Inference for CUB models: a program in R

Maria Iannario and Domenico Piccolo

Department of Political Sciences, University of Naples Federico II, Italy

maria.iannario@unina.it; domenico.piccolo@unina.it

Abstract

An alternative approach has been proposed for the analysis and the modelling of ordinal data: it is based on the psychological process by which a respondent expresses his/her evaluation about the item with an inherent indecision. This class of models has been developed with many variants and it is now indicated as CUB models. The purpose of this paper is to introduce users to the version 4.0 of a program, written in the R statistical environment, to make effective applications of CUB models and variants by exploiting their capabilities both from computational and graphical points of view. After a specification of the different structures, the basic commands are presented with some examples. Generalizations and extensions of the standard models are also mentioned. For a more extensive study a bibliographic note concludes the paper.

Key Words: Ordinal data, CUB models, CUBE models, Shelter effect, G*e*CUB models, CUSH models, IHG models

1 Introduction

In several applied researches, data are collected as categorical ordinal observations (Agresti, 2010; Tutz, 2012). Sometimes they are actually ordered (as in judgements, preferences, degree of adhesion to a sentence, etc.) whereas, in other circumstances, they are categorized for convenience (age of people in classes, measures of objects in block of constant size, blood pressure for classifying heart health status, political ideology, etc.) as discussed by Anderson (1984). It is possible to consider also ranks as ordinal data if we interpret the ranks of a single object as an ordered evaluation. Caution is necessary in interpreting ranks of related objects since these evaluations are not independent (D'Elia and Piccolo, 2005).

The program[1] we are going to introduce is a statistical software coded in the ***R*** environment

[1] The version 4.0 of the CUB program is freely available from Authors upon request and may be downloaded at www.labstat.it/home/wp-content/uploads/2014/11/CUB40.R

able to specify, estimate and test a large class of parametric structures, generated by the family of CUB models. In fact, in the standard option, these models are a (convex) *C*ombination of discrete *U*niform and shifted *B*inomial random variables.

The software is organized on the basis of primary functions (able to perform the general purpose of building statistical models) which in turn call for several other functions (in charge of limited and specific objectives). The program is presented as a large script; thus, users can apply also a subset of the available functions, if necessary and/or convenient for different goals. In most cases the output of the functions shows graphical displays and a list of indicators/tests to check the usefulness and the significance of the estimates. Maximum Likelihood (ML) method is applied to the statistical procedures (estimation and test) for all models built in this program. In most cases convergence is achieved by means of the EM procedure.

After running a primary function for estimation and testing of a model, several quantities are available in the computer memory. In this way, the main information about the estimated models may be saved and maintained for further elaborations and/or model comparisons. Then, simplified version of the main estimation routines are proposed and they are especially useful when simulation runs are performed and only few values are necessary to accelerate the performance of the code.

This paper is organized as follows: in the next Section the structure of the program and some preliminary concepts are examined whereas in Section 3 data input is described. Then, Sections 4-7 are devoted to the presentation of different models and to their corresponding commands. Inferential issues, simulation routines and plotting facilities are pursued in Sections 8, 9 and 10, respectively. Examples are illustrated in Section 11 and several area of applications of CUB models are listed in Section 12. A bibliographic note ends the paper.

2 Structure of the program

The programm is a long script, adequately commented and subdivided as follows:

I General functions

II Probability distributions

III Log-Likelihood functions

IV Variance-covariance matrix of parameter estimates

V Initial values of estimates

VI Plotting facilities

VII Simulation routines

VIII CUB models functions

IX Main calls for CUB, CUBE, CUSH and IHG models

In each part, several functions are described and macro-functions (for instance, CUB) are defined. The different parameters involved in the models are denoted as: `pai` (π), `csi` (ξ), `phi` (ϕ), `delta` (δ), `bet` (β), `gama` (γ), `omega` (ω). The variance-covariance matrix of estimates is designed as `varmat`. Notice that *only* the primary functions of the program are capitalized: CUB, CUBE, CUSH, IHG. All other functions are denoted with lower case letters.

This paper and the program assume a discrete Uniform random variable as the building block useful to account for uncertainty in the responses. Alternative solutions are possible leading to CUB models with a varying uncertainty (=VCUB models) as introduced by Gottard, Iannario and Piccolo (2015). Hereafter, we will not discuss this further option; however, we point out that the introduction of a different free-parameter distribution for uncertainty requires a limited change in the code of the current software.

As a general criterion, the number of ordinal categories to be considered, denoted by m, is a global variable and *must* be *always* specified at the beginning of the running of the program. It is not safe to assume in any case that m is the maximum of the responses since it is possible that no respondent chooses the highest value of the support.

