THE STATA JOURNAL

The *Stata Journal*, electronic version (ISSN 1536-8734) is a publication of Stata Press, and Stata is a registered trademark of StataCorp LP.

The Stata Journal (2005)
5, Number 2, pp. 208–223

Data inspection using biplots

Ulrich Kohler
Wissenschaftszentrum Berlin
kohler@wz-berlin.de

Magdalena Luniak
Wissenschaftszentrum Berlin
luniak@wz-berlin.de

Abstract. Biplots display interunit distances, as well as variances and correlations of variables of large datasets. They can be used as a tool to reveal clustering, multicollinearity, and multivariate outliers, and to guide the interpretation of principal component analyses (PCA). This article describes the uses of biplots and its implementation in Stata.

Keywords: gr0011, biplot, biplot8, principal component analysis, exploratory data analysis, multivariate statistics, euclidean distance, mahalanobis distance, relative variation diagram, projection

[Editors note: This article was received before Stata 9 was announced. Stata 9 has a `biplot` command, so the command documented here is named `biplot8`. `biplot8` has some features not found in Stata 9 `biplot` (and vice versa). Additionally, the exposition here acts as a helpful supplement to the Stata 9 `biplot` manual entry.]

1 Introduction

Biplots are projections of multivariate datasets that show the following quantities of a data matrix:

- the variance–covariance structure of the variables
- the values of observations on variables
- the Euclidean distances between observations in the multidimensional space

They are helpful for revealing clustering, multicollinearity, and multivariate outliers of a dataset, and they can be also used to guide the interpretation of principal component analyses (PCA).

Biplots were first described thoroughly by Gabriel (1971) and were extended more recently in a monograph by Gower and Hand (1996). They are heavily used in the context of principal component analysis (Jolliffe 2002, 90–107) but also useful as a tool for data inspection in the context of statistical modeling. As a projection technique, they share similarities with many other projection techniques, such as multidimensional scaling (Kruskal and Wish 1978), principal coordinate analysis (Fenty 2004), and correspondence analysis (Blasius and Greenacre 1998).[1]

[1] A discussion of the relative merits of several projection techniques can be found in Schnell and Matschinger (1994), who recommend using biplots.

In this article, we start with examples to explain the interpretation of biplots. We then discuss the mathematical background and some computational issues. Finally, we illustrate the uses of the Stata program `biplot8`.

2 Interpretation

Biplots consists of lines and dots. Lines are used to reflect the variables of the dataset, and dots are used to show the observations. An example biplot is shown in figure 1, which uses a dataset from Hamilton (1992, 268). The observations of this dataset are planets, and the variables are their physical characteristics, for example the mass, the number of moons, and the distance from the sun. With the exception of a dummy variable for rings present, all variables are measured on a logarithmic scale.

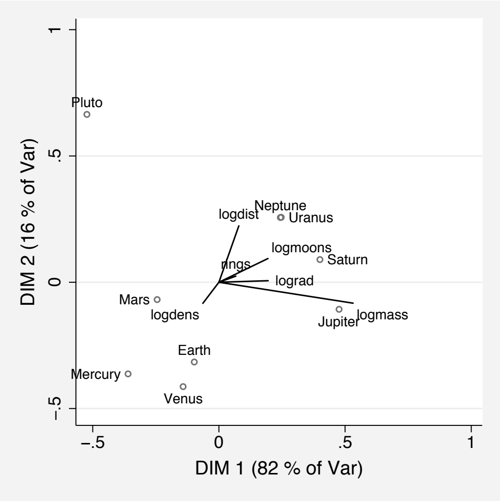

Figure 1: Biplot of `planets.dta`

In a biplot, the length of the lines approximates the variances of the variables. The longer the line, the higher is the variance. Inferring from figure 1 the logarithmic mass of the planets (logmass) has by far the highest variance among the variables in the biplot, while the dummy variable for rings present (rings) has the lowest.

The angle between the lines, or, to be more precise, the cosine of the angle between the lines, approximates the correlation between the variables they represent. The closer the angle is to 90, or 270 degrees, the smaller the correlation. An angle of 0 or 180 degrees reflects a correlation of 1 or −1, respectively. The biplot in figure 1 shows a strong relationship between the ring dummy and the number of moons (`logmoons`), and a weak relationship between the mass and distance from the sun (`logdist`). The correlation between the density and each of the other variables is negative.

The cutpoint of a perpendicular from a specific point to a variable line approximates the value of that observation on the variable that the line represents. If the cutpoint falls on the origin, the value of the observation is approximately the average of the respective variable. Cutpoints far off in the direction of the variable line indicate high values, while cutpoints far off on the variable line, which has been extended through the origin, represent low values. Therefore, Jupiter stands out with the highest mass, followed by Saturn, and Neptune and Uranus, which have almost identical masses. Pluto stands out as the planet with the lowest mass.

