# BIAREG

# Splus functions to perform Biadditive Regressions

**Jean-Baptiste DENIS**

**Rapport technique 1998-2**

**Unité de biométrie**
**INRA**
**route de Saint-Cyr**
**F-78026 Versailles Cedex**

# BIAREG

## Splus functions to perform Biadditive Regressions

## (version 1.0)

Jean-Baptiste DENIS
Biométrie
INRA
route de Saint-Cyr
F-78026 VERSAILLES
*denis@versailles.inra.fr*

February 25, 1998 /BiaReg01/

# Contents

# 1 Introduction

Based on a bilinear modelling, biadditive models are successfully used for explaining interactions between two factors having a high number of levels (Denis & Gower, 1996). The same purpose is attempted by factorial regressions (Denis, 1988) when covariates associated with the factors are available. These latter give more insight for the interpretation of the interaction, but at the cost of numerous linear parameters. Eeuwijk *et al.* (1996) in their review paper underlined that both approaches could be mixed to produce bilinear models built on factor covariates that we will designate here as **biadditive regressions**. A general presentation of these models can be found in Denis (1991).

Here we would like to explain the use of some SPLUS functions devoted to the fitting of such models from data sets. The functions have been gathered under the generic name of BIAREG, they are available under request from the author. They can be viewed as an extension of the INTERA package (Decoux & Denis, 1990) where covariates could intervene only in the linear part of the models. On the other hand this first version of BIAREG is restricted to complete data sets, future functions will be added to reduce this limitation.

No statistical theory is produced here. Just some reminders are given of the models to introduce the notation and to exhibit the spectrum of covered models. The aim is to show how to use the SPLUS functions. Nevertheless, it is supposed that the reader has a minimal ability in SPLUS.

# 2 Data sets

Basically, the models considered are based on three data matrices, the last two being possibly implicit.

## 2.1 Y : two-way table to interpret

We suppose that we are interested in the effects of two factors on a given variate. The basic information is given by $\mathbf{Y}$, a two-way table whose rows correspond to the levels of the first factor and columns to the levels of the second factor and $Y_{ij}$ is the averaged result of the variate for the combination $(i, j)$.

$$\mathbf{Y} = (Y_{ij}) \qquad i = 1, ..., I \text{ and } j = 1, ..., J.$$

Usually, the two factors will be referred to as a row factor and a column factor.

## 2.2 X : row covariates

For each level of the row factor, the values of some covariate(s) are supposed known. These values are gathered in a data matrix $\mathbf{X}$ whose rows are associated with the levels of the row factor and whose $K$ columns correspond to the $K$ different covariates. In most cases, the constant covariate is introduced as the first covariate[1].

$$\mathbf{X} = (X_{ik}) \qquad i = 1, ..., I \text{ and } k = 1, ..., K.$$

In the model, covariates can be used either linearly or bilinearly. To distinguish the two possibilities, we will add a superscript to the covariate matrix:

$$\mathbf{X}^L = \left(X_{ik}^L\right) \qquad i = 1, ..., I \text{ and } k = 1, ..., K^L \qquad \text{for the linear part,}$$

$$\mathbf{X}^B = \left(X_{ik}^B\right) \qquad i = 1, ..., I \text{ and } k = 1, ..., K^B \qquad \text{for the bilinear part.}$$

Some conditions are assumed concerning the redundancy of the covariates.

1. The linearly used covariates are supposed to be noncollinear *i.e.* that $\text{rk}\left(\mathbf{X}^L\right) = K^L$. Not imposing such a constraint would lead to the need for arbitrary identification constraints as in standard multiple linear regression when regressors are not linearly independent.
2. similarly for the bilinearly used covariates, *i.e.* that $\text{rk}\left(\mathbf{X}^B\right) = K^B$,
3. but $\text{rk}\left(\mathbf{X}^L, \mathbf{X}^B\right)$ can be less than $K^L + K^B$, that is some redundancy may exist between the two subsets of covariates. This requirement was necessary to describe under the same pattern classical biadditive models as will be seen below. The consequence is that additional constraints on the parameters are needed. The foundations of the chosen constraints are the orthogonalisation of the model terms, bilinear terms being adjusted from linear terms.

## 2.3 Z : column covariates

Symmetrically, for each level of the column factor, the values of some covariate(s) are supposed known. These values are gathered in a data matrix **Z** whose rows are associated to the levels of the column factor and whose $H$ columns correspond to the $H$ different covariates. In most cases, the constant covariate is introduced as first covariate.

$$\mathbf{Z} = (Z_{jh}) \qquad j = 1, ..., J \text{ and } h = 1, ..., H.$$

Similarly to the row covariates, we will distinguish

$$\mathbf{Z}^L = \left(Z_{jh}^L\right) \qquad j = 1, ..., J \text{ and } h = 1, ..., H^L \qquad \text{for the linear part,}$$

$$\mathbf{Z}^B = \left(Z_{jh}^B\right) \qquad j = 1, ..., J \text{ and } h = 1, ..., H^B \qquad \text{for the bilinear part.}$$

Likewise, it is assumed that

$$\begin{aligned} \text{rk}\left(\mathbf{Z}^L\right) &= H^L, \\ \text{rk}\left(\mathbf{Z}^B\right) &= H^B, \\ \text{rk}\left(\mathbf{Z}^L, \mathbf{Z}^B\right) &\leqslant H^L + H^B. \end{aligned}$$

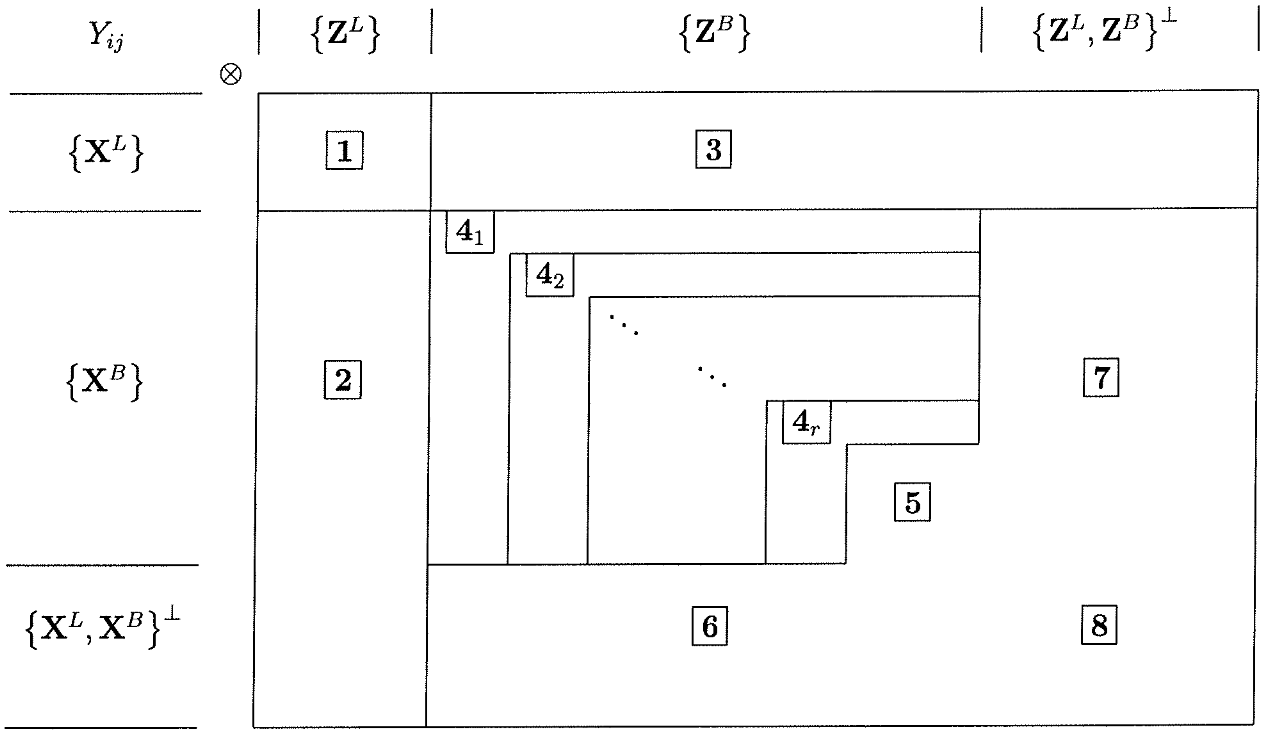
[1] this is the way to introduce main effects in the factorial regression scheme (see examples). Constant covariate can be the single covariate introduced in the model.

# 3 Models

For the sake of the notation, it was thought better to present first the general formulation and to retrieve for each particular model the corresponding application. Nevertheless we advise the reader not at ease with mathematical formulæ to skip the next subsection.

## 3.1 general formulation

Biadditive regression models are fixed models and can be presented by the mathematical expectation of matrix $\mathbf{Y}$. Following the factorial nature of the models, these expectations will be presented by recapitulative tables (see Eeuwijk *et al.*, 1996) where rows are associated to the decomposition of the row factor effects and the columns are associated with the decomposition of the column effects. This presentation was proposed in Figure 3 by Denis (1991), it emphasizes the fact that biadditive regression models are completely defined by the four subsets of previously defined covariates and the number of multiplicative terms. The general model, with $r$ multiplicative terms, can be displayed as follows:




where

$$\boxed{1} = \sum_{k=1}^{K^L} \sum_{h=1}^{H^L} \nu_{kh} X_{ik}^L Z_{jh}^L \qquad \left[\mathbf{X}^L \boldsymbol{\nu} \left(\mathbf{Z}^L\right)^T\right]$$

$$\boxed{2} = \sum_{h=1}^{H^L} \rho_{ih} Z_{jh}^L \qquad \left[\boldsymbol{\rho} \left(\mathbf{Z}^L\right)^T\right]$$

$$\boxed{3} = \sum_{k=1}^{K^L} \tau_{jk} X_{ik}^L \qquad \left[\mathbf{X}^L \boldsymbol{\tau}\right]$$

$$
\begin{aligned}
\boxed{4_1} &= \left(\sum_{k=1}^{K^B} \gamma_{k1} X_{ik}^B\right) \theta_1 \left(\sum_{h=1}^{H^B} \delta_{h1} Z_{jh}^B\right) \qquad \left[\left(\mathbf{X}^B \boldsymbol{\gamma}_1\right) \theta_1 \left(\mathbf{Z}^B \boldsymbol{\delta}_1\right)^T\right] \\
\boxed{4_2} &= \left(\sum_{k=1}^{K^B} \gamma_{k2} X_{ik}^B\right) \theta_2 \left(\sum_{h=1}^{H^B} \delta_{h2} Z_{jh}^B\right) \qquad \left[\left(\mathbf{X}^B \boldsymbol{\gamma}_2\right) \theta_2 \left(\mathbf{Z}^B \boldsymbol{\delta}_2\right)^T\right] \\
& \dots \\
\boxed{4_r} &= \left(\sum_{k=1}^{K^B} \gamma_{kr} X_{ik}^B\right) \theta_r \left(\sum_{h=1}^{H^B} \delta_{hr} Z_{jh}^B\right) \qquad \left[\left(\mathbf{X}^B \boldsymbol{\gamma}_r\right) \theta_r \left(\mathbf{Z}^B \boldsymbol{\delta}_r\right)^T\right] \\
\boxed{5} &= \text{the remaining part of the common linear term with bilinear covariates} \\
\boxed{6} &= \text{the remaining part of the linear terms with bilinear column covariates} \\
\boxed{7} &= \text{the remaining part of the linear terms with bilinear row covariates} \\
\boxed{8} &= \text{the remaining part of all the covariates}
\end{aligned}
\tag{1}
$$

This presentation is based on the distinction between the linear and the bilinear terms. Most model users would have preferred a distinction between main effect and interactive terms. It is not done this way to follow the algebraical structure of the models on which is built the program. Nevertheless, it is not too difficult to introduce the classical distinction on the algebraical scheme. To do so, we need to suppose that the first row and column linearly used covariates are the constant covariates: $\mathbf{1}_I$ and $\mathbf{1}_J$. Then, two new splitting of rows and columns of the recapitulative table can be made to identify with the linear part, the constant term and both main effects. This is exemplified in the next subsection.