To start, it is sufficient to run the following commands in the directory where the main program is resident:

```
> source("CUB.R")
> m=number-of-ordinal-categories
```

In addition, it is also possible to include the value of m in the main commands as follows:

```
> CUB(ordinal,m=numer_of_ordinal_categories)
```

Finally, parameter estimates, variance-covariance matrix of the estimates and log-likelihood value at the maximum are obtained by using `*$estimates`, `*$varmat`, `*$loglik` as in the following example:

```
> m=numer_of_ordinal_categories
> model=CUB(ordinal)
> estim=model$estimates
> varest=model$varmat
> maxlik=model$loglik
```

When the estimation algorithm is iterative also the number of iteration can be retrieved by using `*$niter`.

3 Data input

Generally, ordinal data are available as a sample of n ratings $\boldsymbol{r} = (r_1, r_2, \dots, r_n)'$, where $r_i \in \{1, 2, \dots, m\}$ for a given $m > 3$. Thus, the observations $\boldsymbol{r}$ are realizations of a random variable R and are available as a vector `ordinal` in the ***R*** environment. The same is true for possible covariates which we introduce to explain responses and improve the fitting. If the input is a data frame, ordinal data and (concomitant) covariates should be conveniently designed as vector and matrices, as in the following code:

```
> dati=read.table("...:/.../...",header=T)
> m=...                 # specification of m as a global variable
> ordinal=dati[,j]     # if ordinal data are in the j-th column

> covar=dati[,lista]  # if covariates are listed in the columns
                       #  specified by 'lista'
```

Sometimes, if results of a survey are presented by means of a table, the ratings are available as aggregated absolute frequencies $(n_1, n_2, ..., n_m)'$ in a vector `frequencies`, and they must be expanded to generate a vector of length $n = n_1 + n_2 + ... + n_m$ in order to run the primary functions (for instance, CUB). An example of this code is:

```
> m=5                              # specification of m=5
> frequencies=c(12,18,30,32,8)     # absolute frequencies
> ordinal=rep(1:m,frequencies)     # a vector of ordinal data
> CUB(ordinal)
```

Notice that `frequencies` must be a vector of length m even if some observed frequencies are equal to 0.

Finally, this program does not allow an automatic processing of possible missing values; several softwares with different approaches are available in the literature to impute missing values: see Honaker, King and Blackwell (2011), for instance. Indeed, CUB models are an effective alternative for the imputation of missing data by substituting the modal value of the estimated CUB model for the complete data set, as shown by Cugnata and Salini (2014).

When missing values are present, to drop them from a vector of ordinal data (`ordinal`) the following code may be applied:

```
> newordinal=na.omit(ordinal)
```

In presence of missing values, special care should be used if the function implies both ordinal data and a matrix of covariates since the pattern of missing values may be not homogeneous. Thus, preliminary analysis should be performed to get vectors and matrices with full and comparable information.

4 CUB models

The starting point of the new modelling paradigm for ordinal data is a CUB random variable R defined as the mixture of a shifted Binomial and a discrete Uniform distribution over the support $\{1, 2, \dots, m\}$, for a given $m > 3$ (Piccolo, 2003). Thus, for $j = 1, 2, \dots, m$,

$$Pr\,(R = j \mid \pi,\, \xi) = \pi \binom{m-1}{j-1} \xi^{m-j}\,(1-\xi)^{j-1} + (1-\pi)\,\frac{1}{m}\,. \tag{1}$$

Then, for any given $m > 3$, a CUB model is fully specified when the parameter vector $\boldsymbol{\theta} = (\pi, \xi)'$ is known.

If we consider either or both parameters as functions of subjects' covariates (and we assume a logistic link between them, for instance), for $j = 1, \dots, m$ and $i = 1, \dots, n$, then the *stochastic* and the *systematic* components of a CUB model with covariates are defined by

$$\begin{cases} Pr\,(R_i = j \mid \boldsymbol{\beta}, \boldsymbol{\gamma}) \;\; = \;\; \pi_i\, b_j(\xi_i) + (1 - \pi_i)\, \dfrac{1}{m}\,; \\ \\ logit\,(\pi_i) = \boldsymbol{y}_i \boldsymbol{\beta}\,; \qquad\qquad logit\,(\xi_i) = \boldsymbol{w}_i \boldsymbol{\gamma}\,; \end{cases} \tag{2}$$

where $b_j\,(\xi_i) = \binom{m-1}{j-1} \xi_i^{m-j}\,(1 - \xi_i)^{j-1}$, $j = 1, \dots, m$ is the shifted Binomial distribution and $\boldsymbol{y}_i$ and $\boldsymbol{w}_i$ are the i-th rows of the matrices $\boldsymbol{Y}$ and $\boldsymbol{W}$ which contain the subjects' covariates for explaining π_i and ξ_i, respectively. In model (1) we define $\boldsymbol{\beta} = (\beta_0, \beta_1, \dots, \beta_p)'$ and $\boldsymbol{\gamma} = (\gamma_0, \gamma_1, \dots, \gamma_q)'$. The program automatically adds a column of $1s$; thus, the relevant matrices $\boldsymbol{Y}$ and $\boldsymbol{W}$ to be specified by the user *do not contain* a column of constant.