Finally, the distance between two points approximates the Euclidean distance between two observations in the multivariate space. Observations that are far away from each other have a high Euclidean distance, and vice versa. In the example biplot, the highest Euclidean distance is observed between Jupiter and Pluto, while Neptune and Uranus are the other extremes.

Putting all these together, biplots reveal several characteristics of a dataset, which are useful in the context of statistical modeling. First of all, you might be warned of possible sources for multicollinearity, as for the variables `rings` and `logmoons` in the biplot example in figure 1. Furthermore, biplots show multivariate outliers, such as the planet Pluto. Finally, biplots can be used to detect clusters, such as the inner rocky planets and the outer gas giants.

The latter two interpretations can be also found in a principal component score plot, which is a common technique for plotting the results of a PCA (Hamilton 1992). In fact (see section 4), for a certain type of the biplot, the scatter of observations *is* a principal component score plot. In this special case, the positions of the observations approximate the scores of the observations on the first two principal components, whereby the x- and y-axes represent the first and second principal components, respectively.

Another useful application of biplots in the context of PCA is more obvious in the biplot of the variables miles per gallon, price, weight, and displacement of `auto.dta` in figure 2. As before, this plot reveals the correlation structure of the variables and some clustering of observations. However, more important for here is the position of the endpoints of the variable lines along the graph axes. The variables `mpg`, `weight`, and `displacement` are relatively far from the origin along the x-axis but close to the origin along the y-axis. For `price`, it is the other way around. These relative positions of the variable lines represent the PCA coefficients ("loadings") of the variables on the first two principal components. Therefore, you might interpret the first principal component as a consumption dimension and the second as a price dimension. In addition, looking at the graph, you can conclude that a slight rotation of the axes of the PCA would improve the ease of interpretation of both components.

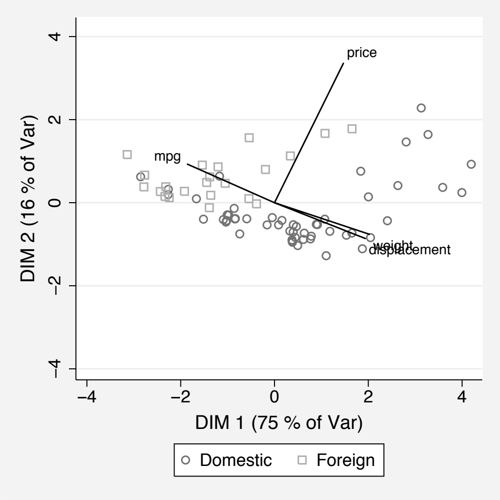

Figure 2: Biplot of `auto.dta`

This interpretation of the biplot is similar to the interpretation of the plot of the PCA coefficients, which is a common way to plot the results of a PCA (Tabachnik and Fidell 1989, 637–638). As for the principal component score plot, the plot of PCA coefficients can be regarded as a special case of a biplot.

3 Mathematical background

Let $\mathbf{Y}$ be an $n \times k$ matrix holding the data. You can decompose $\mathbf{Y}$ with a *singular value decomposition* (SVD) into

$$\mathbf{Y} = \mathbf{ULV}'$$

where $\mathbf{U}$ is $n \times k$, and both $\mathbf{L}$ and $\mathbf{V}$ are $k \times k$. The elements of $\mathbf{L}$, which is diagonal, are the so called *eigenvalues*.

From the singular value decomposition, the coordinates of the observations are given by

$$\mathbf{G} = \mathbf{UL}^c \tag{1}$$

and the coordinates for the variables are given by

$$\mathbf{H}' = \mathbf{L}^{1-c}\mathbf{V}' \tag{2}$$

In (1) and (2), the scalar c can take any value between zero and one. Regardless of the value of c, the equation

$$\mathbf{GH}' = \mathbf{UL}^c\mathbf{L}^{1-c}\mathbf{V}' = \mathbf{ULV}' = \mathbf{Y}$$

always holds. However, as **G** is $n \times k$ and **H** is $k \times k$, all the coordinates have k dimensions. To plot these coordinates in a two-dimensional space, you must select two of them. Usually this is done by choosing those columns of **G** and **H** that correspond to the highest eigenvalues in **L**. This is the default setting in `biplot8`, but other settings are possible (see section 5).

