Biplots in Practice

MICHAEL GREENACRE

Professor of Statistics at the Pompeu Fabra University

Chapter 5 Offprint

Reduced-Dimension Biplots

First published: September 2010
ISBN: 978-84-923846-8-6

Supporting websites:
http://www.fbbva.es
http://www.multivariatestatistics.org

CHAPTER 5

Reduced-Dimension Biplots

In the previous chapter, multidimensional scaling (MDS) involved *reduction of dimensionality* in order to visualize a high-dimensional configuration of points in a low-dimensional representation. In some high-dimensional space distances between points are exact (or as near exact as possible for non-Euclidean dissimilarity measures), while they are approximated in some optimal sense in the low-dimensional version. In this chapter we look at the theory and practice of dimensionality reduction, and how a data matrix of a certain dimensionality can be optimally approximated by a matrix of lower, or reduced, dimensionality. Algebraically, the geometric concept of dimensionality is equivalent to the *rank* of a matrix, hence this chapter could also be called *reduced-rank* biplots. This topic is concentrated almost entirely on one of the most useful results in matrix algebra, the *singular value decomposition* (SVD). Not only does this result provide us with a solution to the optimal reduced-rank approximation of a matrix, but it also gives the coordinate values of the points in the corresponding biplot display.

Contents

Matrix approximation

Data matrices usually have many rows (cases) and many columns (variables), such as the 13 × 6 matrix of Exhibit 4.3. The *rank* of a matrix is the minimum number of row or column vectors needed to generate the rows or columns of the matrix exactly through linear combinations. Geometrically, this algebraic concept is equivalent to the *dimensionality* of the matrix—if we were lucky enough to have a data matrix of rank 2, then we could represent the rows or columns in a two-dimensional plot. In practice, however, no large matrix is of low rank, but we can

approximate it optimally by a matrix of low rank and then view this approximate matrix in a low-dimensional space.

Suppose that $\mathbf{Y}$ is an $n \times m$ matrix with rank r (in most examples in practice, r will be n or m, whichever is the smaller). Then the idea of matrix approximation is to find another $n \times m$ matrix $\hat{\mathbf{Y}}$ of lower rank $p < r$ that resembles $\mathbf{Y}$ as closely as possible. Closeness can be measured in any reasonable way, but least-squares approximation makes the solution of the problem particularly simple. Hence we want to find a matrix $\hat{\mathbf{Y}}$ that minimizes the following objective function over all possible rank p matrices:

$$\underset{\hat{\mathbf{Y}} \text{ of rank } p}{\text{minimize}} \ \text{trace}[(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^{\mathsf{T}}] = \sum_{i=1}^{n}\sum_{j=1}^{m}(y_{ij}-\hat{y}_{ij})^2 \qquad (5.1)$$

The notation trace[…] signifies the sum of the diagonal elements of a square matrix, and the square matrix $(\mathbf{Y}-\hat{\mathbf{Y}})(\mathbf{Y}-\hat{\mathbf{Y}})^{\mathsf{T}}$ has exactly all the squared differences between the corresponding elements of $\mathbf{Y}$ and $\hat{\mathbf{Y}}$ down its diagonal. Thanks to our choosing a least-squares objective function, this minimization problem is very simple to solve using a famous result in matrix algebra.

Singular value decomposition (SVD)

The singular value decomposition, or SVD for short, is one of the most useful results in matrix theory. Not only will it provide us with the solution of the matrix approximation problem described above, but it also provides the solution in exactly the form that is required for the biplot. The basic result is as follows: any rectangular $n \times m$ matrix $\mathbf{Y}$, of rank r, can be expressed as the product of three matrices:

$$\mathbf{Y} = \mathbf{U}\,\mathbf{D}_{\alpha}\mathbf{V}^{\mathsf{T}} \qquad (5.2)$$