## 3.2 examples

In the following examples, some classical and more novel models are proposed as particular applications of Model (1).

### 3.2.1 additive model

The usual additive model

$$\mu + \alpha_i + \beta_j \tag{2}$$

is obtained by simply taking

$$
\begin{aligned}
\mathbf{X}^L &= \mathbf{1}_I \text{ and } \mathbf{X}^B = \emptyset, \\
\mathbf{Z}^L &= \mathbf{1}_J \text{ and } \mathbf{Z}^B = \emptyset,
\end{aligned}
$$

where $\mathbf{1}_n$ is the $n$-vector of ones. To retrieve general Formulation (1), one can explicitly introduce the constant covariates, indeed

$$\mu + \alpha_i + \beta_j = 1\mu 1 + \alpha_i 1 + 1\beta_j = X_{i1}\mu Z_{j1} + \alpha_i Z_{j1} + X_{i1}\beta_j$$

when $X_{i1} = Z_{j1} = 1$. The parametrical dimension of the model (standard degrees of freedom for linear models) is $I + J - 1$ for a total of $I + J + 1$ parameters.

### 3.2.2 biadditive models

The usual biadditive models with $r$ multiplicative terms read

$$\mu + \alpha_i + \beta_j + \gamma_{i1}\theta_1\delta_{j1} + \gamma_{i2}\theta_2\delta_{j2} + ... + \gamma_{ir}\theta_r\delta_{jr} \tag{3}$$

and are obtained by taking

$$\begin{aligned} \mathbf{X}^L &= \mathbf{1}_I \text{ and } \mathbf{X}^B = \mathbf{I}_I \\ \mathbf{Z}^L &= \mathbf{1}_J \text{ and } \mathbf{Z}^B = \mathbf{I}_J \end{aligned}$$

where $\mathbf{I}_n$ is the identity matrix of size $n$. The trick in making use of the identity matrix as a covariate matrix must be explained. Let us consider the case of the row factor:

- In fact the $u$th column of the identity matrix $\mathbf{I}_I$ is a vector of zeros except a one in position $u$. As a covariate this column is the indicator of the $u$th level of the row factor.
- when taking the multiplicative term described in (1), it turns out that the first multiplicative term is

$$\left(\sum_{k=1}^{K^B} \gamma_{k1} X^B_{ik}\right) = \left(\sum_{k=1}^{I} \gamma_{k1} \mathbf{1}_{[i=k]}\right) = \gamma_{i1}.$$

The parametrical dimensions of these models are $(1+r)(I+J-(1+r))$ for $(1+r)(I+J+1)$ parameters.

### 3.2.3 factorial regressions

Factorial regression models are simply obtained by introducing only "linearly used covariates", for instance

$$\mu + \alpha_i + \beta_j + X_i \nu Z_j + \rho_i Z_j + X_i \tau_j \tag{4}$$

is obtained by taking

$$\begin{aligned} \mathbf{X}^L &= (\mathbf{1}_I, X) \text{ and } \mathbf{X}^B = \emptyset, \\ \mathbf{Z}^L &= (\mathbf{1}_J, Z) \text{ and } \mathbf{Z}^B = \emptyset. \end{aligned}$$

This model can be generalized with $(K-1) < I$ row covariates and $(H-1) < J$ column covariates:

$$\mu + \alpha_i + \beta_j + \sum_{k=2}^{K}\sum_{h=2}^{H} X_{ik}\nu_{kh}Z_{jh} + \sum_{h=2}^{H} \rho_{ih}Z_{jh} + \sum_{k=2}^{K} X_{ik}\tau_{jk} \tag{5}$$

by taking

$$\begin{aligned} \mathbf{X}^L &= (\mathbf{1}_I, X_2, ..., X_K) \text{ and } \mathbf{X}^B = \emptyset, \\ \mathbf{Z}^L &= (\mathbf{1}_J, Z_2, ..., Z_H) \text{ and } \mathbf{Z}^B = \emptyset. \end{aligned}$$

The parametrical dimensions of these models are $HI+KJ-HK$ for $HI+KJ+HK$ parameters.

The reader could have noticed that "true" covariates are numbered from 2 to $K$ (and $H$) to include the constant covariate implicitly associated with main effects. When $X_{i1} = 1$ and $Z_{j1} = 1$

$$X_{i1}\nu_{11}Z_{j1} + \rho_{i1}Z_{j1} + X_{i1}\tau_{j1} = \nu_{11} + \rho_{i1} + \tau_{j1} \equiv \mu + \alpha_i + \beta_j.$$

#### 3.2.4 INTERA models

Models fitted by the program REGFAM of INTERA are factorial regressions complemented by free multiplicative terms. For instance

$$\mu + \alpha_i + \beta_j + X_i \nu Z_j + \rho_i Z_j + X_i \tau_j + \gamma_i \theta \delta_j \tag{6}$$

is obtained by taking

$$\begin{aligned} \mathbf{X}^L &= (\mathbf{1}_I, X) \text{ and } \mathbf{X}^B = \mathbf{I}_I \\ \mathbf{Z}^L &= (\mathbf{1}_J, Z) \text{ and } \mathbf{Z}^B = \mathbf{I}_J \end{aligned}$$

and $r = 1$. The parametrical dimension of this model is $3(I + J - 3)$ for $3(I + J + 3)$ parameters.

#### 3.2.5 bilinear regression models

Parsimonious models can be obtained by involving true[2] covariates only in the bilinear part. Let us consider the following model

$$\mu + \alpha_i + \beta_j + \left(\sum_{k=1}^{K^B} \gamma_{k1} X_{ik}\right) \theta_1 \left(\sum_{h=1}^{H^B} \delta_{h1} Z_{jh}\right) + \left(\sum_{k=1}^{K^B} \gamma_{k2} X_{ik}\right) \theta_2 \left(\sum_{h=1}^{H^B} \delta_{h2} Z_{jh}\right) \tag{7}$$

obtained by taking

$$\begin{aligned} \mathbf{X}^L &= \mathbf{1}_I \text{ and } \mathbf{X}^B = (X_1, X_2, ..., X_{K^B}), \\ \mathbf{Z}^L &= \mathbf{1}_J \text{ and } \mathbf{Z}^B = (Z_1, Z_2, ..., Z_{H^B}), \end{aligned}$$

and $r = 2$. It is defined by $(I + J - 1) + 2\left(K^B + H^B - 2\right)$ free parameters meanwhile Model (5) needs $(I + J - 1) + (JK + IH - HK)$ free parameters; it comprises a total of $(I + J + 1) + 2\left(K^B + H^B + 2\right)$ parameters.

#### 3.2.6 linear structuration models

As developed by Eeuwijk *et al.* (1996), covariates can be qualitative as well. Let us imagine that the rows of $\mathbf{Y}$ are clustered into 3 distinct groups defined by a function $s(i)$ taking values in $\{1, 2, 3\}$. In such a case an interesting model could be

$$\mu_{s(i)} + \alpha_i + \beta_{s(i),j}. \tag{8}$$

It is obtained by taking

$$\begin{aligned} \mathbf{X}^L &= \left(\mathbf{1}_{s(i)=1}, \mathbf{1}_{s(i)=2}, \mathbf{1}_{s(i)=3}\right) \text{ and } \mathbf{X}^B = \emptyset, \\ \mathbf{Z}^L &= \mathbf{1}_J \text{ and } \mathbf{Z}^B = \emptyset, \end{aligned}$$

where $\mathbf{1}_{s(i)=g}$ is the vector with ones for rows belonging to group $g$ and zeros everywhere else. The parametric dimension of the model is $I + 3J - 3$ for $I + 3J - 3$ parameters.

[2] from the interpretation point of view, $\mathbf{1}_I$ and $\mathbf{1}_J$, sometimes called constant covariates are not "true covariates" because they do not vary.

### 3.2.7 bilinear structuration models

To obtain more parsimonious models, qualitative covariates can be involved in the bilinear part of the model. Let us suppose that $S$ groups, defined by $s(i)$ on the levels of the row factor, and $T$ groups, defined by $t(j)$ on the levels of the column factor, are available. Then the following model can be proposed

$$\mu + \alpha_i + \beta_j + \gamma_{s(i),1}\theta_1\delta_{t(j),1} + \gamma_{s(i),2}\theta_2\delta_{t(j),2} + \ldots + \gamma_{s(i),r}\theta_r\delta_{t(j),r} \tag{9}$$

obtained by taking

$$\begin{aligned} \mathbf{X}^L &= \mathbf{1}_I \text{ and } \mathbf{X}^B = \left(\mathbf{1}_{s(i)=1}, \mathbf{1}_{s(i)=2}, \ldots, \mathbf{1}_{s(i)=S}\right), \\ \mathbf{Z}^L &= \mathbf{1}_J \text{ and } \mathbf{Z}^B = \left(\mathbf{1}_{t(j)=1}, \mathbf{1}_{t(j)=2}, \ldots, \mathbf{1}_{t(j)=T}\right). \end{aligned}$$

This model can be presented as a submodel of the biadditive model (proposed in 3) when the vectorial parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\delta}$ are restricted to be constant over the groups of levels. The parametric dimensions are $(r+1)(I+J-(r+1))$ for Model (3), and $(I+J-1)+r((S-1)+(T-1)-r)$ for Model (9) with $(I+J+1)+r(S+T+1)$ parameters.

### 3.2.8 quantitative and qualitative covariates

Of course, covariates need not be all quantitative or all qualitative but every kind of mixture can be used, within and/or between factors, for the linear and/or the bilinear part of the model. Let us give the example when qualitative covariates are used for the row factor, and quantitative covariates are available for the column factor, applying them bilinearly after the classical additive terms.

$$\mu + \alpha_i + \beta_j + \gamma_{s(i),1}\theta_1\left(\sum_{h=2}^{H^B}\delta_{h1}Z_{jh}\right) + \gamma_{s(i),2}\theta_2\left(\sum_{h=2}^{H^B}\delta_{h2}Z_{jh}\right) \tag{10}$$

is obtained by taking

$$\begin{aligned} \mathbf{X}^L &= \mathbf{1}_I \text{ and } \mathbf{X}^B = \left(\mathbf{1}_{s(i)=1}, \mathbf{1}_{s(i)=2}, \ldots, \mathbf{1}_{s(i)=S}\right), \\ \mathbf{Z}^L &= \mathbf{1}_J \text{ and } \mathbf{Z}^B = \left(Z_1, Z_2, \ldots, Z_{H^B}\right), \end{aligned}$$

and $r = 2$. The parametric dimension is $(I+J-1) + 2\left(S + H^B - 2\right)$ for $(I+J+1) + 2\left(S + H^B + 1\right)$ parameters.

## 3.3 identifiability constraints

Model parameters are defined up to some constraints; this need can be seen from the discrepancies between parametrical dimensions and the numbers of parameters. For instance, the additive model $\mu + \alpha_i + \beta_j$ need two independent constraints. Those used in BIAREG are $\sum_i \alpha_i = \sum_j \beta_j = 0$ which ensures the orthogonality of the three terms $\left(\mu, \alpha_i, \beta_j\right)$ of the model. When bilinear terms are present, some orthonormalizing constraints are also needed. The following general constraints are used in BIAREG. Finally, as already mentioned, orthogonality between linear and bilinear terms is achieved.