In any case, using fewer than k dimensions to plot the points will lead to a loss of information, and the data matrix **Y** is only approximated by the multiplication of the reduced forms of **G** and **H**. In effect, the interpretations shown in section 2 get less valid if this approximation gets bad. To indicate the quality of the approximation, the default axis titles mention the amount of explained variances by the selected dimensions. Unless the sum of these explained variances is sufficiently large, "the interpretation of the plot is suspect" (Jackson 1991, 199). However, there is no known boundary below which the interpretation is erroneous. We have found explained variances of about 70% enough to obtain good approximations of the key quantities for small datasets.

Choosing a value for c defines the coordinates for different types of biplots. Three values for c are most commonly used and are therefore implemented in `biplot8`:

- $c = 0$, the GH, or column-metric preserving biplot
- $c = 1$, the JK, or row-metric preserving biplot
- $c = .5$, the SQ, or symmetric biplot

GH biplots are called *column-metric preserving* because the variance–covariance structure of the variables is best approximated in the GH biplot. JK biplots, on the other hand, are *row-metric preserving*, since the approximations of the Euclidean distances are optimal in this biplot. Finally, the SQ biplots represent the observational values of **Y** better than the other types.

4 Computational issues

The Stata command to calculate a singular value decomposition is

```
. matrix svd U L V = Y
```

where `Y` is the name of the matrix that ought to be decomposed and `U`, `L`, and `V` are arbitrary names for the resulting matrices of the SVD. To calculate the coordinates of the biplot, this command requires that the complete data matrix be stored in `Y`. The maximum dimension of a single matrix in Intercooled Stata is 800×800. In Intercooled Stata, the SVD of a data matrix therefore can be only done for datasets with up to 800 observations. In Stata/SE, this limit is raised to 11,000 observations. Given that there is no general maximum number of observations in Stata, the maximum number of observations to be used in a biplot is restrictive[2].

[2] In Stata 9, these limitations can be circumvented using Mata; see the *Mata Reference Manual* for details.

In the case of the JK biplot ($c = 1$), the restriction can be circumvented. As Jolliffe (2002, 94–95) shows, the elements in $\mathbf{G}$ are equal to the respective values of the observations on the principal components. Accordingly, the elements in $\mathbf{H}$ are equal to the coefficients (loadings) of a PCA. Therefore, the coordinates of the JK biplot can be easily calculated from a PCA, bypassing the calculation of the SVD. To this extent, the biplot with $c = 1$ is nothing new since the component score plot and the plot of PCA coefficients are widely used on their own. The superimposing of both plots, however, gives additional information.

The possibility that you can calculate the plot coordinates by means of a PCA for the JK biplot raises the question whether this is also possible for the other biplot types. In fact, the coordinates of the JK biplot and the GH biplot are closely related. It follows from the definition of both biplots and from (1) and (2) that

$$\begin{array}{lcl} (\mathbf{G}_{\mathrm{JK}} = \mathbf{UL} \quad \wedge \quad \mathbf{G}_{\mathrm{GH}} = \mathbf{U}) & \Rightarrow & \mathbf{G}_{\mathrm{JK}} = \mathbf{G}_{\mathrm{GH}}\mathbf{L} \\ (\mathbf{H}'_{\mathrm{JK}} = \mathbf{V}' \quad \wedge \quad \mathbf{H}'_{\mathrm{GH}} = \mathbf{LV}') & \Rightarrow & \mathbf{H}'_{\mathrm{GH}} = \mathbf{LH}'_{\mathrm{JK}} \end{array}$$

Therefore, the coordinates of the JK biplot can be transformed into the coordinates of the GH biplot with

$$\mathbf{G}_{\mathrm{GH}} = \mathbf{G}_{\mathrm{JK}}\mathbf{L}^{-1} \tag{3}$$

$$\mathbf{H}'_{\mathrm{GH}} = \mathbf{LH}'_{\mathrm{JK}} \tag{4}$$

The SVD, however, is still needed to calculate $\mathbf{L}$. At the same time, it is possible to calculate the eigenvalues in $\mathbf{L}$ by transforming the eigenvalues of a PCA ($\mathbf{L}_{\mathrm{JK}}$) as shown below[3]:

$$\mathbf{L} = \mathbf{U}'\mathbf{Y}_{\mathrm{S}}\mathbf{S}^{-1}\mathbf{U}_{\mathrm{S}}\mathbf{L}_{\mathrm{JK}} \tag{5}$$

where $\mathbf{S}$ is the covariance matrix of the centered data matrix and $\mathbf{U}_{\mathrm{S}}$ are the coefficients of a PCA. Unfortunately, to get $\mathbf{U}$, it is again necessary to calculate the SVD of $\mathbf{Y}$, which once more restricts the maximum number of observations to be used.