where $\mathbf{U}$ is $n \times r$, $\mathbf{V}$ is $m \times r$ and $\mathbf{D}_{\alpha}$ is a $r \times r$ diagonal matrix with positive numbers $\alpha_1, \alpha_2, \ldots, \alpha_r$, on the diagonal in descending order: $\alpha_1 \geq \alpha_2 \geq \ldots \geq \alpha_r > 0$. Furthermore, the columns of $\mathbf{U}$ and of $\mathbf{V}$ are *orthonormal*, by which we mean that they have unit length (sum of squares of their elements = 1) and are orthogonal, or perpendicular to one another (i.e., scalar products between columns = 0, that is they are geometrically perpendicular to one another); this property can be written as $\mathbf{U}^{\mathsf{T}}\mathbf{U} = \mathbf{V}^{\mathsf{T}}\mathbf{V} = \mathbf{I}$ (where $\mathbf{I}$ denotes the identity matrix, a diagonal matrix with 1's down the diagonal). The columns $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_r$, and $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r$, of $\mathbf{U}$ and $\mathbf{V}$ are called *left* and *right singular vectors* respectively, and the values $\alpha_1, \alpha_2, \ldots, \alpha_r$ the *singular values* of $\mathbf{Y}$.

If the rank of $\mathbf{Y}$ happened to be low, say 2 or 3, then (5.2) would give us immediately the form "target matrix = left matrix · right matrix" of the biplot we need

(see (1.2)) for a two- or three-dimensional display, the only decision being how to distribute the matrix $\mathbf{D}_\alpha$ of singular values to the left and the right in order to define the biplot's left matrix and right matrix (we shall discuss this critical decision at length soon). In the more usual case that the rank of $\mathbf{Y}$ is much higher than 2 or 3, then the SVD provides us immediately with any low-rank matrix approximation we need, as follows. Define $\hat{\mathbf{Y}}$ as in (5.2) but use only the first p columns of $\mathbf{U}$, the upper left $p \times p$ part of $\mathbf{D}_\alpha$ and the first p columns of $\mathbf{V}$, in other words the first p components of the SVD: $\hat{\mathbf{Y}} = \mathbf{U}_{[p]}\, \mathbf{D}_{\alpha\,[p]} \mathbf{V}_{[p]}^{\mathsf{T}}$, where the subindex $_{[p]}$ means the "first p components". $\hat{\mathbf{Y}}$ is of rank p and is exactly the solution to the least-squares matrix approximation problem. And, once more, the decomposition provided by the SVD is exactly in the form that we need for the biplot.

The singular values provide us with quantifications of the closeness of the approximation of $\hat{\mathbf{Y}}$ to $\mathbf{Y}$. The sum-of-squares of the singular values is equal to the sum-of-squares of the matrix $\mathbf{Y}$: $\text{trace}(\mathbf{Y}\mathbf{Y}^{\mathsf{T}}) = \sum_i \sum_j y_{ij}^2 = \alpha_1^2 + \alpha_2^2 + \ldots + \alpha_r^2$, and the sum-of-squares of the matrix $\hat{\mathbf{Y}}$ is $\text{trace}(\hat{\mathbf{Y}}\hat{\mathbf{Y}}^{\mathsf{T}}) = \alpha_1^2 + \alpha_2^2 + \ldots + \alpha_p^2$, the sum of the first p squared singular values. The latter is expressed as a fraction (or percentage) of the first to quantify the quality of the approximation, while the remainder from 1 quantifies the error (5.1) of the approximation.

Some numerical examples

Consider again the target matrix on the left-hand side of (1.1) and let us pretend we do not know that it decomposes as shown there. The SVD of this matrix calculated in R using the built-in function `svd` in the second command below:

```
> Y<-matrix(c(8,5,-2,2,4,2,0,-3,3,6,2,3,3,-3,-6,-6,-4,1,-1,-2),
  nrow=5)
> svd(Y)
$d
[1]    1.412505e+01  9.822577e+00  6.351831e-16  3.592426e-33

$u
             [,1]         [,2]          [,3]           [,4]
[1,]   -0.6634255   -0.4574027   -0.59215653    2.640623e-35
[2,]   -0.3641420   -0.4939878    0.78954203    2.167265e-34
[3,]    0.2668543   -0.3018716   -0.06579517   -9.128709e-01
[4,]   -0.2668543    0.3018716    0.06579517   -1.825742e-01
[5,]   -0.5337085    0.6037432    0.13159034   -3.651484e-01

$v
             [,1]         [,2]         [,3]         [,4]
[1,]   -0.7313508   -0.2551980   -0.6276102   -0.0781372
[2,]   -0.4339970    0.4600507    0.2264451    0.7407581
[3,]    0.1687853   -0.7971898    0.0556340    0.5769791
[4,]    0.4982812    0.2961685   -0.7427873    0.3350628
```

