide the corresponding coordinates (usually $p = 2$ or 3).

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 ⋯ 0] for the first variable, [0 1 0 ⋯ 0] for the second variable, etc…, constituting an identity matrix $\mathbf{I}$): $\mathbf{I}\mathbf{D}_w\mathbf{A} = \mathbf{I}\mathbf{D}_w(\mathbf{D}_w^{-1/2}\mathbf{V}) = \mathbf{D}_w^{-1/2}\mathbf{V} = \boldsymbol{\Gamma}$ (see (8)). The contribution coordinates in $\boldsymbol{\Gamma}^*$, simply equal to the singular vectors $\mathbf{V}$ (i.e., the standard coordinates $\boldsymbol{\Gamma}$ 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 $\boldsymbol{\Gamma}$ or $\boldsymbol{\Gamma}^*$ for the variables. The total inertia is the sum of squares of the singular values $\alpha_1^2+\alpha_2^2+\dots$; 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: $\alpha_3^2+\alpha_4^2+\dots$.

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 (see Greenacre 2007, Chapter 11). 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.

4. 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 $\mathbf{w}$. 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

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

over $\mathbf{w}$. The notation $\sum_{i<j}$ denotes the double summation $\sum_{j=2}^{n}\sum_{i=1}^{j-1}$ over the upper triangle of the corresponding $n \times n$ square matrix. 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 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 $\mathbf{w}$ 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.

An important property of the weighted biplot is that there is an additive decomposition of the sum of squared dissimilarities into error sum of squares and the sum of squares of the weighted data matrix. This property can be seen in the following, where for notational simplicity, we assume without loss of generality that the denominator of (10) is $\sum_{i<j} r_i r_j \delta_{ij}^2 = 1$.

Expanding $\sigma^2(\mathbf{w})$ gives

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

From Borg and Groenen (2005) it is known that at a minimum **w***, the equality $\eta^2(\mathbf{w}) = \rho(\mathbf{w})$ holds. Therefore, we have

$$\sigma^2(\mathbf{w}^*) = \eta_\delta^2 - \eta^2(\mathbf{w}^*) \quad \text{and}$$

$$\eta_\delta^2 = \eta^2(\mathbf{w}^*) + \sigma^2(\mathbf{w}^*) \ ,$$

implying that the decomposition of the squared dissimilarities equals the sum of the squared distances and the sum of squared errors. The (weighted) sum of the squared distances at the minimum **w***can be conveniently expressed as

$$\eta^2(\mathbf{w}^*) = \text{trace}\left[\mathbf{D}_{w^*}^{1/2}\mathbf{X}^\top\mathbf{J}^\top\mathbf{D}_r\mathbf{J}\mathbf{X}\mathbf{D}_{w^*}^{1/2}\right] = \text{trace}\left[\mathbf{Y}^\top\mathbf{D}_r\mathbf{Y}\mathbf{D}_{w^*}\right] = \text{In}(\mathbf{Y}) = \sum_j \alpha_j^2 \ ,$$

where $\mathbf{J} = \mathbf{I} - \mathbf{1}\mathbf{r}^\top$ is the weighted row-centering matrix, and $\mathbf{Y} = (\mathbf{I} - \mathbf{1}\mathbf{r}^\top)\mathbf{X}$. These equations show that η_δ^2 is additively decomposed in an overall error term and the inertia, that is reconstructions per dimension of size α_j^2. Note that the decomposition of $\eta^2(\mathbf{w}^*)$ can also be

done by summing contributions per variable or per observation. Therefore, the inertia refers to that part of the sum of squared proximities that can be reconstructed in a given dimensionality.

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

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

(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 biplot of the weighted Euclidean distances for the cases along with the associated vectors for the variables, as well as complete sets of contributions by dimensions, cases or variables.

In the cases of the aforementioned methods already based on weighted Euclidean distances, namely CA and PCA of standardized data, these are subsumed in the weighted Euclidean biplot. 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 biplot.

5. Application – the Bhattacharyya (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 – Statistics and Operations Research Transactions*) by Vives and Villaroya (1996), who applied "intrinsic data analysis" (Rios, Villaroya and Oller, 1994) to visualize a compositional data matrix. The data consisted of the composition of eight different professional groups in each of the 41 Catalan counties (called

comarques). The full table, in percentage form, is given in the Appendix 2, along with labels and abbreviations for the rows. 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, also called the arc cos distance:

$$d^2(\mathbf{p}_i, \mathbf{p}_j) = \arccos\left(\sum_k \sqrt{p_{ik} p_{jk}}\right) \tag{12}$$

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. This distance function is not Euclidean embeddable, hence a good example for our procedure.