- for the linear parameters of the model

$$\sum_{i=1}^{I} X_{ik}^{L} \rho_{ih} = 0 \qquad \forall k = 1, ..., K^L \ ; \forall h = 1, ..., H^L$$

$$\sum_{j=1}^{J} Z_{jh}^{L} \tau_{jk} = 0 \qquad \forall h = 1, ..., H^L \ ; \forall k = 1, ..., K^L$$

- for the bilinear parameters of the model

$$\sum_{i=1}^{I} \left( \sum_{k=1}^{K^B} \gamma_{ku} X_{ik}^{B} \right)^2 = 1 \qquad \forall u = 1, ..., r$$

$$\sum_{j=1}^{J} \left( \sum_{h=1}^{H^B} \delta_{hu} Z_{jh}^{B} \right)^2 = 1 \qquad \forall u = 1, ..., r$$

$$\begin{aligned} & \sum_{i=1}^{I} \left( \sum_{k=1}^{K^B} \gamma_{ku} X_{ik}^{B} \right) \left( \sum_{k=1}^{K^B} \gamma_{kv} X_{ik}^{B} \right) \\ & = \sum_{k=1}^{K^B} \sum_{n=1}^{K^B} \gamma_{nv} \gamma_{ku} \left( \sum_{i=1}^{I} X_{ik}^{B} X_{in}^{B} \right) = 0 \end{aligned} \qquad \forall u = 1, ..., r \ ; \forall v = 1, ..., K \ ; u \neq v$$

$$\begin{aligned} & \sum_{j=1}^{J} \left( \sum_{h=1}^{H^B} \delta_{hu} Z_{jh}^{B} \right) \left( \sum_{h=1}^{H^B} \delta_{hv} Z_{jh}^{B} \right) \\ & = \sum_{h=1}^{H^B} \sum_{n=1}^{H^B} \delta_{nv} \delta_{hu} \left( \sum_{j=1}^{J} Z_{jh}^{B} Z_{jn}^{B} \right) = 0 \end{aligned} \qquad \forall u = 1, ..., r \ ; \forall v = 1, ..., K \ ; u \neq v$$

$$\sum_{i=1}^{I} X_{ik}^{L} \left( \sum_{n=1}^{K^B} \gamma_{nu} X_{in}^{B} \right) = \sum_{n=1}^{K^B} \gamma_{nu} \left( \sum_{i=1}^{I} X_{ik}^{L} X_{in}^{B} \right) = 0 \qquad \forall k = 1, ..., K^L \ ; \forall u = 1, ..., r$$

$$\sum_{j=1}^{J} Z_{jh}^{L} \left( \sum_{n=1}^{H^B} \delta_{nu} Z_{jn}^{B} \right) = \sum_{n=1}^{H^B} \delta_{nu} \left( \sum_{j=1}^{J} Z_{jh}^{L} Z_{jn}^{B} \right) = 0 \qquad \forall h = 1, ..., H^L \ ; \forall u = 1, ..., r$$

In fact these awful formulæ look much more sympathetic when written with matrices[3]! The important fact to retain is that they lead to orthogonal terms in Model (1, p5).

---

[3] $\left(\mathbf{X}^L\right)^T \boldsymbol{\rho} = \mathbf{0}; \left(\mathbf{Z}^L\right)^T \boldsymbol{\tau} = \mathbf{0};$
$\left(\mathbf{X}^B\boldsymbol{\gamma}\right)^T \mathbf{X}^B\boldsymbol{\gamma} = \boldsymbol{\gamma}^T \left(\left(\mathbf{X}^B\right)^T \mathbf{X}^B\right) \boldsymbol{\gamma} = \mathbf{I}_r;$
$\left(\mathbf{Z}^B\boldsymbol{\delta}\right)^T \mathbf{Z}^B\boldsymbol{\delta} = \boldsymbol{\delta}^T \left(\left(\mathbf{Z}^B\right)^T \mathbf{Z}^B\right) \boldsymbol{\delta} = \mathbf{I}_r;$
$\left(\mathbf{X}^L\right)^T \mathbf{X}^B\boldsymbol{\gamma} = \mathbf{0}; \left(\mathbf{Z}^L\right)^T \mathbf{Z}^B\boldsymbol{\delta} = \mathbf{0}$

# 4 SPLUS functions

Here, only the main functions useful for a standard user are described. Of course interested readers can go through the code lines which are given in the Appendix; abundant comments are included and allow, when the equations are known, every detail to find out.

## 4.1 preparing the data

As is clear from Section 2 (page 2), three basic matrices are needed for the statistical analyses: $\mathbf{Y}$, $\mathbf{X}$ and $\mathbf{Z}$. Of course rows of $\mathbf{X}$ must correspond to the rows of $\mathbf{Y}$, and the rows of $\mathbf{Z}$ must correspond to the columns of $\mathbf{Y}$. These three matrices must be provided by means of a single structure (see Appendix B) which implicitly defines the model (up to the number of multiplicative terms) for the included covariates. This structure is produced by the function DONNEES. Here is its header

```
DONNEES <- function  (titre,Y,RML=NULL, RMB=diag(nrow(Y)),
                      CML=NULL, CMB=diag(ncol(Y)),
                      ad1r=T,ad1c=T,
                      cenL=T,norL=T,
                      cenB=F,norB=F)
```

- `titre` is a character string used as a general identifier of the data set,
- `Y` is the data matrix, $\mathbf{Y}$, to analyse; if not present identifiers for rows and columns will be added,
- `RML` is the $\mathbf{X}^L$ matrix (Row Matrix Linear); its row identifiers will be copied from those of $\mathbf{Y}$. If not present, identifiers for its columns will be added. The default is `NULL` which means that there are no linearly used covariates for the row factor,
- `RMB` is the $\mathbf{X}^B$ matrix (Row Matrix Bilinear); its row identifiers will be copied from those of $\mathbf{Y}$. If not present, identifiers for its columns will be added. The default is `diag(nrow(Y))`, the identity matrix, which means that standard multiplicative terms will be considered,
- `CML` is the $\mathbf{Z}^L$ matrix (Column Matrix Linear); its row identifiers will be copied from the identifiers of $\mathbf{Y}$ columns. If not present, identifiers for its columns will be added. The default is `NULL` which means that there are no linearly used covariates for the column factor,
- `RMB` is the $\mathbf{Z}^B$ matrix (Column Matrix Bilinear); its row identifiers will be copied from the identifiers of $\mathbf{Y}$ columns. If not present, identifiers for its columns will be added. The default is `diag(ncol(Y))`, the identity matrix, which means that standard multiplicative terms will be considered,
- `ad1r` indicates when True that a constant covariate must be added, in the first position, to the linear row covariates,

- `adlc` indicates when True that a constant covariate must be added, in the first position, to the linear column covariates,
- `cenL` indicates when True that the centring of linear covariates (both row and column ones) must be performed (of course not for any added constant covariate),
- `norL` indicates when True that the normalization of linear covariates (both row and column ones) must be performed (not for any added constant covariate),
- `cenB` indicates when True that the centring of bilinear covariates (both row and column ones) must be performed,
- `norB` indicates when True that the normalization of bilinear covariates (both row and column ones) must be performed.

Due to the choice of defaults, models presented in Section (3.2) are easily prepared. For example

```
data3 <- DONNEES(''Model 3'',Y)
data5 <- DONNEES(''Model 5'',Y,RML=X,CML=Z,RMB=NULL,CMB=NULL)
data6 <- DONNEES(''Model 6'',Y,RML=X,CML=Z)                      (11)
data8 <- DONNEES(''Model 8'',Y,RML=S,RMB=NULL,CMB=NULL)
data9 <- DONNEES(''Model 9'',Y,RMB=S,CMB=T)
```

The result of the function DONNEES is the major argument for processing the statistical calculations.

## 4.2 performing the statistical calculations

BR performs all statistical calculations according to the model implicitly defined by the covariates and `nbmul`. Results are stored in a complex structure (described in Appendix B page 14) which can be ignored when one wants only to print them. For the sake of safety, BR performs a lot of checking for consistency between arguments; if something appears wrong, an explicit error message is displayed beginning with the French word ERREUR. Here is its header

```
BR <- function  (d,nbmul=1,epsilon=1e-5)
```

- `d` is the resultant structure provided by DONNEES,
- `nbmul` is the number of multiplicative terms, default is one,
- `epsilon` is the threshold value used for detecting collinearities within covariate sets. From a statistical viewpoint too small values should be avoided; the default ($10^{-5}$) seems a good choice.

Examples corresponding to preparations (11) are

```
calcul3 <- BR(data3,nbmul=r)    where r is a given number
calcul5 <- BR(data5,nbmul=0)
calcul6 <- BR(data6)                                        (12)
calcul8 <- BR(data8,nbmul=0)
calcul9 <- BR(data9,nbmul=r)    where r is a given number
```

## 4.3 displaying data and results

For the moment, three specific functions have been written for displaying the results.

### 4.3.1 displaying the data sets

To check precisely what was stored in the data structure, one can use `printBR0`.

```
printBR0 <- function  (data,valeurs=F)
```

is intended to display the structure produced by `DONNEES`. When `valeurs=T`, the numerical values of the data set are printed; if not, only the numbers and names of the levels and covariates involved are. For instance

```
printBR0(data5,valeurs=T),
printBR0(data9).
```
(13)

### 4.3.2 displaying the anova tables

`printBR1` can be used for obtaining the anova table of an analysis performed by function `BR`. The lines of the table are identified by the numbers used when describing the model in Section 3.1 (page 4) and generic denominations. Examples are given in Appendix C.1 (page 15). The header is

```
printBR1 <- function  (calcul,lar=c(5,3,19,6,15,10),fmt=c(4,2),sep=''|'')
```

where `calcul` must be a structure returned by `BR`, and other arguments are facilities to adjust the output :

- `lar[1]` is the number of characters to display in the term numbers,
- `lar[2]` is the number of space characters to introduce after the term numbers,
- `lar[3]` is the number of characters to display in the identification of the line,
- `lar[4]` is the number of characters to display in parametrical dimensions (degrees of freedom),
- `lar[5]` is the number of characters to display in the Sum of Squares,
- `lar[6]` is the number of characters to display in the Mean Squares,
- `fmt[1]` is the number of decimals for the Sum of Squares,
- `fmt[2]` is the number of decimals for the Mean Squares,
- `sep` is the string of characters for separating the columns of the anova table.

For instance

```
printBR1(calcul5),
printBR1(calcul9).
```
(14)

### 4.3.3 displaying estimations, ...

`printBR2` can be used for printing the detailed information concerning **one** of the terms of the model adjusted by `BR`. Examples are given in Appendix C.1 (page 15). The header is

```
printBR2 <- function  (Bu,quoi=1:6)
```

where `Bu` must be one of the eight components of the list produced by function `BR`. The numbering of these components is identical to the numbers used when describing the general Model (3) in Section 3.1 (page 6). They must be indicated in double brackets. `quoi` allows one to produce only a part of the information

- `quoi[1]` for obtaining the identification of the term,
- `quoi[2]` for obtaining the pseudo degrees of freedom of the term,
- `quoi[3]` for obtaining the Sum of Squares [& Mean Squares of the term],
- `quoi[4]` for obtaining the value fitted by the term,
- `quoi[5]` for obtaining the estimated parameters of the term.