The `svd` function returns the three parts of the decomposition: the singular values in `$d`, the left singular vectors in `$u` and the right singular vectors in `$v`. It is clear from the singular values that only the first two are nonzero, so the matrix is of rank 2 and can be written as (showing values to 4 decimal places):

$$\begin{pmatrix} 8 & 2 & 2 & -6 \\ 5 & 0 & 3 & -4 \\ -2 & -3 & 3 & 1 \\ 2 & 3 & -3 & -1 \\ 4 & 6 & -6 & -2 \end{pmatrix} = \begin{pmatrix} -0.6634 & -0.4574 \\ -0.3641 & -0.4940 \\ 0.2669 & -0.3019 \\ -0.2669 & 0.3019 \\ -0.5337 & 0.6037 \end{pmatrix} \begin{pmatrix} 14.1251 & 0 \\ 0 & 9.8226 \end{pmatrix} \begin{pmatrix} -0.7314 & -0.4340 & 0.1688 & 0.4983 \\ -0.2552 & 0.4601 & -0.7972 & 0.2962 \end{pmatrix}$$

To define a left and right matrix for the biplot, we can—for example—split the singular values in the middle equally between the left and right singular vectors. That is, multiply the two left singular vectors (two columns above) and the two right singular vectors (two rows above) by the square roots $\sqrt{14.1251} = 3.7583$ and $\sqrt{9.8226} = 3.1341$ respectively. (This way of splitting the singular values equally between the left and right vectors leads to the so-called "symmetric biplot"). This gives the following biplot solution and corresponding plot in Exhibit 5.1, which is:

$$\begin{pmatrix} 8 & 2 & 2 & -6 \\ 5 & 0 & 3 & -4 \\ -2 & -3 & 3 & 1 \\ 2 & 3 & -3 & -1 \\ 4 & 6 & -6 & -2 \end{pmatrix} = \begin{pmatrix} -2.4934 & -1.4335 \\ -1.3686 & -1.5482 \\ 1.0029 & -0.9461 \\ -1.0029 & 0.9461 \\ -2.0059 & 1.8922 \end{pmatrix} \begin{pmatrix} -2.7487 & -1.6311 & 0.6344 & 1.8727 \\ -0.7998 & 1.4418 & -2.4985 & 0.9282 \end{pmatrix}$$

Generalized matrix approximation and SVD

In (5.1) the problem of minimizing fit to a given matrix by another of lower rank was formulated. The idea can be generalized to include a system of weighting on both the rows and columns of the table, the objective being to give them differential importance in the fitting process. For example, in survey analysis the rows are respondents that are often not representative of the population from which they are sampled. If there are proportionally too many women, say, in the sample, then giving lower weights to the individual female respondents can restore the representativeness in the sample. The same is true for the column variables: there are many reasons why some variables may need to be downweighted, for example their variance is by their very nature too high, or there are several variables that basically measure the same trait in the population. The idea of weighting can be carried to the limit of giving zero weight to some respondents or variables—this is the idea behind supplementary points, which will be explained in future chapters.