Right now, you cannot circumvent the restriction on the maximum number of observations for the GH or SQ biplot. In the future, it might be worthwhile for StataCorp to program the calculation of the eigenvalues from the dataset without storing the dataset in a matrix beforehand. In this case, at least the GH biplot could be easily derived from a PCA with (3) and (4).

From a practical point of view, the described restriction is not as restrictive as it sounds. It has been already stated that the interpretation of the biplot will be suspect if the variance explained by the dimensions of the biplot are small. Small explained variances, however, are quite common in working with datasets with many observations. To this extent, the biplot has its strength mainly for datasets with small to moderate number of observations. For huge datasets, the JK biplot can be calculated in any case.

[3] The derivation of this formula can be found in the appendix.

5 The biplot8 command

5.1 Syntax

`biplot8` *varlist* [*weight*] [`if` *exp*] [`in` *range*] [`,`
[`jk|sq|gh|mixed(jk|sq|gh jk|sq|gh)`] `covariance mahalanobis rv`
[`obsonly|varonly`] `dimensions(`*##*`)` `generate(`*name1* [*name2*]`)`
`subpop(`*varname* [`,` *scatter_options*]`)` `stretch(`*#*`)` `flip(x|y|xy)`
scatter_options line_options twoway_options]

`aweights` and `fweights` are allowed; see [U] **11.1.6 weight**. However, no weights are allowed with option `rv`, and `aweights` are not allowed with options `sq` and `gh`.

5.2 Options

`jk|sq|gh` specifies the biplot type. `jk` specifies the default, a JK biplot. `gh` and `sq` specifies GH and SQ biplots, respectively (see section 5.4).

`mixed(jk|sq|gh jk|sq|gh)` can be used instead of the biplot types to combine the relative advantages of the different biplot types. Inside the parentheses, you first state the type for the observations and then a type for the variables (see section 5.4).

`covariance` is used to plot the unstandardized data matrix. The default is standardization (see section 5.4).

`mahalanobis` can be used for GH biplots to rescale the graph in a way that the distances between the observations approximate the Mahalnobis distances (see section 5.4).

`rv` is used to produce relative variation diagrams (see section 5.4).

`obsonly|varonly` are used to suppress the plotting of observations or variables, respectively (see section 5.3).

`dimensions(`*##*`)` is used to specify the space in which the variables and observations are drawn. The default is to use the dimension with the highest eigenvalues (i.e., the first two principal components for JK biplots) (see section 5.3).

`generate(`*name1* [*name2*]`)` is used to store the coordinates for the observations and the variables as variables in the dataset. The y-axis coordinates for the observations are stored in *name1*`_y`, and the x-axis coordinates for the observations are stored in *name1*`_x`. Accordingly, the coordinates for the variables are stored in *name2*`_y` and *name2*`_x`.

`subpop(`*varname)* is used to highlight observations from different subpopulations with different marker symbols (see section 5.5).

`stretch(`*#*`)` draws longer (or if needed shorter) lines for the variables. By default, `stretch()` is set to a value that improves readability (see section 5.3).

`flip(x|y|xy)` exchanges the signs of the axes. `flip(x)` and `flip(y)` exchange signs of the indicated axis, `flip(xy)` flips both axes. `flip()` is seldom used but might be useful if you want to compare your results with the results of other software packages.

scatter_options are the following options allowed with `twoway scatter`.

`jitter(`*relativesizelist*`)`	add spherical random noise to plot symbols
`msymbol(`*symbolstylelist*`)`	shape of marker
`mcolor(`*colorstylelist*`)`	color of marker, inside and out
`msize(`*markersizestylelist*`)`	size of marker
`mlabel(`*varlist*`)`	specify marker variables
`mlabposition(`*clockposlist*`)`	where to locate label
`mlabvposition(`*varname*`)`	where to locate label 2
`mlabgap(`*relativesizelist*`)`	gap between marker and label
`mlabsize(`*textsizestylelist*`)`	size of label
`mlabcolor(`*colorstylelist*`)`	color of label

Up to two elements are allowed for each option. The first element refers to the display of the observations, and the second element refers to the variables. Note that the default plot symbol for the position of the variables is invisible; that is, the default value for msymbol is `msymbol(oh i)`. The lines for the variables are, however, changed with the *line_options*.

line_options are the following set of the options allowed with `line`. Note that the *line_options* only refer to the display of the variable lines.

`clpattern(linepatternstylelist)`	whether line is solid, dashed, etc.
`clwidth(linewidthstylelist)`	thickness of line
`clcolor(colorstylelist)`	color of line

twoway_options are those options allowed with `graph twoway`; see [G] ***twoway_options***.