For instance

```
printBR2(calcul5[[1]]),
printBR2(calcul10[[4]],c(1,2,4)).
```
(15)

# Acknowledgment

This manual benefited from the careful reading and suggestions of Maryse BRANCOURT-HULMEL and John C. GOWER: thanks to them.

# A References

1. (1990/1991/1992) G. Decoux et J.-B. Denis, **INTERA, version 3.1**, Notice d'Utilisation de Logiciels pour l'Interprétation Statistique de l'Interaction entre Deux Facteurs, *Laboratoire de Biométrie, INRA*, route de St-Cyr, F-78026 Versailles Cédex, 120pp. Révisée pour la version 3.2 en mars 1991, 175pp. Remise à jour et rééditée pour la version 3.3 en mars 1992.
2. (1988) J.-B. Denis, **Two Way Analysis Using Covariates**, *Statistics*, 19(1), 123-132.
3. (1991) J.-B. Denis, **Ajustement de Modèles Linéaires et Bilinéaires sous Contraintes Linéaires avec Données Manquantes**, *Revue de Statistique Appliquée*, 39(2), 5-24.

4. (1996) Denis, J.-B. et J.C. Gower, **Asymptotic confidence regions for biadditive models: interpreting genotype-environment interactions,** *Appl. Statist.* 4, 479-493.

5. (1996) F.A. van Eeuwijk, J.-B. Denis and M.S. Kang, **Incorporating additional information on genotypes and environments in models for two-way genotype by environment tables,** *In* Kang, M.S. et Gauch, H.G., eds. Genotype-by-Environment Interaction, CRC-Press, Boca Raton, Fl, U.S.A. pp15-49.

# B SPLUS structures

There are two basic structures in BIAREG: the one produced by `DONNEES` and the one produced by `BR`. Probably the best way to know their details is to examine the SPLUS code given in Appendix D.1 (page 23). Nevertheless some prior rough description can be of use.

- Let `don` be a structure produced by the function `DONNEES`:
  - `don$titre` is a character string intended as a title,
  - `don$Y` is the data matrix to be analysed,
  - `don$RML` is the data of row covariates to be used linearly,
  - `don$RMB` is the data of row covariates to be used bilinearly,
  - `don$CML` is the data of column covariates to be used linearly,
  - `don$CMB` is the data of column covariates to be used bilinearly.

- Let `cal` be a structure produced by the function `BR`:
  - `cal[[u]]` for `u=1,...,8` are lists associated to each of the eight terms of the model as numbered in the general Formulation (1),
    * `cal[[u]][[1]]` is a character string identifying the term,
    * `cal[[u]][[2]]` is the number of pseudo-degrees of freedom of the term,
    * `cal[[u]][[3]]` is the Sum of Squares associated with the term,
    * `cal[[u]][[4]]` is the fitted matrix of size $\mathbf{Y}$ associated with the term,
    * `cal[[u]][[5]]` is the number of estimated parameter matrices, 0 if none,
    * `cal[[u]][[5+k]]` for `k=1,...,cal[[u]][[5]]` is a two components list,
      · `cal[[u]][[5+k]][[1]]` is a character string identifying the matrix,
      · `cal[[u]][[5+k]][[2]]` is the matrix itself,
  - `cal[[9]]` is a list giving main dimensions of the model, namely $(I, J, K^L, K^B, H^L, H^B, r)$,
  - `cal[[10]]` is a character string giving identification inherited from the data (`$titre`).

# C Examples

## C.1 SPLUS program

```
# more or less printing
doni   <-  T;  # initial data set
first <-  T;  # first analysis
secon <-  T;  # second analysis
third <-  T;  # third analysis
fourth <- T;  # fourth analysis

# preparing the three needed matrices
source("biareg01.don");
Z <- Z[,c(1,3,6,8)];
A91 <- c(rep(1,8),rep(0,8));
A92 <- c(rep(0,8),rep(1,8));
Z2 <- cbind(A91,A92);

options(width=80);
sink("biareg01.res");

# printing them
if (doni) {
  cat("\n The variate to analyse \n\n");
  print(Y);
  cat("\n Genotypic covariates \n\n");
  print(X);
  cat("\n Environmental covariates \n\n");
  print(cbind(Z,Z2));
  }

if (first) {
   titre("the basic model without covariates",10);
   data <- DONNEES("Classical Biadditive Analysis",Y);
  # printBR0(data);
   anal <- BR(data,2);
   printBR1(anal,fmt=c(6,0));
  }

if (secon) {
  # only linear approach
  # looking for the "best" genotypic covariate
  titre("trying separately each genotypic covariate",10);
  for (chen in 1:3) {
   XX <- X[,c(chen),drop=F];
   titre(paste("Covariate number",chen),2);
   data <- DONNEES(paste("second trial [",chen,"]"),Y,
                   RML=XX,CML=Z,
                   RMB=NULL,CMB=NULL);
   printBR0(data);
   anal <- BR(data,0);
   printBR1(anal,lar=c(7,0,20,8,12,12));
  }}

if (third) {
 titre("Additive plus Bilinear approach by Covariates",10)
 data <- DONNEES("biareg01.don",Y,
                 RMB=X,CMB=Z);
 printBR0(data);
 anal <- BR(data,1);
 printBR1(anal);
 for (jbd in 1:8) {
   printBR2(anal[[jbd]]);
   }
 }
```

```
print(fourth)
if (fourth) {
 titre("\n Bilinear approach with \n Quantitative and Qualitative
Covariates",10)
 data <- DONNEES("biareg01.don",Y,
                   ad1r=F,ad1c=F,
                   RMB=X,CMB=Z2);
 printBR0(data);
 anal <- BR(data,1);
 printBR1(anal);
 printBR2(anal[[4]],4:5);
 }

sink();
```

## C.2 results

```
The variate to analyse

    DIJI91 DIJ291 MINF91 MINI91 MONF91 MONI91 RENF91 RENI91 DIJI92 DIJ292
ARM  83.15  71.80  63.30  85.95  57.45  73.00  52.85  74.25  73.75  73.10
CAR  79.55  74.35  73.60  87.45  56.30  71.85  64.40  88.45  69.25  76.75
SOI  89.25  80.55  89.85 108.20  63.35  80.15  77.85  97.40  82.50  84.55
TAL  81.00  75.95  67.70  90.50  51.95  73.60  74.80  87.20  76.40  76.05
    MINF92 MINI92 MONF92 MONI92 RENF92 RENI92
ARM   50.5  58.45  63.05  66.80  49.20  59.30
CAR   58.2  58.70  60.25  66.70  45.85  61.65
SOI   72.3  84.65  64.85  71.05  65.85  85.60
TAL   62.3  70.70  68.05  62.85  49.75  53.15

 Genotypic covariates

     ht f1rp v1ma
ARM 105    2    5
CAR 103    3    4
SOI  91    2    1
TAL  90    1    2

 Environmental covariates

          stmpg spetpem   stmpr  spetpg A91 A92
DIJI91 1551.375 -60.975 635.450 198.425   1   0
DIJ291 1377.225 -90.725 628.375 171.150   1   0
MINF91 1605.850 -40.275 621.725 155.625   1   0
MINI91 1605.850 -40.275 621.725 155.625   1   0
MONF91 1714.825 -46.600 636.425 163.250   1   0
MONI91 1714.825 -46.600 636.425 163.250   1   0
RENF91 1547.000   2.825 627.500 130.200   1   0
RENI91 1547.000   2.825 627.500 130.200   1   0
DIJI92 1451.750  -9.175 627.700 155.350   0   1
DIJ292 1307.250 -47.150 636.650 120.975   0   1
MINF92 1512.275 -55.125 626.225  48.925   0   1
MINI92 1512.275 -55.125 626.225  48.925   0   1
MONF92 1626.575 -54.875 637.925  84.025   0   1
MONI92 1626.575 -54.875 637.925  84.025   0   1
RENF92 1539.075 -38.550 630.375  29.850   0   1
RENI92 1539.075 -38.550 630.375  29.850   0   1

   ==================================
   the basic model without covariates
   ==================================


   ANOVA TABLE  ( Classical Biadditive Analysis )
   ==============================================

           Size of Data set = (4,16)
       Decomposition of Rows:     4 =     1 [lin.] +     3 [bil.] +     0
    Decomposition of Columns:    16 =     1 [lin.] +    15 [bil.] +     0
Number of Multiplicative Terms = 2
```

| Term(s) | Source | P.Dim. | S.S. | M.S. |
|---|---|---|---|---|
| 1 | lin. common term | 1 | 326198.043906 | 326198 |
| 2 | lin. col cov. | 3 | 2156.687656 | 719 |
| 3 | lin. row cov. | 15 | 6960.397344 | 464 |
| 4 | bil. cov. [ 1 ] | 17 | 646.247148 | 38 |
| 4 | bil. cov. [ 2 ] | 15 | 408.276831 | 27 |
| 5 | bil. cov. rem. | 13 | 160.567114 | 12 |
| 9 | 1 + 2 + 3 | 19 | 335315.128906 | 17648 |
| 10 | 4 | 32 | 1054.523979 | 33 |
| 11 | 4 + 5 | 45 | 1215.091094 | 27 |
| 12 | 4 + 5 + 6 + 7 | 45 | 1215.091094 | 27 |
| 13 | 4 + 5 + 6 + 7 + 8 | 45 | 1215.091094 | 27 |

(Term(s) with null degrees of freedom are eliminated,
for more details see the manual)

===========================================
trying separately each genotypic covariate
===========================================

Covariate number 1

Row Decomposition:
2 covariates for the linear part
1 -1-
2 ht
0 covariates for the bilinear part

Column Decomposition:
5 covariates for the linear part
1 -1-
2 stmpg
3 spetpem
4 stmpr
5 spetpg
0 covariates for the bilinear part

ANOVA TABLE ( second trial [ 1 ] )
==================================

| | | | | | |
|---|---|---|---|---|---|
| Size of Data set = | (4,16) | | | | |
| Decomposition of Rows: | 4 = | 2 [lin.] + | 0 [bil.] + | 2 |
| Decomposition of Columns: | 16 = | 5 [lin.] + | 0 [bil.] + | 11 |

No Multiplicative Term

| Term(s) | Source | P.Dim. | S.S. | M.S. |
|---|---|---|---|---|
| 1 | lin. common term | 10 | 331183.1256 | 33118.31 |
| 2 | lin. col cov. | 10 | 1335.1233 | 133.51 |
| 3 | lin. row cov. | 22 | 3510.8662 | 159.58 |
| 8 | all cov. reminder | 22 | 501.1048 | 22.78 |
| 9 | 1 + 2 + 3 | 42 | 336029.1152 | 8000.69 |
| 13 | 4 + 5 + 6 + 7 + 8 | 22 | 501.1048 | 22.78 |

(Term(s) with null degrees of freedom are eliminated,
for more details see the manual)

Covariate number 2

Row Decomposition:
2 covariates for the linear part
1 -1-
2 f1rp
0 covariates for the bilinear part

Column Decomposition:
5 covariates for the linear part
1 -1-
2 stmpg
3 spetpem
4 stmpr
5 spetpg
0 covariates for the bilinear part

ANOVA TABLE ( second trial [ 2 ] )

Size of Data set = (4,16)
Decomposition of Rows: 4 = 2 [lin.] + 0 [bil.] + 2
Decomposition of Columns: 16 = 5 [lin.] + 0 [bil.] + 11
No Multiplicative Term

| Term(s) | Source | P.Dim. | S.S. | M.S. |
|---|---|---|---|---|
| 1 | lin. common term | 10 | 329823.1741 | 32982.32 |
| 2 | lin. col cov. | 10 | 2695.0748 | 269.51 |
| 3 | lin. row cov. | 22 | 3613.2809 | 164.24 |
| 8 | all cov. reminder | 22 | 398.6902 | 18.12 |
| 9 | 1 + 2 + 3 | 42 | 336131.5298 | 8003.13 |
| 13 | 4 + 5 + 6 + 7 + 8 | 22 | 398.6902 | 18.12 |