Suppose then that we have a set of positive weights $w_1, w_2, \ldots, w_n$ for the rows of a matrix and a set of positive weights $q_1, q_2, \ldots, q_m$ for the columns. We can as-

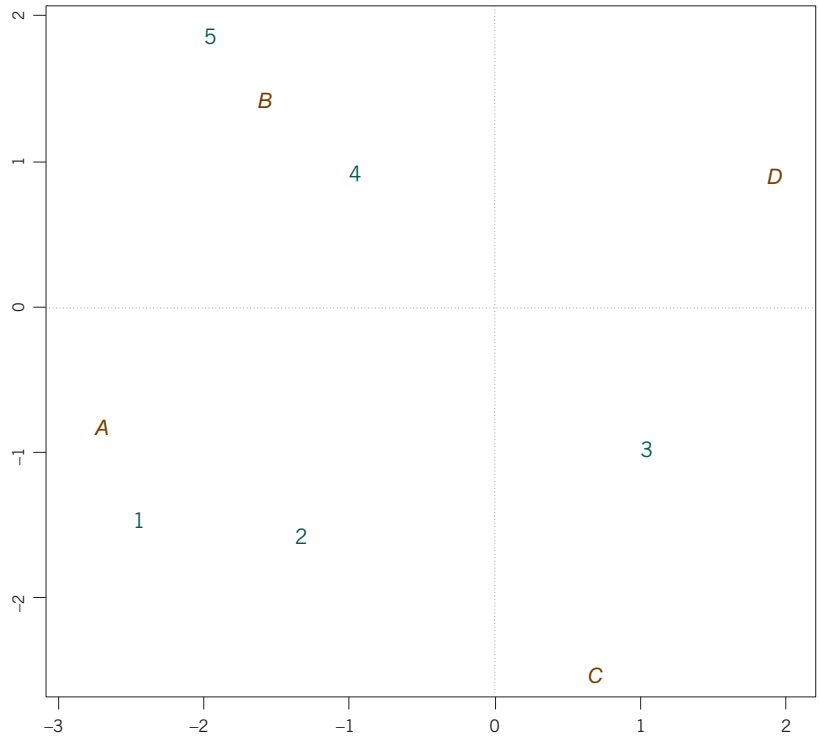

Exhibit 5.1:
Symmetric biplot of the rank 2 example, rows labelled 1 to 5, columns A to D. The square roots of the singular values are assigned to both left and right singular vectors to establish the left and right coordinate matrices. The row–column scalar products perfectly reproduce the original target matrix

sume that the weights add up 1 in each case. Then, rather than the objective (5.1), the weighted (or generalized) matrix approximation is formulated as follows:

$$\underset{\hat{\mathbf{Y}} \text{ of rank } p}{\text{minimize}} \ \text{trace}[\mathbf{D}_w(\mathbf{Y}-\hat{\mathbf{Y}})\mathbf{D}_q(\mathbf{Y}-\hat{\mathbf{Y}})^{\mathsf{T}}] = \sum_{i=1}^{n}\sum_{j=1}^{m} w_i q_j (y_{ij}-\hat{y}_{ij})^2 \qquad (5.3)$$

This solution involves a weighted (or generalized) SVD, which can be solved using the usual (unweighted) SVD as follows: (1) pre-transform the matrix $\mathbf{Y}$ by multiplying its rows and columns by the square roots of the weights, (2) perform the SVD on this transformed matrix as in (5.2), and (3) "untransform" the left and right singular vectors by the inverse square roots of the respective row and column weights. These three steps can be expressed as follows:

$$(1)\ \mathbf{S} = \mathbf{D}_w^{1/2}\mathbf{Y}\mathbf{D}_q^{1/2} \qquad (5.4)$$

$$(2)\ \mathbf{S} = \mathbf{U}\mathbf{D}_\beta\mathbf{V}^{\mathsf{T}} \qquad (5.5)$$

$$(3)\ \tilde{\mathbf{U}} = \mathbf{D}_w^{-1/2}\mathbf{U}, \text{ and } \tilde{\mathbf{V}} = \mathbf{D}_q^{-1/2}\mathbf{V} \qquad (5.6)$$

The best-fitting matrix of rank p, which minimizes (5.3), is calculated as before, but using $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$:

$$\hat{\mathbf{Y}} = \tilde{\mathbf{U}}_{[p]}\mathbf{D}_{\beta[p]}\tilde{\mathbf{V}}_{[p]}^{\mathsf{T}} \tag{5.7}$$