5.3 JK biplot and common PCA plots

Invoking the command `biplot8` with a *varlist* and no other options brings up a JK biplot (figure 3).[4]

(*Continued on next page*)

[4] The examples in this section use the `iris` dataset. The data contains the sepal length, sepal width, petal length, and petal width of 150 flowers from the iris species *setosa*, *versicolor*, and *virginica*. It was collected by Anderson (1935) and was used by Fisher (1936) in his initiation of the linear-discriminant-function technique.

```
. biplot8 sepallen-petalwid
```

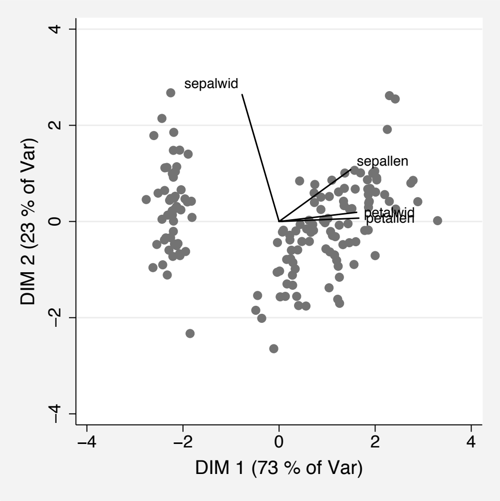

Figure 3: The standard JK biplot of `iris.dta`

As stated above, the JK biplot superimposes two of the most-often described plots for principal component analysis: the component score plot and the plot of the PCA coefficients. However, in the default setting of the command `biplot8`, there is a difference between the variable lines of the JK biplot and the plot of the PCA coefficients. The `biplot8` command stretches the variable lines to optimally fill the plot region given by the observations (Digby and Kempton 1987, section 3.2). The positions of the variable lines along the graph axis therefore represent the relative sizes of the PCA coefficients, as opposed to the absolute ones, used in the plot of PCA coefficients. High values still represent high "loadings", but the square of the loadings cannot be interpreted as communalities, as is the case for the plot of PCA coefficients.

It is, however, still possible to use `biplot8` as a means to produce the plot of PCA coefficients and the component score plot. The plot of PCA coefficients can be produced with the options `stretch(#)` and `varonly`. In the former option, # stands for a number by which the length of the variable lines are multiplied. By default, `biplot8` automatically chooses this stretch factor to ensure optimal readability. Setting the stretch factor to 1 forces Stata to use the original values, which are the PCA coefficients in the case of the JK biplot. Using the option `varonly`, in addition, suppresses the display of the observations entirely and thereby sets the graph scales according to the coordinates of the variables. This brings up the plot of the PCA coefficients (figure 4).

```
. biplot8 sepallen-petalwid, st(1) varonly
```

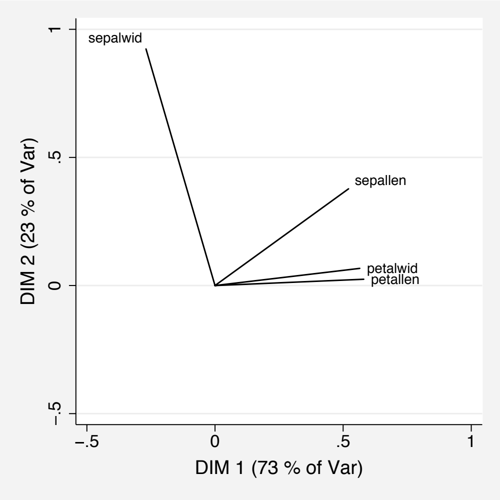

Figure 4: Plot of PCA coefficients

Accordingly, the option `obsonly` as used in

```
. biplot8 sepallen-petalwid, obsonly
```

brings up the component score plot (figure 5).