The majorization algorithm is programmed in the **smacof** package (de Leeuw and Mair 2009) in R (R development core team 2013). Using the function `smacofConstraint` in this package to fit weighted Euclidean distances to the arc cos distances, the weights are estimated to be the following for the eight professional groups

Prof&Tech	*Managemt*	*Admin&Serv*	*Comm&Sales*	*Hotel&Tour*	*Agric&Fish*	*Indust*	*ArmedFor*
1.62	2.10	2.23	1.52	1.47	1.31	0.90	5.37

The fit of the weighted Euclidean distances to the arc cos distances is excellent: Tucker's squared congruence coefficient equals 0.989. The 41×40/2 = 820 pairs of distances are plotted in Figure 1.

Insert Figure 1 about here

Figure 2 shows the weighted Euclidean biplot 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. The column contributions are exactly the squares of the contribution coordinates (Greenacre 2013). Thus the most outlying professional groups on the two principal axes are the most determinant in the solution, while the biplot property is preserved

whereby projections of the counties onto the directions defined by the variables give an approximation to the original data values.

.

Insert Figure 2 about here

Clearly, the professional categories that most distinguish the counties are "Agriculture&Fisheries" on the first axis and "Industrial" on the second. A closely related set of four categories towards bottom right separate out the counties such as *Garraf* (Ga), *Barcelona* (Br) and *Val d'Aran* (VA), which are higher on some or all of "Services&Administration", "Hotels&Tourism", "Professional&Technical" and "Commercial&Sales", but less than average 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 constitute one of the main benefits of our approach – here we show the contributions for the column categories, 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. Hence "Agriculture&Fisheries" and "Industrial" have very high contributions to dimensions 1 and 2 respectively, as seen in the contribution biplot of Figure 2. The columns headed `COR` show inertia components relative to the inertias of the respective column categories, interpreted as squared correlations between the categories and the dimensions (i.e., squared factor loadings). The quality of display of each column's inertia in the two-dimensional solution is the sum of the two `COR` values, reported as `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		2	
	QLT	CTR	COR	CTR	COR
Professional&Technical	625	20	210	57	415
Management	411	2	275	2	136
Administration&Services	773	110	621	39	152
Commerical&Sales	777	44	501	35	276
Hotel&Tourism	661	33	219	98	442
Agriculture&Fisheries	998	784	979	22	19
Industrial	999	6	12	745	987
ArmedForces	142	0	5	1	137
Principal inertias		0.0146		0.0100	
(% of total)		(54.2%)		(37.1%)	

***Table 1**:* Contributions of the eight column categories 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" (i.e., thousandths). The inertia of a point is decomposed along the principal axes according to the values in the columns COR (also multiplied by 1000), which can be interpreted as squared correlations of the points with the principal axes. The column QLT refers to quality of display of each variable in the plane, and is the sum of the COR columns.

	Quality	*Principal axes*			
		1		2	
	QLT	CTR	COR	CTR	COR
Alt Camp (AC)	947	1	53	16	894
Alt Empordà (AE)	714	4	179	17	534
Alt Penedés (AP)	886	7	361	15	525
Alt Urgell (AU)	800	1	101	6	699
Alta Ribagorça (AR)	279	0	12	8	267
:	:	:	:	:	:
Terra Alta (TA)	991	194	990	0	1
Urgell (Ur)	778	8	763	0	14
Val d'Aran (VA)	640	15	195	49	445
Vallés Occidental (VO)	968	34	835	8	133
Vallés Oriental (VE)	989	21	473	33	516
Principal inertias		0.0146		0.0100	
(% of total)		(54.2%)		(37.1%)	

***Table 2**:* Contributions of the first and last five counties along first two principal axes. This table is read in an identical way as Table 1 (see Table 1 caption).

Table 2 shows that a similar set of contributions can be obtained for the row points. The first and last five counties are reported: for example, *Terra Alta* (TA) is a county with high contribution to dimension 1 (19.4% of dimension 1's inertia) and the point TA can be seen in Figure 2 to be outlying on the left-hand side of the horizontal axis. On the other hand, AR (*Alta Ribagorça*), with low quality of 0.279 (27.9% of its inertia explained by the first two dimensions) lies near the middle of the configuration, and its multidimensional position is not well represented in the biplot.