Notice that $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$ have columns (the generalized singular vectors) that are of unit length and orthogonal in terms of weighted sum of squares and weighted sum of cross-products: $\tilde{\mathbf{U}}^{\mathsf{T}}\mathbf{D}_w\tilde{\mathbf{U}} = \tilde{\mathbf{V}}^{\mathsf{T}}\mathbf{D}_q\tilde{\mathbf{V}} = \mathbf{I}$. The singular values β_1, β_2,..., β_p in (5.7) are then split between the left and right singular vectors to obtain the left and right matrices in the biplot, either by assigning their square roots to the left and right, or by assigning the singular values completely to either the left or the right.

Generalized principal component analysis

The introduction of weights into the matrix approximation broadens the class of methods that can be defined in terms of the SVD. All the biplots of interest turn out to be special cases, depending on the definition of the matrix $\mathbf{Y}$ to be approximated, and the weights $\mathbf{w}$ and $\mathbf{q}$ assigned to the rows and columns. A useful general method, which we call generalized *principal component analysis* (PCA), includes almost all the techniques to be described in the rest of this book. In this definition we think of the matrix either as a set of rows or as a set of columns—we shall assume that we think of the rows as points in a multidimensional space.

Suppose that the rows of $\mathbf{X}$ ($n \times m$) define n points $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$, in m-dimensional space—notice that vectors are always denoted as column vectors, so that

$$\mathbf{X} = \begin{pmatrix} \mathbf{x}_1^{\mathsf{T}} \\ \mathbf{x}_2^{\mathsf{T}} \\ \vdots \\ \mathbf{x}_n^{\mathsf{T}} \end{pmatrix}$$

The points have weights in the $n \times 1$ vector $\mathbf{w}$, where the weights are positive and sum to 1: $\mathbf{1}^{\mathsf{T}}\mathbf{w} = 1$ ($\mathbf{1}$ is an appropriate vector of ones in each case). Distances in the m-dimensional space are defined by a weighted metric where the dimensions are weighted by the positive elements of the $m \times 1$ vector $\mathbf{q}$: for example, the square of the distance between the i-th and i'-th rows $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ is $(\mathbf{x}_i - \mathbf{x}_{i'})^{\mathsf{T}}\mathbf{D}_q(\mathbf{x}_i - \mathbf{x}_{i'})$. The objective is to find a low-dimensional version of the rows of $\mathbf{X}$ which are the closest to the original ones in terms of weighted least-squared distances.

There is a side result which proves that the low-dimensional solution necessarily includes the centroid (weighted average) of the points, so we can centre all the points beforehand. This is easily proved by assuming that the low-dimensional so-

lution does not include the centroid and then arrive at a contradictory conclusion. The centroid can, in fact, be thought of as the closest "zero-dimensional subspace" (i.e., a point) to the n points. This means that we first centre the row points by subtracting their centroid $\mathbf{w}^{\mathsf{T}}\mathbf{X}$:

$$\mathbf{Y} = \mathbf{X} - \mathbf{1}\mathbf{w}^{\mathsf{T}}\mathbf{X} = (\mathbf{I} - \mathbf{1}\mathbf{w}^{\mathsf{T}})\mathbf{X} \tag{5.8}$$

The matrix $(\mathbf{I} - \mathbf{1}\mathbf{w}^{\mathsf{T}})$ is called the *centring matrix*: the rows of $\mathbf{Y}$ now have a centroid of $\mathbf{0}$: $\mathbf{w}^{\mathsf{T}}\mathbf{Y} = \mathbf{0}$.