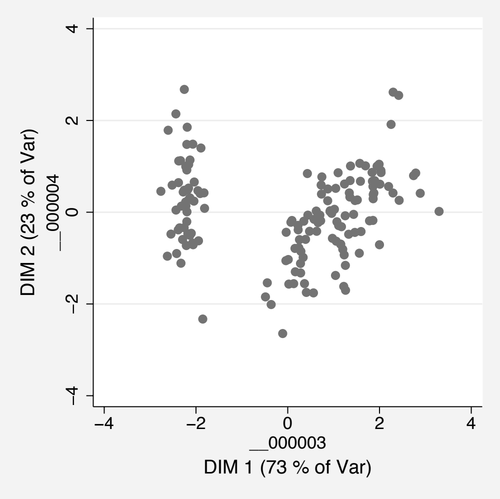

Figure 5: Component score plot

As shown in section 3, the data coordinates of the biplot have k dimensions. To plot these coordinates in a two-dimensional graph, you must select the dimensions to be plotted. By default, this is done by selecting those coordinates that refer to the two highest eigenvalues. The option `dimensions(##)` allows you to change this. Inside the parentheses, you can state the ordinal rank of the eigenvalue for which the coordinates ought to be selected. This is useful for JK biplots since you might be interested in a display of the PCA coefficients for arbitrary principal components. Moreover, the component score plot in the space of the two last principal components is said to show a special kind of outlier (Gnanadesikan 1977, 261). Such a plot can be produced with

```
. biplot8 sepallen-petalwid, dim(3 4)
```

5.4 Biplot types and variations

The JK, GH, and SQ biplot can be displayed by using the options `jk`, `gh`, or `sq`, respectively. It is possible in any case to calculate the coordinates from a standardized or a nonstandardized data matrix. By default, `biplot8` standardizes the data matrix, which is why the variable lines tend to have the same length. To get lengths for the variable lines according to variances of the variables, the option `covariance` must be used. Figure 6 gives an example of the GH biplot for the nonstandardized data matrix, which has been produced with the following command:

```
. biplot8 sepallen-petalwid, gh cov
```

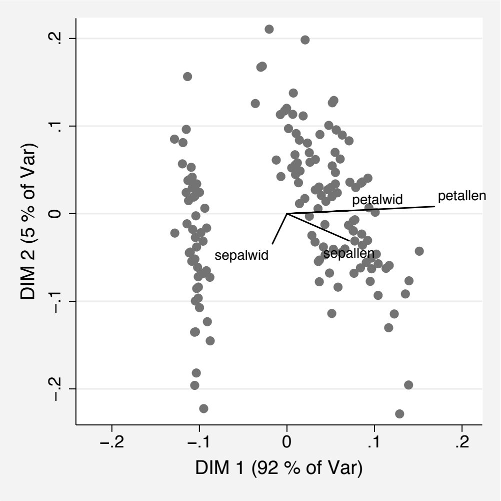

Figure 6: GH biplot for unstandardized data

As mentioned in section 3, the biplot types differ in the quality of the approximations of the key quantities shown in a biplot. While the approximation of the Euclidean distance is best represented in the JK biplot, the variance–covariance structure is better represented in the GH biplot. It seems, therefore, relatively straightforward to mix the different biplot types. Gabriel (2002), for example, proposed a “correspondence analysis” that uses the coordinates of a GH biplot for the variables and the coordinates of a JK biplot for the observations. Such mixed biplots can be produced with the option `mixed()`. The option allows you to list the names of two biplot types inside the parentheses. The first name refers to the observational part, and the second refers to the variable part. To obtain Gabriel’s correspondence analysis, you might type

```
. biplot8 sepallen-petalwid, mixed(jk gh)
```

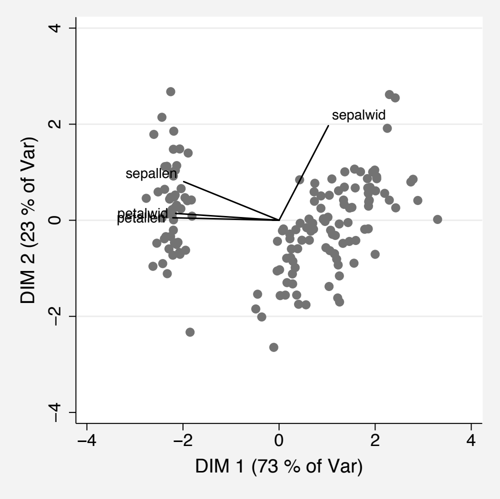

Figure 7: Gabriel’s correspondence analysis

Note, however, that while it is possible to give optimal approximations to two of the quantities shown in a biplot, this is not possible for all three of them (Gower and Hand 1996; Gabriel 2002). Mixing the GH and JK biplot as in the example above does not optimally represent the observational values.

A further variant is biplots for compositional data. Compositional data are datasets with constant row sums and only positive values, e.g., row percentages of contingency tables. The standard data analysis techniques of compositional data usually tends to be misleading, and therefore a set of specialized techniques are available for such data (Aitchison 1986). The equivalent to biplots for compositional data is the “relative variation diagram” (RV plot) (Aitchison 1990). A relative variation diagram refers to a biplot of a transformed data matrix. The transformation is

$$y^*_{ik} = \ln y_{ik} - \overline{y}_i - \overline{y}_k$$

with y_{ik} being the untransformed value of $\mathbf{Y}$ in the ith row and kth column and $\overline{y}_i$ and $\overline{y}_k$ being the row and column means of the data matrix. The option `rv` forces Stata to make this transformation before producing the biplot.