Vives and Villaroya (1996) report that their two-dimensional configuration of the counties is very similar to that obtained by correspondence analysis (CA). In CA the squared weights inherent in its weighted Euclidean distance (the chi-square distance) are the inverses $1/c_k$ of the column averages. This weighting is a type of normalization where it is assumed that the variance of a variable consisting of relative counts is approximately equal to (or proportional to) its mean. For multinomial data the chi-square distance is the Mahalanobis distance, based on probabilities equal to the mean relative frequencies − see, for example, Greenacre (2007, pp. 270−271). In our analysis of the arc cos distances the weights are computed to give weighted Euclidean distances as close as possible to the arc cos distances, where the arc cos distances incorporate no apparent assumption about the column variances in their definition (12). Yet when we compare these weights to the CA ones, there is a close agreement, as shown in the scatterplot of Figure 3 (notice that we have rescaled each set to sum to 100 to make them comparable). The variable "*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. The fact that the estimated weights and the CA weights resemble one another finally explains why the results obtained by Vives & Villaroya (1996) are so similar to those obtained by CA.

Insert Figure 3 about here

5. Discussion and conclusion

The idea in this paper is to replace the user's preferred distance/dissimilarity measure, be it metric, non-metric or non-Euclidean, by the "closest" weighted Euclidean distance. The approximating weighted Euclidean distance is not only more manageable but brings with it a host of additional results to assist in the interpretation as well as the simplicity of linear 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.

In summary, the benefits of weighted Euclidean biplots are:

1) The ability to handle any proximity measure computed on a data matrix;
2) The estimation of weights in a weighted Euclidean distance so that the distances optimally approximate the proximities, thus bringing the problem into a standard Euclidean setting and providing weights which show how the variables combine additively in the distance;
3) The decomposition of total variance (i.e., inertia) of the original data matrix, incorporating estimated weights, into contributions by dimensions, by rows or by columns.
4) For a general proximity measure, the maximum dimensionality of the problem is reduced to that of the variables (columns) of the data matrix, not that of the rows, which can be much higher.

The obvious disadvantage of this approach is the loss of some variance in the proximities in the estimation of the weighted Euclidean approximation, a loss that is small in the application considered here. But this disadvantage is no different from the one present in all metric approaches that visualize non-Euclidean proximity measures in a Euclidean space, where the non-Euclidean part of the variance is necessarily sacrificed.

As an alternative approach to the weight estimation step, 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. Hence, the weighted Euclidean biplot extends biplots such as PCA and CA based on weighted Euclidean distances to proximity measures of any type.

References

Borg, I. and Groenen, P.J.F. (2005). *Modern Multidimensional Scaling, 2nd 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.

Gabriel, K.R. and Odoroff, C.L. (1990) Biplots in biomedical research. *Statistics in Medicine*, **9**, 469–485.

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

Gower, J.C. and Harding, S.A. (1988) Nonlinear biplots. *Biometrika*, **75**, 445–455.

Gower, J.C. and 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, 2nd 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 (2013). 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:

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

where $d_{ij}(\mathbf{w}) = \sqrt{\sum_k w_k^2 (x_{ik} - x_{jk})^2}$ and the r_i's are pre-specified non-negative masses, summing to 1, assigned to the cases (for equal masses, $r_i = 1/n$, for n cases). For notational simplicity we assume without loss of generality that $\sum_{i<j} r_i r_j \delta_{ij}^2 = 1$. Expanding $\sigma^2(\mathbf{w})$:

$$\sigma^2(\mathbf{w}) = \sum_{i<j} r_i r_j \delta_{ij}^2 + \sum_{i<j} r_i r_j d_{ij}^2(\mathbf{w}) - 2\sum_{i<j} r_i r_j \delta_{ij} d_{ij}(\mathbf{w})$$

$$= \eta_\delta^2 + \eta^2(\mathbf{w}) - 2\rho(\mathbf{w}).$$

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

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

where $a_k = \sum_{i<j} r_i r_j (x_{ik} - x_{jk})^2 = \sum_i r_i (x_{ik} - \bar{x}_k)^2 = \text{In}(\mathbf{x}_k)$, the inertia (weighted variance) of the k^{th} variable, 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 $-2\,\hat{\rho}(\mathbf{w},\mathbf{s}) = -2\mathbf{w}^\mathsf{T}\mathbf{b}(\mathbf{s})$ 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

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

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

$$\mathbf{w}^+ = \mathbf{D}_a^{-1}\,\mathbf{b}(\mathbf{s}) \tag{12}$$

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

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

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

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

Multiplying both sides by δ_{ij} gives

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

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 $-d_{ij}(\mathbf{s}) \leq 0$ that is trivially true due to the nonnegativity of the Euclidean distance. Thus, let $c_{ij} = \delta_{ij} \big/ d_{ij}(\mathbf{s})$ if $d_{ij}(\mathbf{s}) > 0$ and $c_{ij} = 0$ if $d_{ij}(\mathbf{s}) = 0$. Then