(Term(s) with null degrees of freedom are eliminated,
for more details see the manual)

Covariate number 3

Row Decomposition:
2 covariates for the linear part
1 -1-
2 v1ma
0 covariates for the bilinear part

Column Decomposition:
5 covariates for the linear part
1 -1-
2 stmpg
3 spetpem
4 stmpr
5 spetpg
0 covariates for the bilinear part

ANOVA TABLE ( second trial [ 3 ] )

Size of Data set = (4,16)
Decomposition of Rows: 4 = 2 [lin.] + 0 [bil.] + 2
Decomposition of Columns: 16 = 5 [lin.] + 0 [bil.] + 11
No Multiplicative Term

| Term(s) | Source | P.Dim. | S.S. | M.S. |
|---|---|---|---|---|
| 1 | lin. common term | 10 | 331882.158 | 33188.22 |
| 2 | lin. col cov. | 10 | 636.091 | 63.61 |
| 3 | lin. row cov. | 22 | 3479.692 | 158.17 |
| 8 | all cov. reminder | 22 | 532.279 | 24.19 |
| 9 | 1 + 2 + 3 | 42 | 335997.941 | 7999.95 |
| 13 | 4 + 5 + 6 + 7 + 8 | 22 | 532.279 | 24.19 |

(Term(s) with null degrees of freedom are eliminated,
for more details see the manual)

Additive plus Bilinear approach by Covariates

Row Decomposition:
1 covariates for the linear part
1 -1-
3 covariates for the bilinear part
1 ht
2 f1rp
3 v1ma

Column Decomposition:
1 covariates for the linear part
1 -1-
4 covariates for the bilinear part
1 stmpg
2 spetpem
3 stmpr
4 spetpg

ANOVA TABLE ( biareg01.don )
============================

Size of Data set = (4,16)
Decomposition of Rows: 4 = 1 [lin.] + 3 [bil.] + 0
Decomposition of Columns: 16 = 1 [lin.] + 3 [bil.] + 12
Number of Multiplicative Terms = 1

| Term(s) | Source | P.Dim. | S.S. | M.S. |
|---|---|---|---|---|
| 1 | lin. common term | 1 | 326198.0439 | 326198.04 |
| 2 | lin. col cov. | 3 | 2156.6877 | 718.9 |
| 3 | lin. row cov. | 15 | 6960.3973 | 464.03 |
| 4 | bil. cov. [ 1 ] | 5 | 514.1511 | 102.83 |
| 5 | bil. cov. rem. | 4 | 66.946 | 16.74 |
| 7 | bil. row cov. rem. | 36 | 633.9939 | 17.61 |
| 9 | 1 + 2 + 3 | 19 | 335315.1289 | 17648.16 |
| 10 | 4 | 5 | 514.1511 | 102.83 |
| 11 | 4 + 5 | 9 | 581.0971 | 64.57 |
| 12 | 4 + 5 + 6 + 7 | 45 | 1215.0911 | 27 |
| 13 | 4 + 5 + 6 + 7 + 8 | 45 | 1215.0911 | 27 |

(Term(s) with null degrees of freedom are eliminated,
for more details see the manual)

lin. common term
================

"DoF" = 1 S.S. = 326198.043906249 M.S. = 326198.043906249

Fitted Term

| | DIJI91 | DIJ291 | MINF91 | MINI91 | MONF91 | MONI91 | RENF91 | RENI91 |
|---|---|---|---|---|---|---|---|---|
| ARM | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 |
| CAR | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 |
| SOI | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 |
| TAL | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 |

| | DIJI92 | DIJ292 | MINF92 | MINI92 | MONF92 | MONI92 | RENF92 | RENI92 |
|---|---|---|---|---|---|---|---|---|
| ARM | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 |
| CAR | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 |
| SOI | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 |
| TAL | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 | 71.39219 |

Parameters

| | -1- |
|---|---|
| -1- | 4569.1 |

lin. col cov.
=============

"DoF" = 3 S.S. = 2156.68765624999 M.S. = 718.895885416663

Fitted Term

| | DIJI91 | DIJ291 | MINF91 | MINI91 | MONF91 | MONI91 | RENF91 |
|---|---|---|---|---|---|---|---|
| ARM | -5.398437 | -5.398437 | -5.398437 | -5.398437 | -5.398437 | -5.398437 | -5.398437 |
| CAR | -3.060937 | -3.060937 | -3.060937 | -3.060937 | -3.060937 | -3.060937 | -3.060937 |
| SOI | 9.729687 | 9.729687 | 9.729687 | 9.729687 | 9.729687 | 9.729687 | 9.729687 |
| TAL | -1.270312 | -1.270312 | -1.270312 | -1.270312 | -1.270312 | -1.270312 | -1.270312 |

| | RENI91 | DIJI92 | DIJ292 | MINF92 | MINI92 | MONF92 | MONI92 |
|---|---|---|---|---|---|---|---|
| ARM | -5.398437 | -5.398437 | -5.398437 | -5.398437 | -5.398437 | -5.398437 | -5.398437 |
| CAR | -3.060937 | -3.060937 | -3.060937 | -3.060937 | -3.060937 | -3.060937 | -3.060937 |
| SOI | 9.729687 | 9.729687 | 9.729687 | 9.729687 | 9.729687 | 9.729687 | 9.729687 |
| TAL | -1.270312 | -1.270312 | -1.270312 | -1.270312 | -1.270312 | -1.270312 | -1.270312 |

| | RENF92 | RENI92 |
|---|---|---|
| ARM | -5.398437 | -5.398437 |
| CAR | -3.060937 | -3.060937 |
| SOI | 9.729687 | 9.729687 |
| TAL | -1.270312 | -1.270312 |

Parameters

| | -1- |
|---|---|
| ARM | -86.375 |
| CAR | -48.975 |
| SOI | 155.675 |
| TAL | -20.325 |

lin. row cov.
=============

"DoF" = 15 S.S. = 6960.39734375 M.S. = 464.026489583333

Fitted Term

| | DIJI91 | DIJ291 | MINF91 | MINI91 | MONF91 | MONI91 | RENF91 | RENI91 |
|---|---|---|---|---|---|---|---|---|
| ARM | 11.84531 | 4.270313 | 2.220313 | 21.63281 | -14.12969 | 3.257813 | -3.917187 | 15.43281 |
| CAR | 11.84531 | 4.270313 | 2.220313 | 21.63281 | -14.12969 | 3.257813 | -3.917187 | 15.43281 |
| SOI | 11.84531 | 4.270313 | 2.220313 | 21.63281 | -14.12969 | 3.257813 | -3.917187 | 15.43281 |
| TAL | 11.84531 | 4.270313 | 2.220313 | 21.63281 | -14.12969 | 3.257813 | -3.917187 | 15.43281 |

| | DIJI92 | DIJ292 | MINF92 | MINI92 | MONF92 | MONI92 | RENF92 |
|---|---|---|---|---|---|---|---|
| ARM | 4.082813 | 6.220313 | -10.56719 | -3.267187 | -7.342187 | -4.542187 | -18.72969 |
| CAR | 4.082813 | 6.220313 | -10.56719 | -3.267187 | -7.342187 | -4.542187 | -18.72969 |
| SOI | 4.082813 | 6.220313 | -10.56719 | -3.267187 | -7.342187 | -4.542187 | -18.72969 |
| TAL | 4.082813 | 6.220313 | -10.56719 | -3.267187 | -7.342187 | -4.542187 | -18.72969 |

| | RENI92 |
|---|---|
| ARM | -6.467187 |
| CAR | -6.467187 |
| SOI | -6.467187 |
| TAL | -6.467187 |

Parameters

| | -1- |
|---|---|
| DIJI91 | 47.38125 |
| DIJ291 | 17.08125 |
| MINF91 | 8.88125 |
| MINI91 | 86.53125 |
| MONF91 | -56.51875 |
| MONI91 | 13.03125 |
| RENF91 | -15.66875 |
| RENI91 | 61.73125 |
| DIJI92 | 16.33125 |
| DIJ292 | 24.88125 |
| MINF92 | -42.26875 |
| MINI92 | -13.06875 |
| MONF92 | -29.36875 |
| MONI92 | -18.16875 |
| RENF92 | -74.91875 |
| RENI92 | -25.86875 |

bil. cov.
=========

"DoF" = 5 S.S. = 514.151138224161 M.S. = 102.830227644832

Fitted Term

| | DIJI91 | DIJ291 | MINF91 | MINI91 | MONF91 | MONI91 |
|---|---|---|---|---|---|---|
| ARM | 5.6223591 | 1.9497476 | -3.9632342 | -3.9632342 | 4.6341182 | 4.6341182 |
| CAR | 1.3323710 | 0.4620457 | -0.9391962 | -0.9391962 | 1.0981804 | 1.0981804 |
| SOI | -6.5800353 | -2.2818550 | 4.6383059 | 4.6383059 | -5.4234638 | -5.4234638 |
| TAL | -0.3746948 | -0.1299384 | 0.2641246 | 0.2641246 | -0.3088348 | -0.3088348 |

| | RENF91 | RENI91 | DIJI92 | DIJ292 | MINF92 | MINI92 |
|---|---|---|---|---|---|---|
| ARM | -2.7599528 | -2.7599528 | -1.5085088 | 3.8621857 | -3.8643784 | -3.8643784 |
| CAR | -0.6540459 | -0.6540459 | -0.3574822 | 0.9152500 | -0.9157696 | -0.9157696 |
| SOI | 3.2300653 | 3.2300653 | 1.7654584 | -4.5200454 | 4.5226115 | 4.5226115 |
| TAL | 0.1839335 | 0.1839335 | 0.1005326 | -0.2573903 | 0.2575365 | 0.2575365 |

| | MONF92 | MONI92 | RENF92 | RENI92 |
|---|---|---|---|---|
| ARM | 3.5980448 | 3.5980448 | -2.6074893 | -2.6074893 |
| CAR | 0.8526546 | 0.8526546 | -0.6179155 | -0.6179155 |
| SOI | -4.2109125 | -4.2109125 | 3.0516322 | 3.0516322 |
| TAL | -0.2397870 | -0.2397870 | 0.1737727 | 0.1737727 |