Finally, the option `mahalanobis` can be used to rescale the coordinates in **G** and **H** by

$$\begin{aligned} \mathbf{G}^* = &\quad \mathbf{G} \times \sqrt{n} \\ \mathbf{H}^* = &\quad \mathbf{H} \times \tfrac{1}{\sqrt{n}} \end{aligned}$$

before producing the biplot. According to Gabriel (1971) the resulting biplot reflects the Mahalanobis distances between the observations instead of the Euclidean distances.

5.5 Options to control the graph appearance

Several options are available for controlling the appearance of the graph. Among them are most of the options allowed for `twoway scatter` and `twoway line`. Here *scatter_options* allow up to two arguments, where the first argument refers to the observations (the dots) and the second refers to the points at the end of the variable lines (which are invisible by default). *line_options* refer to the variable lines.

The option `subpop()` is specific to `biplot8` and is used to distinguish observations from different subgroups with different markers. Therefore, the name of the variable that identifies the subgroup is placed inside the parentheses. Note that the *scatter_options* for the observations are ignored if you specify `subpop()`. However, you can use the complete set of *scatter_options* as suboptions within `subpop()` to control the appearance of the observations.

The `subpop()` option is especially useful for illustrating the substantial meaning of data clusters. Figure 8, which has been produced with the command below, gives an illustrative example.

(*Continued on next page*)

```
. biplot8 sepallen-petalwid, subpop(species, msymbol(Oh X Th))
> legend(ring(0) pos(4))
```

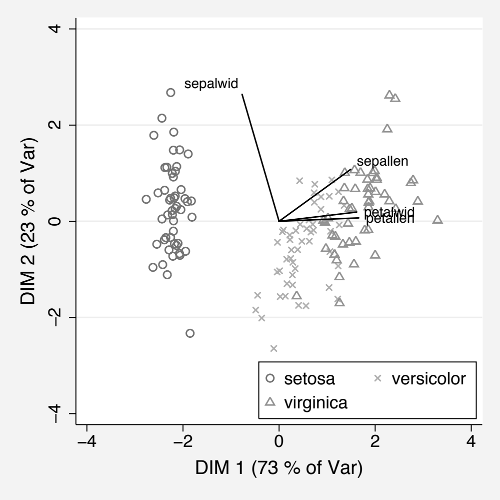

Figure 8: Illustrative example of representation options

Note that the default positioning of legends changes the aspect ratio of the biplot. If you don't like this, you can move the legend position to the inner ring, as shown in the example. Alternatively, you can turn the legend off or refine the aspect ratio with the options `xsize()` or `ysize()`.

6 Appendix

Consider a PCA of the data matrix $\mathbf{Y}$, which is a SVD of the variance–covariance matrix $\mathbf{S}$ of $\mathbf{Y}$

$$\mathbf{S} = \mathbf{U}_{\mathrm{S}}\mathbf{L}_{\mathrm{JK}}\mathbf{V}'_{\mathrm{S}} \tag{6}$$

Also consider the coordinates of the observations for the JK biplot from (1):

$$\mathbf{G}_{\mathrm{JK}} = \mathbf{UL} \tag{7}$$

From Jolliffe (2002, 94), it is known that $\mathbf{G}_{\mathrm{JK}}$ are equal to the scores of the observations on the principal components, which are given by

$$\mathbf{G}_{\mathrm{JK}} = \mathbf{YU}_{\mathrm{S}} \tag{8}$$

From (7) and (8), we obtain

$$\begin{aligned} \mathbf{UL} &= \mathbf{YU}_{\mathrm{S}} \\ \mathbf{U}'\mathbf{UL} &= \mathbf{U}'\mathbf{YU}_{\mathrm{S}} \end{aligned}$$

From the properties of the SVD, we know that $\mathbf{U}$ is an unitary matrix, so $\mathbf{U}'\mathbf{U} = \mathbf{I}$. Hence

$$\mathbf{L} = \mathbf{U}'\mathbf{Y}\mathbf{U}_{\mathrm{S}} \tag{9}$$

In order to find the relation between $\mathbf{L}$ and $\mathbf{L}_{\mathrm{JK}}$, we look at $\mathbf{U}_{\mathrm{S}}$ from (6). The matrix $\mathbf{S}$ is symmetric, so

$$\begin{aligned} \mathbf{U}'_{\mathrm{S}} &= \quad \mathbf{V}'_{\mathrm{S}} \\ \mathbf{S} &= \quad \mathbf{U}_{\mathrm{S}}\mathbf{L}_{\mathrm{JK}}\mathbf{U}'_{\mathrm{S}} \\ \mathbf{S}\mathbf{U}_{\mathrm{S}} &= \quad \mathbf{U}_{\mathrm{S}}\mathbf{L}_{\mathrm{JK}}\mathbf{U}'_{\mathrm{S}}\mathbf{U}_{\mathrm{S}} \end{aligned}$$