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

Summing over *i, j* and incorporating the row masses r_i gives

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

with

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

The majorization algorithm thus proceeds as follows:

1. Choose a starting value of **s**, for example, **s** = **1**.
2. For $k = 1,...,m$, $w_k^+ = b_k(\mathbf{s})/a_k = \left(s_k \sum_{i<j} r_i r_j c_{ij} (x_{ik} - x_{jk})^2\right) / \mathrm{In}(\mathbf{x}_k)$.
3. Set **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	Professional &Technical	Management	Administration &Services	Commercial &Sales	Hotel &Tourism	Agriculture &Fisheries	Industrial	Armed Forces
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

Figure captions

Figure 1:
Scatterplot of the weighted Euclidean distances versus the arc cos distances computed between the 41 Catalan counties.

Figure 2:
Weighted Euclidean biplot of 41 Catalan counties (rows, in principal coordinates) and 8 professional categories (columns, in contribution coordinates), based on arc cos (Bhattacharyya) distances between counties. 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%.

Figure 3:
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 and each set of weights has been rescaled to sum to 100.

Figure 1

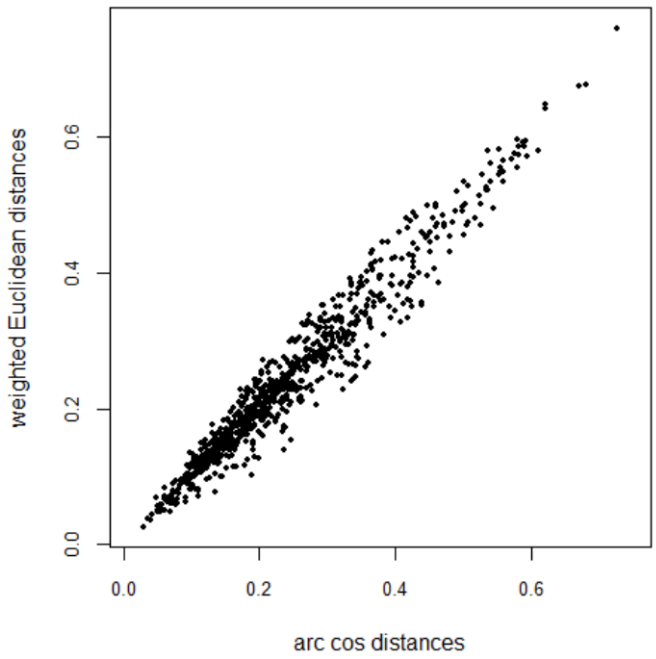

weighted Euclidean distances
0.0
0.2
0.4
0.6
0.0
0.2
0.4
0.6
arc cos distances

Figure 2

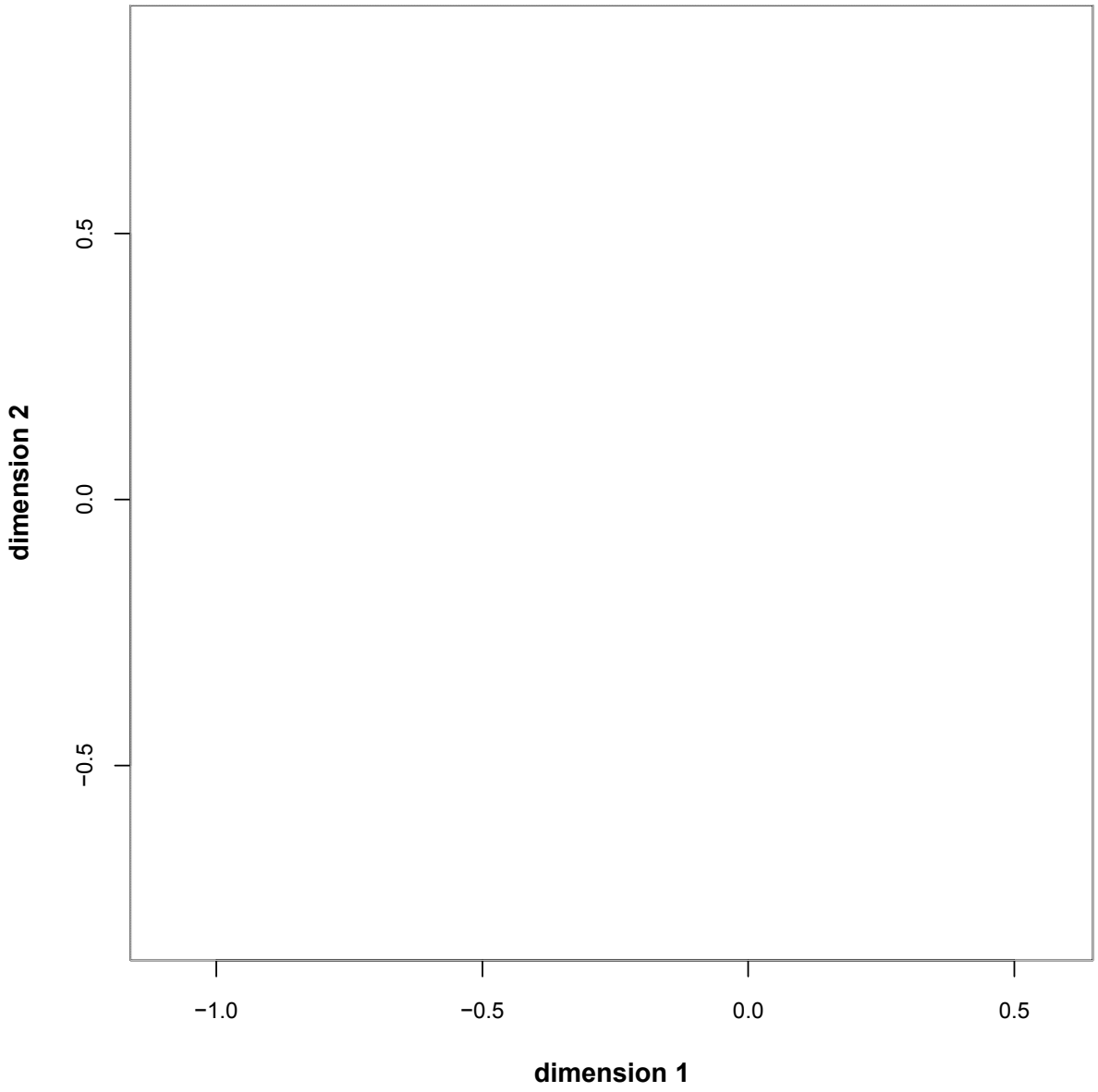
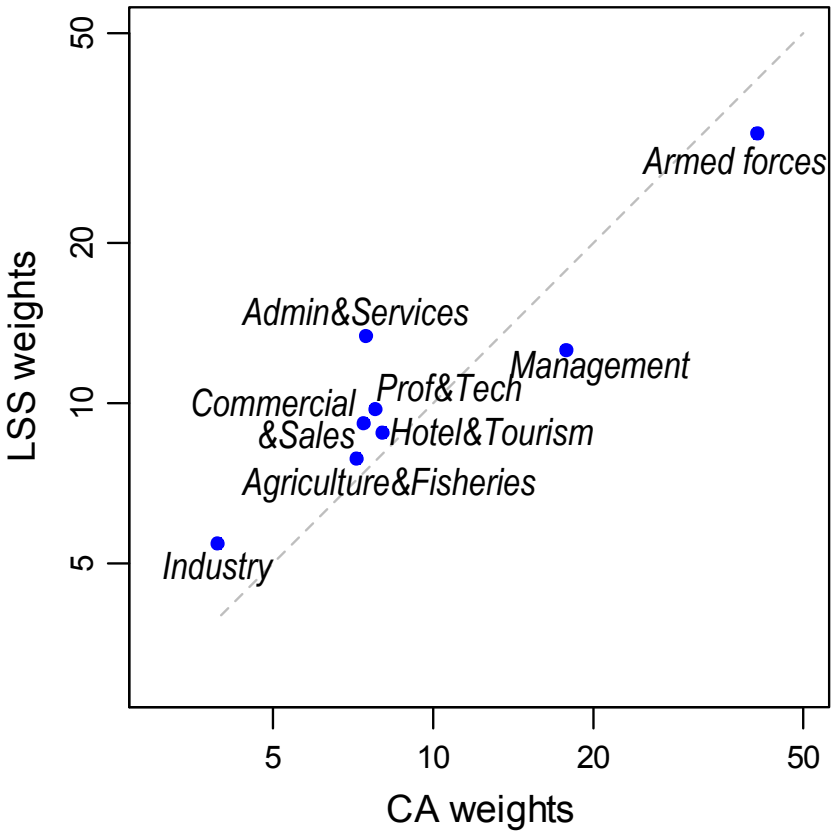


Figure 3


LSS weights
50
20
10
5
CA weights
5
10
20
50
Armed forces
Admin&Services
Management
Prof&Tech
Commercial
&Sales
Hotel&Tourism
Agriculture&Fisheries
Industry