Row Scores

| | M_1 |
|---|---|
| ht | -10.8473824 |
| f1rp | -0.1947617 |
| v1ma | -2.9791404 |

```
SVD value(s)

        M_1
M_1 22.6749

 Column Scores

              M_1
  stmpg -60.68109
spetpem  40.15306
  stmpr -19.23778
 spetpg -90.32880


   bil. cov. rem.
   ==============

    "DoF" =  4     S.S. =  66.9460087851437     M.S. =  16.7365021962859


 Fitted Term

        DIJI91      DIJ291      MINF91       MINI91      MONF91     MONI91
ARM -0.5008568  0.22540552 -0.02096487 -0.02096487  0.2149954  0.2149954
CAR  0.4542822 -0.38871309  0.49145708  0.49145708  0.3271451  0.3271451
SOI -0.3590738  0.11090612  0.11493681  0.11493681  0.2977753  0.2977753
TAL  0.4056484  0.05240145 -0.58542902 -0.58542902 -0.8399158 -0.8399158
       RENF91    RENI91    DIJI92      DIJ292      MINF92     MINI92      MONF92
ARM -1.646059 -1.646059 -2.018240 -1.22965478  1.2539498  1.2539498  1.0271296
CAR  1.121444  1.121444  1.164424 -0.02787666 -0.9823677 -0.9823677 -0.6928952
SOI -1.282303 -1.282303 -1.630168 -1.19604932  0.9416145  0.9416145  0.8020404
TAL  1.806918  1.806918  2.483984  2.45358076 -1.2131967 -1.2131967 -1.1362747
        MONI92     RENF92     RENI92
ARM  1.0271296  0.9326220  0.9326220
CAR -0.6928952 -0.8658413 -0.8658413
SOI  0.8020404  0.6631283  0.6631283
TAL -1.1362747 -0.7299090 -0.7299090


   bil. col cov. rem.
   ==================

    "DoF" =  0     S.S. =  0


 Fitted Term

NULL


   bil. row cov. rem.
   ==================

    "DoF" =  36     S.S. =  633.993946740697     M.S. =  17.6109429650194


 Fitted Term

        DIJI91     DIJ291     MINF91     MINI91     MONF91     MONI91
ARM  0.1894352 -0.6392156 -0.9298634  2.3076366  0.7368239 -1.100676
CAR -2.4132157  1.6751048  3.4961766 -2.0663234  0.6731120 -1.164388
SOI  3.2219216 -2.6712387  1.7545698  0.6920698  1.4835010  0.896001
TAL -0.9981412  1.6353494 -4.3208831 -0.9333831 -2.8934369  1.369063
        RENF91    RENI91     DIJI92     DIJ292    MINF92    MINI92     MONF92
ARM -4.8205507 -2.770551  7.2001862 -1.746593 -2.316134 -1.666134 -0.2267369
CAR -0.4814605  4.218540 -3.9710040  1.311064  2.334075 -4.465925 -0.8988219
SOI -1.3024499 -1.102450 -2.8399783  2.923907 -3.718914  1.331086 -5.5208154
TAL  6.6044610 -0.345539 -0.3892039 -2.488378  3.700973  4.800973  6.6463742
        MONI92     RENF92     RENI92
ARM  0.7232631  3.6108048  1.448305
CAR  2.7511781 -2.2678057  1.269694
SOI -2.1208154 -0.2569479  7.230552
TAL -1.3536258 -1.0860512 -9.948551
```

```
   all cov. reminder
   =================

    "DoF" =  0      S.S. =  0

 Fitted Term

NULL
[1] T

   ===================================================================

 Bilinear approach with
 Quantitative and Qualitative Covariates
   ===================================================================

 Row Decomposition:
    0 covariates for the linear part
    3 covariates for the bilinear part
       1    ht
       2    f1rp
       3    v1ma

 Column Decomposition:
     0 covariates for the  linear part
    2 covariates for the bilinear part
       1    A91
       2    A92

   ANOVA TABLE  ( biareg01.don )
   =============================

          Size of Data set = (4,16)
      Decomposition of Rows:    4 =     0 [lin.] +     3 [bil.] +     1
   Decomposition of Columns:   16 =     0 [lin.] +     2 [bil.] +    14
Number of Multiplicative Terms = 1

|--------|-------------------|------|---------------|----------|
| Term(s)|             Source|P.Dim.|          S.S. |   M.S.   |
|--------|-------------------|------|---------------|----------|
|    4   |   bil. cov.  [ 1 ]|     4|    329862.2292|  82465.56|
|    5   |     bil. cov. rem.|     2|        13.1126|      6.56|
|    6   | bil. col cov. rem.|     2|       161.7763|     80.89|
|    7   | bil. row cov. rem.|    42|      6176.2761|    147.05|
|    8   |  all cov. reminder|    14|       316.8258|     22.63|
|--------|-------------------|------|---------------|----------|
|   10   |                  4|     4|    329862.2292|  82465.56|
|   11   |              4 + 5|     6|    329875.3418|  54979.22|
|   12   |      4 + 5 + 6 + 7|    50|    336213.3942|   6724.27|
|   13   |  4 + 5 + 6 + 7 + 8|    64|      336530.22|   5258.28|
|--------|-------------------|------|---------------|----------|
 (Term(s) with null degrees of freedom are eliminated,
     for more details see the manual)

 Fitted Term

       DIJI91   DIJ291   MINF91   MINI91   MONF91   MONI91   RENF91   RENI91
ARM 69.13302 69.13302 69.13302 69.13302 69.13302 69.13302 69.13302 69.13302
CAR 74.71837 74.71837 74.71837 74.71837 74.71837 74.71837 74.71837 74.71837
SOI 85.31438 85.31438 85.31438 85.31438 85.31438 85.31438 85.31438 85.31438
TAL 76.66585 76.66585 76.66585 76.66585 76.66585 76.66585 76.66585 76.66585
       DIJI92   DIJ292   MINF92   MINI92   MONF92   MONI92   RENF92   RENI92
ARM 59.99907 59.99907 59.99907 59.99907 59.99907 59.99907 59.99907 59.99907
CAR 64.84646 64.84646 64.84646 64.84646 64.84646 64.84646 64.84646 64.84646
SOI 74.04252 74.04252 74.04252 74.04252 74.04252 74.04252 74.04252 74.04252
TAL 66.53665 66.53665 66.53665 66.53665 66.53665 66.53665 66.53665 66.53665

 Row Scores

             M_1
  ht -193.134136
f1rp   -3.975790
v1ma   -5.759007

 SVD value(s)

         M_1
M_1 574.3363

 Column Scores

         M_1
A91 -2.136130
A92 -1.853901
```