$\mathbf{U}_{\mathrm{S}}$ is orthogonal, which means that $\mathbf{U}'_{\mathrm{S}} = \mathbf{U}_{\mathrm{S}}^{-1}$ and $\mathbf{U}'_{\mathrm{S}}\mathbf{U}_{\mathrm{S}} = \mathbf{U}_{\mathrm{S}}\mathbf{U}'_{\mathrm{S}} = \mathbf{I}$. Hence

$$\begin{aligned} \mathbf{S}\mathbf{U}_{\mathrm{S}} &= \quad \mathbf{U}_{\mathrm{S}}\mathbf{L}_{\mathrm{JK}} \\ \mathbf{U}_{\mathrm{S}} &= \quad \mathbf{S}^{-1}\mathbf{U}_{\mathrm{S}}\mathbf{L}_{\mathrm{JK}} \end{aligned} \tag{10}$$

Imposing (10) in (9) gives (5) on page 213:

$$\mathbf{L} = \mathbf{U}'\mathbf{Y}\mathbf{S}^{-1}\mathbf{U}_{\mathrm{S}}\mathbf{L}_{\mathrm{JK}}$$

7 Acknowledgments

We like to point German readers to the book *Graphisch gestützte Datenanalyse*, written by Rainer Schnell (1994, 176–186), upon which parts of this article are heavily based. We also like to thank Frauke Kreuter for carefully reading an earlier draft of this article and Vince Wiggins and Nick Cox for helping us write `biplot8`.

8 References

Aitchison, J. 1986. *The Statistical Analysis of Compositional Data.* London: Chapman & Hall.

—. 1990. Relative variation diagrams for describing patterns of compositional variablity. *Mathematical Geology* 22: 487–512.

Anderson, E. 1935. The irises of the Gaspé peninsula. *Bulletin of the American Iris Society* 59: 2–5.

Blasius, J. and M. Greenacre, ed. 1998. *Visualization of Categorical Data.* San Diego: Academic Press.

Digby, P. G. N. and R. A. Kempton. 1987. *Multivariate Analysis of Ecological Communities.* London: Chapman and Hall.

Fenty, J. 2004. Analyzing distances. *Stata Journal* 4(1): 1–26.

Fisher, R. A. 1936. The use of multiple measurements in taxonomic problems. *Annals of Eugenics* 7: 179–188.

Gabriel, K. 1971. The biplot graphic display of matrices with application to principal component analysis. *Biometrika* 58(3): 453–467.

—. 2002. Goodness of fit of biplots and correspondence analysis. *Biometrica* 89(2): 423–436.

Gnanadesikan, R. 1977. *Methods for Statistical Data Analysis of Multivariate Observations*. New York: Wiley.

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

Hamilton, L. C. 1992. *Regression with Graphics: A Second Course in Applied Statistics*. Pacific Grove, CA: Brooks/Cole.

Jackson, J. E. 1991. *A User's Guide to Principal Components*. New York: Wiley.

Jolliffe, I. 2002. *Principal Component Analysis*. 2nd ed. New York: Springer.

Kruskal, J. B. and M. Wish. 1978. *Multidimensional Scaling*. Beverly Hills, CA: Sage.

Schnell, R. 1994. *Graphisch gestützte Datenanalyse*. München, Wien: Oldenbourg.

Schnell, R. and H. Matschinger. 1994. Multivariate graphics: Current use and implementations in the social sciences. In *Computational Statistics. Papers Collected on the Occasion of the 25th Conference on Statistical Computing at Schloß Reisensburg*, ed. P. Dirschedl and R. Ostermann, 275–294. Heidelberg: Physica.

Tabachnik, B. and L. S. Fidell. 1989. *Using Multivariate Statistics*. 2nd ed. New York: Harper and Row.

About the Authors

Ulrich Kohler is a sociologist at the Wissenschaftszentrum Berlin (Social Science Research Center) who has used Stata for several years. His research interests include social inequality and political sociology. With Frauke Kreuter, he is author of the upcoming textbook *Data Analysis Using Stata*.

Magdalena Luniak studies sociology at the Warsaw University and IT at the TU Berlin. The main focus of her studies is the application of mathematics in sociology.