To find an approximating matrix $\hat{\mathbf{Y}}$ of rank $p < m$, the rows of which come closest to the rows of $\mathbf{Y}$ in terms of weighted sum of squared distances, we need to solve the following:

$$\underset{\hat{\mathbf{Y}} \text{ of rank } p}{\text{minimize}} \ \sum_{i=1}^{n} w_i(\mathbf{y}_i - \hat{\mathbf{y}}_i)^{\mathsf{T}}\mathbf{D}_q(\mathbf{y}_i - \hat{\mathbf{y}}_i) \tag{5.9}$$

which is identical to (5.3). Hence the solution is contained in the generalized SVD described above, with the matrix approximation given by (5.7). The coordinates of the row points in the low-dimensional display are given by

$$\mathbf{F} = \tilde{\mathbf{U}}_{[p]}\mathbf{D}_{\beta[p]} \tag{5.10}$$

called the *principal coordinates* (of the rows in this case), which thus form the left matrix of the biplot. The right matrix representing the columns is then $\tilde{\mathbf{V}}_{[p]}$, defining biplot axes. The singular values are thus assigned totally to the singular vectors corresponding to the rows in this case. Most of the subsequent chapters will deal with applications of this theory.

Classical multidimensional scaling with weighted points

The MDS problem of Chapter 4 can be formulated as a SVD problem as well, in fact the matrix decomposed is square symmetric and the SVD reduces to its special case, the eigenvalue/eigenvector decomposition, or *eigendecomposition*. The general formulation for the case when points are weighted is as follows:

- Suppose that the matrix of squared distances between n objects is denoted by $\mathbf{D}^{(2)}$ and that the objects are weighted by the n positive elements in the vector $\mathbf{w}$.
- Double-centre $\mathbf{D}^{(2)}$ using the weights in $\mathbf{w}$ (the centring matrix is $\mathbf{I} - \mathbf{1}\mathbf{w}^{\mathsf{T}}$, pre-multiplied to centre the rows, or transposed and post-multiplied to centre the columns), weight the points by pre- and post-multiplying by $\mathbf{D}_w^{½}$, and finally multiply the result by $-½$ before calculating the eigendecomposition

$$\mathbf{S} = -\tfrac{1}{2}\,\mathbf{D}_w^{\frac{1}{2}}(\mathbf{I} - \mathbf{1}\mathbf{w}^\top)\mathbf{D}^{(2)}(\mathbf{I} - \mathbf{1}\mathbf{w}^\top)^\top\mathbf{D}_w^{\frac{1}{2}} = \mathbf{U}\mathbf{D}_\lambda\mathbf{U}^\top \tag{5.11}$$

– Calculate the coordinates of the points: $\mathbf{F} = \mathbf{D}_w^{-\frac{1}{2}}\mathbf{U}\mathbf{D}_\lambda^{\frac{1}{2}}$ (5.12)

If we start off with a matrix $\mathbf{X}$ as in the previous section, where squared distances between rows are calculated in the metric $\mathbf{D}_q$, with points weighted by $\mathbf{w}$, then the above algorithm gives the same coordinates as the principal coordinates in (5.10), and the eigenvalues here are the squared singular values in the generalized PCA: $\lambda_k = \beta_k^2$.

SUMMARY: Reduced-Dimension Biplots

1. Reduced-dimension biplots rely on approximating a matrix of high dimensionality by a matrix of lower dimensionality. The matrix of low dimensionality is then the target matrix for the biplot.
2. If approximation of a matrix is performed using a least-squares objective, then the singular value decomposition (SVD) provides the solution in a very convenient form as the product of three matrices: the left and right matrices of the biplot are then provided by distributing the second matrix of the SVD (the diagonal matrix of singular values) to the first and third matrices (of singular vectors).
3. The objective of least-squares matrix approximation can be generalized to include weights for the rows and the columns. This leads to a simple modification of the SVD, called the generalized SVD, involving pre-transformation of the matrix to be approximated and post-transformation of the singular vectors.
4. Generalized principal component analysis (PCA) is a geometric version of matrix approximation, where a set of n vectors in m-dimensional space is projected onto a subspace of lower dimensionality. The resultant reduced-dimension biplot depicts the approximate positions of the n points along with m directions showing the biplot axes.
5. Multidimensional scaling (MDS), including the general case where points have any positive weights, can also be formulated as an eigenvalue/eigenvector special case of the SVD problem, because the matrix decomposed is square and symmetric. The resultant coordinates are identical to those found in generalized PCA, if the interpoint distances are defined using the same metric.