# D Splus code

## D.1 BiaReg functions

```
####################
printBR0 <- function(data,valeurs=F)
{
# prints the data sets "dat"
# main argument for "BR"
I <- nrow(data$Y); J <- ncol(data$Y);
if (is.null(data$RML)) {KL <- 0}
else {KL <- ncol(data$RML)}
if (is.null(data$RMB)) {KB <- 0}
else {KB <- ncol(data$RMB)}
if (is.null(data$CML)) {HL <- 0}
else {HL <- ncol(data$CML)}
if (is.null(data$CMB)) {HB <- 0}
else {HB <- ncol(data$CMB)}

if (valeurs) {
  cat("\n\n Two-Way Table to
interpret\n\n")
  print(data$Y);
  }

titre("\n Row Decomposition: \n",0,0);
titre(V(KL,0),0,5);
titre(paste(" covariates for the",
          "linear part"),0);
if (KL > 0) {
  if (valeurs) {cat("\n");
          print(data$RML)}
  else {
if (is.null(dimnames(data$RML)[[2]]))
    {titre(" without names\n",0,0);}
    else { cat("\n");for(jbd in 1:KL){
    titre(V(jbd,0),0,8);repete(" ",3);
    titre(dimnames(data$RML)[[2]][jbd],
                                0,0);

    cat("\n");
    }}}}else{cat("\n")}
titre(V(KB,0),0,5);
titre(paste(" covariates for",
          "the bilinear part"),0);
if (KB > 0) {
  if (valeurs){cat("\n");
                  print(data$RMB)}
  else {
    if (is.null(dimnames(
                  data$RMB)[[2]])) {
      titre(" without names\n",0,0);}
   else { cat("\n");for(jbd in 1:KB) {
    titre(V(jbd,0),0,8);repete(" ",3);
    titre(dimnames(data$RMB)[[2]]
                        [jbd],0,0);
    cat("\n");
    }}}}else{cat("\n")}
titre("\n Column Decomposition:\n ",0,0);
titre(V(HL,0),0,5);
titre(paste(" covariates for the",
                " linear part"),0);
if (HL > 0) {
  if (valeurs) {cat("\n");
                  print(data$CML)}
  else {
    if (is.null(dimnames(data$CML)
                          [[2]])) {
      titre(" without names\n",0,0);}
    else { cat("\n");
                 for (jbd in 1:HL) {
     titre(V(jbd,0),0,8);repete(" ",3);
     titre(dimnames(data$CML)[[2]]
                         [jbd],0,0);
     cat("\n");
     }}}}else{cat("\n")}
 titre(V(HB,0),0,5);
 titre(paste(" covariates for the",
                 "bilinear part"),0);
 if (HB > 0) {
   if (valeurs) {cat("\n");
                   print(data$CMB)}
   else {
     if (is.null(dimnames(data$CMB)
                          [[2]])) {
       titre(" without names\n",0,0);}
     else { cat("\n");
                  for (jbd in 1:HB) {
     titre(V(jbd,0),0,8);repete(" ",3);
     titre(dimnames(data$CMB)
                 [[2]][jbd],0,0);
     cat("\n");
     }}}}else{cat("\n")}
NULL;
}

####################
printBR1 <- function(BB,
              lar=c(5,3,19,6,15,10),
              fmt=c(4,2),sep="|")
{
# prints the anova table
# "BB" being an output of "BR"
#
carac <- BB[[9]];
titre(paste("ANOVA TABLE  (",
                  BB[[10]],")"),5);
titre(paste("            Size of ",
                 "Data set = (",
        carac[1],",",carac[2],")\n",
                         sep=""),0);
titre("Decomposition of Rows: ",0,30);
titre(V(carac[1],0),0,4);
titre(" = ",0);
titre(V(carac[3],0),0,4);
titre(" [lin.] + ",0);
titre(V(carac[5],0),0,4);
titre(" [bil.] + ",0);
titre(V(carac[1]-(carac[3]+carac[5]),
                         0),0,4);
cat("\n");
titre("Decomposition of Columns: ",
                           0,30);
titre(V(carac[2],0),0,4);
titre(" = ",0);
titre(V(carac[4],0),0,4);
titre(" [lin.] + ",0);
titre(V(carac[6],0),0,4);
titre(" [bil.] + ",0);
titre(V(carac[2]-(carac[4]+carac[6]),
                         0),0,4);
cat("\n");

if (carac[7] >0) {
titre(paste("Number of ",
   "Multiplicative Terms = ",carac[7],
              "\n\n",sep=""),0);
}
else{titre("No Multiplicative Term\n",
                              0)}
sepa(lar,sep);cat(sep);
titre("Term(s)",0,lar[1]+lar[2]);
```

```
cat(sep);
titre("Source    ",0,lar[3]);cat(sep);
titre("P.Dim.",0,lar[4]);cat(sep);
titre("S.S.    ",0,lar[5]);cat(sep);
titre("M.S.    ",0,lar[6]);cat(sep);
cat("\n");
sepa(lar,sep);
for (jbd in 1:3)
     ligne(BB[[jbd]],jbd,lar,fmt,sep);
if (carac[7] > 0)
            {for (jbd in 1:carac[7]) {
  UU <- list(paste(BB[[4]][[1]],
                        " [",jbd,"]"));
  UU[[2]] <- carac[5] + carac[6] -
                      1 - (jbd - 1)*2;
  UU[[3]] <-
        BB[[4]][[7]][[2]][jbd,jbd]^2;
  ligne(UU,4,lar,fmt,sep);
  }}
for (jbd in 5:8)
     ligne(BB[[jbd]],jbd,lar,fmt,sep);
sepa(lar,sep);
lignegr(BB,1:3,9,lar,fmt,sep);
lignegr(BB,4,10,lar,fmt,sep);
lignegr(BB,4:5,11,lar,fmt,sep);
lignegr(BB,4:7,12,lar,fmt,sep);
lignegr(BB,4:8,13,lar,fmt,sep);
sepa(lar,sep);
cat(" (Term(s) with null degrees of
freedom are eliminated,\n");
cat("     for more details see the
manual)\n");

NULL
}

####################
printBR2 <- function(Bu,quoi=1:5)
{
# prints the "Bu", one term of "BB"
# "BB" being an output of "BR"
#
if (sum(quoi == 1)>0) {
  titre(Bu[[1]],5);
  }
#
if (sum(quoi == 2)>0) {
  repete(" ",4);
  cat("\"DoF\" = ",Bu[[2]]);
  }
#
if (sum(quoi == 3)>0) {
  repete(" ",4);
  cat(" S.S. = ",Bu[[3]]);
  if (Bu[[2]] > 0) {
    repete(" ",4);
    cat(" M.S. = ",Bu[[3]]/Bu[[2]]);
    }
  }
if (sum((quoi == 3) +
      (quoi == 2))> 0) repete("\n",2);
#
if (sum(quoi == 4)>0) {
  titre("Fitted Term",1);
  print(Bu[[4]]);
  }
#
if (sum(quoi == 5)>0) {
  if (Bu[[5]]>0)
          { for (jbd in 1:Bu[[5]]) {
    titre(Bu[[5+jbd]][[1]],1);
    print(Bu[[5+jbd]][[2]])
        }}
    }

}


####################
ligne <-function(llii,jbd,lar,fmt,sep)
{# prints an anova table line
  if (!(is.null(llii[[2]]))) {
  if (llii[[2]]>0) {cat(sep);
    titre(jbd,0,lar[1]);
    repete(" ",lar[2]);cat(sep);
    titre(llii[[1]]        ,0,lar[3]);
    cat(sep);
    titre(V(llii[[2]],0)   ,0,lar[4]);
    cat(sep);
    titre(V(llii[[3]],fmt[1])   ,0,
                    lar[5]);cat(sep);
    titre(V(llii[[3]]/llii[[2]],
                  fmt[2]),0,lar[6]);
    cat(sep);cat("\n");
    }}
}

####################
sepa <- function(lar,sep)
{# prints a separating line
  cat(sep);
  repete("-",lar[1]+lar[2]);cat(sep);
  repete("-",lar[3]);cat(sep);
  repete("-",lar[4]);cat(sep);
  repete("-",lar[5]);cat(sep);
  repete("-",lar[6]);
  cat(sep);cat("\n");
}

####################
lignegr <-
      function(BB,gri,sd,lar,fmt,sep)
{
  UU <- vector("list",3);
  UU[[1]] <- NULL;
  UU[[1]] <- paste(gri,collapse=" + ");
  x <- 0; y <- 0;
  for (jbd in gri) {
    x <- x + BB[[jbd]][[2]];
    y <- y + BB[[jbd]][[3]];
    }
  UU[[2]] <- x ;
  UU[[3]] <- y;
  ligne(UU,sd,lar,fmt,sep);
}

####################
DONNEES <- function(titre,Y,
         RML=NULL, RMB=diag(nrow(Y)),
         CML=NULL, CMB=diag(ncol(Y)),
         ad1r=T,ad1c=T,
         cenL=T,norL=T,
         cenB=F,norB=F)
{
# compacting dependent
# and independent data
# in a single structure
# before calling BR
#
# "titre" is a character string
#              qualifying the data set
# "Y" is the matrix of dependent
#                            variates
# "RML" matrix whose columns are the
```

```
#       row covariates to be used
#       linearly (NULL if none)
# "RMB" matrix whose columns are the
#       row covariates to be used
#       bilinearly (NULL if none)
# "CML" matrix whose columns are the
#       column covariates to be used
#       linearly (NULL if none)
# "CML" matrix whose columns are the
#       column covariates to be used
#       bilinearly (NULL if none)
# "ad1r" if the constant covariate
#       must be added in the linear
#       part for rows
# "ad1c" if the constant covariate
#       must be added in the linear
#       part for columns
# "cenL" if linear covariates must
#       be centred
# "norL" if linear covariates must
#       be normalized
# "cenB" if bilinear covariates must
#       be centred
# "norB" if bilinear covariates must
#       be normalized
#
# dimnames are automatically
# incorporated if not existing
# those from "Y" are first considered
d <- vector("list",6);
d$titre <- titre;
d$Y     <- Y;
if (!is.null(RML)) d$RML <-
           transfor(RML,cenL,norL);
if (!is.null(RMB)) d$RMB <-
           transfor(RMB,cenB,norB);
if (!is.null(CML)) d$CML <-
           transfor(CML,cenL,norB);
if (!is.null(CMB)) d$CMB <-
           transfor(CMB,cenB,norB);
irfa <- didino(Y,"R_");
icfa <- didino(Y,"C_",T);
ilrc <- didino(d$RML,"LRC_",T);
ilcc <- didino(d$CML,"LCC_",T);
ibrc <- didino(d$RMB,"BRC_",T);
ibcc <- didino(d$CMB,"BCC_",T);
dimnames(d$Y)    <- list(irfa,icfa);
if(is.matrix(d$RML))
    dimnames(d$RML) <- list(irfa,ilrc);
if(is.matrix(d$RMB))
    dimnames(d$RMB) <- list(irfa,ibrc);
if(is.matrix(d$CML))
    dimnames(d$CML) <- list(icfa,ilcc);
if(is.matrix(d$CMB))
    dimnames(d$CMB) <- list(icfa,ibcc);
if (ad1r) {
  if(is.matrix(d$RML))
  { d$RML <- cbind(1,d$RML);
    dimnames(d$RML)[[2]][1] <- "-1-";}
  else {d$RML <- matrix(1,nrow(Y));
 dimnames(d$RML) <- list(irfa,"-1-");}
  }
if (ad1c) {
  if(is.matrix(d$CML))
   { d$CML <- cbind(1,d$CML);
    dimnames(d$CML)[[2]][1] <- "-1-";}
  else {d$CML <- matrix(1,ncol(Y));
 dimnames(d$CML) <- list(icfa,"-1-");}
  }
d
}
```

```
####################
BR <- function(d,nbmul=1,epsilon=1e-5)

 {
# "BR" is the equivalent of "biareg"
#      when data set "Y" is complete
#      and weight matrix "WY" is the full
#      one matrix
#
# "d" is the data set most often
#      produced by "DONNEES"
# "nbmul" is the number of multipl.
#      terms to include in the model
# "epsilon" is a small value used for
#      detecting colinearities
#
# results are returned by the mean of
#  a list of 10 components, mainly
#  lists. Each one of the first 8
#  corresponds to a term of the model.
#  They are described below.
#  the 9th list gives the dimensions
#  the 10th one gives the title
#      copied from "d"
#
# each of the first eight list
#   comprises a different number
#   of list according to its nature:
#     1. title
#     2. parametric dimension (and
#          nothing more when null)
#     3. associated sum of squares
#     4. fitted term
#     5. number of matrix of
#        parameters (say "nmp")
#     5+1 until 5+nmp : list for each
#         parameter matrix. These lists
#         are composed of two elements:
#           a. a title
#           b. the estimated matrix
#          [c. the approximated variance
#            when parameters stacked
#            by columns...]
#
#   Have a look to the code and
#    my RSA paper to better understand
#
res <- vector("list",10);
for (jbd in 1:8)
        res[[jbd]] <- vector("list",5)
res[[9]] <- vector("numeric",7);
res[[10]] <- d$titre;
#
# checking
#
Check(d,nbmul,epsilon);
#
# preparing the identifiers
#
irfa <- didino(d$Y,"R_");
icfa <- didino(t(d$Y),"C_");
if (!is.null(d$RML))
     ilrc <- didino(t(d$RML),"LRC_");
if (!is.null(d$CML))
     ilcc <- didino(t(d$CML),"LCC_");
if (!is.null(d$RMB))
     ibrc <- didino(t(d$RMB),"BRC_");
if (!is.null(d$CMB))
     ibcc <- didino(t(d$CMB),"BCC_");
imul <- didi(nbmul,"M_");
reste <- d$Y;
```

```
#
# identifying the 8 terms
#
res[[1]][[1]] <- "lin. common term";
res[[2]][[1]] <- "lin. col cov.";
res[[3]][[1]] <- "lin. row cov.";
res[[4]][[1]] <- "bil. cov."
res[[5]][[1]] <- "bil. cov. rem."
res[[6]][[1]] <- "bil. col cov. rem."
res[[7]][[1]] <- "bil. row cov. rem."
res[[8]][[1]] <- "all cov. reminder"
#
# assigning parametric dimensions
#
I <- nrow(d$Y); J <- ncol(d$Y);
if (is.null(d$RML)) {Kl <- 0}
else {Kl <- ncol(d$RML)};
if (is.null(d$CML)) {Hl <- 0}
else {Hl <- ncol(d$CML)};
if (nbmul > 0) {
  if (is.null(d$RMB)) {Kb <- 0}
  else {Kb <- rang(cbind(d$RML,
                    d$RMB)) - Kl};
  if (is.null(d$CMB)) {Hb <- 0}
  else {Hb <- rang(cbind(d$CML,
                    d$CMB)) - Hl};
  }
else {
  Kb <- 0; Hb <- 0;
  }
res[[1]][[2]] <- Kl * Hl;
res[[2]][[2]] <- (I-Kl) * Hl;
res[[3]][[2]] <- Kl * (J-Hl);
res[[5]][[2]] <- (Kb - nbmul) *
                  (Hb - nbmul);
res[[4]][[2]] <-  Kb * Hb -
                  res[[5]][[2]];
res[[6]][[2]] <- (I-Kl-Kb) *
                      Hb;
res[[7]][[2]] <-       Kb  *
                  (J-Hl-Hb);
res[[8]][[2]] <- (I-Kl-Kb) *
                     (J-Hl-Hb);
for (jbd in 1:8) {
  res[[jbd]][[3]] <- 0
  res[[jbd]][[5]] <- 0
  }
res[[9]] <- c(I,J,Kl,Hl,Kb,Hb,nbmul);
#
# fitting additive terms
#
### <<< following lines modified
###     on 29/3/1998 >>>
if (Kl == 0) {PG <- NULL}
else {PG <- projcolq(d$RML)}
if (Hl == 0) {PD <- NULL}
else {PD <- projcolq(d$CML)}
### <<< end of modification >>>
ax <- 1;
if (res[[ax]][[2]] > 0) {
  res[[ax]][[4]] <-
       PG %*% d$Y %*% PD;
  reste <- reste - res[[ax]][[4]];
  dimnames(res[[ax]][[4]]) <-
                 list(irfa,icfa);
  res[[ax]][[3]] <-
           sum(res[[ax]][[4]]^2);
  res[[ax]][[5]] <- 1;
  res[[ax]][[6]] <- vector("list",2);
  res[[ax]][[6]][[1]] <- "Parameters"
  res[[ax]][[6]][[2]] <-
      projparq(d$RML) %*% d$Y %*%
                       t(projparq(d$CML));
  dimnames(res[[ax]][[6]][[2]]) <-
                          list(ilrc,ilcc);
  }
ax <- 2;
if (res[[ax]][[2]] > 0) {
  res[[ax]][[4]] <- d$Y %*% PD;
  dimnames(res[[ax]][[4]]) <-
                        list(irfa,icfa);
  if (res[[1]][[2]] > 0) {
  res[[ax]][[4]] <- res[[ax]][[4]] -
                        res[[1]][[4]]}
  reste <- reste - res[[ax]][[4]];
  res[[ax]][[3]] <-
                 sum(res[[ax]][[4]]^2);
  res[[ax]][[5]] <- 1;
  res[[ax]][[6]] <- vector("list",2);
  res[[ax]][[6]][[1]] <- "Parameters"
  res[[ax]][[6]][[2]] <-
  res[[ax]][[4]] %*% t(projparq(d$CML));
  dimnames(res[[ax]][[6]][[2]]) <-
                          list(irfa,ilcc);
  }
ax <- 3;
if (res[[ax]][[2]] > 0) {
  res[[ax]][[4]] <- PG %*% d$Y;
  dimnames(res[[ax]][[4]]) <-
                        list(irfa,icfa);
  if (res[[1]][[2]] > 0)
   {res[[ax]][[4]] <- res[[ax]][[4]] -
                        res[[1]][[4]]}
  reste <- reste - res[[ax]][[4]];
  res[[ax]][[3]] <-
                 sum(res[[ax]][[4]]^2);
  res[[ax]][[5]] <- 1;
  res[[ax]][[6]] <- vector("list",2);
  res[[ax]][[6]][[1]] <- "Parameters"
  res[[ax]][[6]][[2]] <-
 t(projparq(d$RML) %*% res[[ax]][[4]]);
  dimnames(res[[ax]][[6]][[2]]) <-
                        list(icfa,ilrc);
  }


#
# fitting the biadditive part
#

if (nbmul > 0) {

  ### <<< following lines modified
  ###     on 29/3/1998 >>>
  if (Kb > 0) {
   if (Kl == 0) {
    PG <- projcolq(d$RMB)
    }
   else {
    PG <- projcolq(cbind(d$RML,d$RMB)) -
          projcolq(d$RML);
    }
   }
  if (Hb > 0) {
   if (Hl == 0) {
    PD <- projcolq(d$CMB)
    }
   else {
    PD <- projcolq(cbind(d$CML,d$CMB)) -
          projcolq(d$CML);
    }
   }
  ### <<< end of modification >>>
  if ((I-Kb) > 0) PGG <- diag(I) -
      projcolq(cbind(d$RML,d$RMB))
```

```
   if ((J-Hb) > 0) PDD <- diag(J) -
        projcolq(cbind(d$CML,d$CMB))

   ax <- 4;
   if (res[[ax]][[2]] > 0) {
     bx <- svd(PG %*% reste %*% PD);
     mm <- 1:nbmul;
     res[[ax]][[4]] <-
              t(bx$v[,mm,drop=F] %*%
  (t(bx$u[,mm,drop=F]) * bx$d[mm]));
     dimnames(res[[ax]][[4]]) <-
                    list(irfa,icfa);
     res[[ax]][[3]] <-
              sum(res[[ax]][[4]]^2);
     res[[ax]][[5]] <- 3;
 res[[ax]][[6]] <- vector("list",2);
 res[[ax]][[7]] <- vector("list",2);
 res[[ax]][[8]] <- vector("list",2);
 res[[ax]][[6]][[1]] <- "Row Scores";
 res[[ax]][[6]][[2]] <-
 projparq(d$RMB) %*% bx$u[,mm,drop=F];
 dimnames(res[[ax]][[6]][[2]]) <-
                     list(ibrc,imul);
 res[[ax]][[7]][[1]] <-
                      "SVD value(s)";
 res[[ax]][[7]][[2]] <-
           diag(bx$d[mm],nrow=nbmul);
 dimnames(res[[ax]][[7]][[2]]) <-
                     list(imul,imul);
 res[[ax]][[8]][[1]] <-
                     "Column Scores";
 res[[ax]][[8]][[2]] <-
 projparq(d$CMB) %*% bx$v[,mm,drop=F];
 dimnames(res[[ax]][[8]][[2]]) <-
                     list(ibcc,imul);
 }

 ax <- 5;
 if (res[[ax]][[2]] > 0) {
   res[[ax]][[4]] <-
   (PG %*% reste %*% PD) -
             res[[4]][[4]];
   dimnames(res[[ax]][[4]]) <-
                   list(irfa,icfa);
   res[[ax]][[3]] <-
             sum(res[[ax]][[4]]^2);
   res[[ax]][[5]] <- 0;
   }

 ax <- 6;
 if (res[[ax]][[2]] > 0) {
   res[[ax]][[4]] <-
              PGG %*% reste %*% PD;
   dimnames(res[[ax]][[4]]) <-
                   list(irfa,icfa);
   res[[ax]][[3]] <-
             sum(res[[ax]][[4]]^2);
   }

 ax <- 7;
 if (res[[ax]][[2]] > 0) {
   res[[ax]][[4]] <-
               PG %*% reste %*% PDD;
   dimnames(res[[ax]][[4]]) <-
                    list(irfa,icfa);
   res[[ax]][[3]] <-
              sum(res[[ax]][[4]]^2);
   }

   reste <- PGG %*% reste %*% PDD;
   }
```

```
 ax <- 8;
 if (res[[ax]][[2]] > 0) {
   res[[ax]][[4]] <- reste
   dimnames(res[[ax]][[4]]) <-
                   list(irfa,icfa);
   res[[ax]][[3]] <-
              sum(res[[ax]][[4]]^2);

 }
 res
 }
```

## D.2 basic functions

```
####################
transfor <- function (M,cen=T,nor=T)
{# returns "M" after centring (if cen)
 # and normalizing (if nor)
 # its columns
if(is.matrix(M)){
  if (cen) {
   res <- M - outer(rep(1,nrow(M)),
                  apply(M,2,mean),"*");
   dimnames(res) <- dimnames(M);
   }
  else {
   res <- M;
   }
  if (nor) {
   res <- res %*%
         diag(1/sqrt(apply(M,2,var)),
                      nrow=ncol(M));
   dimnames(res) <- dimnames(M);
   }}else{res<-NULL}
res
}

####################
repete <- function(cha,nb=10)
{
# prints a character "nb" times "cha"
if (nb>0) {for (jbd in 1:nb) cat(cha)}
NULL
}

####################
centring <- function(n)
{
# returns the centring matrix
# of size n
res <- diag(n) - 1/n;
res;
}

##############################
V <- function(nombres,d=3)
{
# displaying fixed decimal numbers

v <- round(nombres*(10^d))/(10^d);
```

```
v
}

####################
titre <- function(cha,type=1,J=0)
{
# prints a character string "cha"
# with more or less emphasis
# when "type" == 0 then (except if 0)
# "J" gives the minimum number of
#                             characters
if (type == 0) {
  if ((J>0) & (J>nchar(cha)))
  { repete(" ",J-nchar(cha))}
  cat(cha);
  }
if (type == 1) {
  repete("\n",1); repete(" ",1);
  cat(cha);
  repete("\n",2);
  }
if (type == 2) {
  repete("\n",2); repete(" ",1);
  cat(cha);
  repete("\n",3);
  }
if (type == 5) {
  repete("\n",2); repete(" ",3);
  cat(cha);
  repete("\n",1);repete(" ",3);
  repete("=",nchar(cha));
  repete("\n",2);
  }
if (type == 10) {
  repete("\n",1);repete(" ",3);
  repete("=",nchar(cha));
  repete("\n",1); repete(" ",3);
  cat(cha);
  repete("\n",1);repete(" ",3);
  repete("=",nchar(cha));
  repete("\n",2);
  }
NULL
}

####################
projcolq <- function(A,eps=1e-8)
{
# "A" can be a singular matrix.
# Returns the matrix of projection
# onto the columns of "A" through
# singular value decomposition
s <- svd(A);
s$u <- s$u[,(s$d/s$d[1]>eps),drop=F];
s$u%*%t(s$u);
}

####################
projparq <- function(A,eps=1e-8)
{
# derived from "projcolq"
# for parameter estimation
s <- svd(A);
s$u <- s$u[,(s$d/s$d[1]>eps),drop=F];
s$v <- s$v[,(s$d/s$d[1]>eps),drop=F];
s$d <- s$d[(s$d/s$d[1]>eps)];
# s$v%*%diag(s$d)%*%t(s$u);
t(s$u %*% (t(s$v) * s$d))
}

####################
trac <- function(A)
{
# returns the trace of "A"
 A <- as.matrix(A)
 if (nrow(A) != ncol(A))
       {stop("trac: A is not squared")}
 sum(diag(A))
}

####################
didi <- function(n,ide="id")
{
# producing standard "n" dimnames
if (n < 1) {li <- NULL}
else {li <- paste(ide,1:n,sep="")};
li
}

####################
didino <- function(A,ide="r",transpo=F)
{
# giving standard dimnames to "A" rows
# when it does not exist
#
# for columns put "transpo"
 if (is.matrix(A)) {
   if (transpo) A <- t(A);
   li <- dimnames(A)[[1]];
   if (is.null(li))
      {li <- didi(dim(A)[1],ide)}
   }
 else { li <- NULL }

li
}

####################
dino <- function(A,iro="r",ico="c")
{
# an always already version of dimnames
# for matrix "A"

M<-list(didino(A,iro),didino(t(A),ico))

 M
}

####################
rang <- function(A,tol=1e-10)
{
# returns the rank of "A"

 if (nrow(A) < ncol(A)) { A <- t(A) }
 A <- t(A) %*% A
 eps <- tol * trac(A)
 v <- eigen(A)$values
 sum (v > eps)
}

####################
Check <- function(d, nbmul,eps)
{
# Checks covariate matrices for
dimensions
#                           and redundancies
#
# arguments are described in "biareg"
#
# If any constraint is not fulfilled,
#      the program stops and an error
# message is returned.
#
```

```
 if (is.null(d$RML))
       {Kl <- 0} else {Kl <- ncol(d$RML)};
 if (is.null(d$CML))
       {Hl <- 0} else {Hl <- ncol(d$CML)};
 if (is.null(d$RMB)){Kb <- 0} else
    {Kb <- rang(cbind(d$RML,d$RMB)) - Kl};
 if (is.null(d$CMB)) {Hb <- 0} else
    {Hb <- rang(cbind(d$CML,d$CMB)) - Hl};


 if (nbmul < 0 )
  stop("ERREUR: nbmul cannot be
 negative");

 if (Kl > 0) {
    if(nrow(d$RML)!=nrow(d$Y))
    stop(paste("ERREUR: row covariate",
    "matrices must have same number",
    "of rows as Y"));
    if(rang(d$RML,eps)!=ncol(d$RML))
    stop("ERREUR: d$RML has redundancies")
    }

 if (Hl > 0) {
    if(nrow(d$CML)!=ncol(d$Y))
    stop(paste("ERREUR: column covariate",
    "matrices must have same number of",
    "columns as Y"));
    if(rang(d$CML,eps)!=ncol(d$CML))
    stop("ERREUR: d$CML has redundancies")
    }
 if (Kb == 0) {
    if (nbmul > 0) stop(paste("ERREUR:
no",
    " row covariates => nbmul = 0"));
    }

 if (Hb == 0) {
    if (nbmul > 0) stop(paste("ERREUR:
no",
    "column covariates => nbmul = 0"));
    }

 if (Kb * Hb > 0) {
   if (nbmul > min((nrow(d$Y)-Kl),
                          (ncol(d$Y)-Hl)))
   stop(paste("ERREUR: nbmul should not",
   "exceed nrow(d$Y)-ncol(d$RML),",
   "or ncol(d$Y)-ncol(d$CML)"));
   }

 if (nbmul > 0) {
  if(nbmul > min(ncol(d$RMB),ncol(d$CMB)))
  stop(paste("ERREUR: nbmul must not",
    " exceed the number of columns of",
    "RMB or CMB"));
    if (nrow(d$RMB)!=nrow(d$Y))
    stop(paste("ERREUR: row covariate",
    "matrices must have same number of",
    "rows as Y"));
    if(rang(d$RMB,eps)!=ncol(d$RMB))
    stop("ERREUR: RMB has redundancies")
    if (nrow(d$CMB)!=ncol(d$Y))
    stop(paste("ERREUR: column covariate",
         "matrices must have same number",
         "of columns as Y"));
   if(rang(d$CMB,eps)!=ncol(d$CMB))
    stop("ERREUR: CMB has redundancies")
   }

if (sum(is.na(d$Y)) != 0)
 { stop("ERREUR: No missing values !") }

NULL
}
```