For a given m, a CUB model with covariates is fully specified when the parameter vector $\boldsymbol{\theta} = (\boldsymbol{\beta}', \boldsymbol{\gamma}')'$ is known.

For making more immediate the interpretation of a CUB model, *uncertainty* and *feeling* may be related to $(1 - \pi_i)$ and $(1 - \xi_i)$, respectively. So, the relationships between parameters and covariates are more conveniently expressed as:

$$logit\,(1 - \pi_i) \; = \; -\boldsymbol{y}_i \boldsymbol{\beta}\,; \qquad logit\,(1 - \xi_i) \; = \; -\boldsymbol{w}_i \boldsymbol{\gamma}\,. \tag{3}$$

In this regard, we observe that the selection of covariates for *uncertainty* and/or *feeling* is a relevant issue which is currently under investigation; the proposed strategies include backward-forward approaches and penalized likelihood methods.

In some papers, CUB models without covariates, with covariates only for π_i, with covariates only for ξ_i, with covariates for both π_i and ξ_i have been designed as CUB$(0, 0)$, CUB$(p, 0)$, CUB$(0, q)$, CUB(p, q) models, respectively.

If ordinal data are available in the vector `ordinal`, to build a CUB model without covariates, it is sufficient to run:

```
> m=number-of-ordinal-categories

> CUB(ordinal)
```

When matrices $\boldsymbol{Y}$ and $\boldsymbol{W}$ contain the selected columns of subjects' covariates for *uncertainty* (parameters $1-\pi_i$) and *feeling* (parameters $1-\xi_i$), respectively, the commands to build a CUB model with covariates are the following:

```
> CUB(ordinal,Y=paicov)            # if paicov includes
                                   # covariates for uncertainty


> CUB(ordinal,W=csicov)            # if csicov includes
                                   # covariates for feeling


> CUB(ordinal,Y=paicov, W=csicov) # if paicov and csicov
                                   # include covariates for
                                   # uncertainty and feeling
```

If covariates are obtained by manipulations or they derive from different data sets it is possible to `bind` them (if all vectors have the same length as `ordinal`) as in the following example:

```
> x1=dat1[,1]; x2=log(dat2[,3]); x3=1:n        # covariates


> CUB(ordinal,Y=cbind(x1,x2),W=cbind(x3,x1*x2) # CUB model
                                               # with covariates
                                               # for uncertainty
                                               # and feeling
```

After the estimation of a CUB model without covariates, a plot of the observed relative frequencies and the expected distribution is automatically generated, except when the command `CUB(ordinal,makeplot=FALSE)` is performed.

In presence of covariates, the output does not include plots, except when a single dichotomous covariate `dum` (*strictly* defined with values $0, 1$) is introduced to explain *uncertainty* and/or *feeling*. In these circumstances, an automatic plot is produced to show the estimated probability distributions conditioned on `dum=0` (circled) and `dum=1` (dotted), respectively. Typical commands are:

```
> dum=ifelse(Gender=="male",0,1) # dichotomous covariate for Gender

> CUB(ordinal,Y=dum)             # CUB model (Gender for uncertainty)

> CUB(ordinal,W=dum)             # CUB model (Gender for feeling)

> CUB(ordinal,Y=dum,W=dum)       # CUB model (Gender for both
                                 # uncertainty and feeling)
```

To visualize an estimated CUB model in the parameter space, with an asymptotic confidence ellipse around the estimates, the `library(ellipse)` should be loaded. Then, to draw a 95% confidence ellipse for the parameter vector (π, ξ), we require the variance-covariance matrix of estimates (denoted as `varmat` in the output of the estimation procedure). Thus, the code is the following.

```
> library(ellipse)
> source("CUB.R")
> m=number_of_ordinal_categories
> CUB(ordinal)
> plot(1-pai,1-csi,main="Estimated CUB model for ordinal",
        xlim=c(0,1),ylim=c(0,1),
        xlab=expression(paste("Uncertainty  ", (1-pi))),
        ylab=expression(paste("Feeling  ", (1-xi))))
> lines(ellipse(varmat,centre=c(1-pai,1-csi)), lwd=2)
```

With a special emphasis on Sensometric analysis, let us consider the H sensory measurements on the K products collected in the $(K \times H)$ matrix:

$$\boldsymbol{Z} = \{z_{kh},\ k = 1, 2, \dots, K;\ \ h = 1, 2, \dots, H\}\,,$$

so that $\boldsymbol{z}_k = (z_{k1}, z_{k1}, \dots, z_{kH})$ is the row vector of the H sensory measurements available for the k-th product, $k = 1, 2, \dots, K$. Then, both subjects' and objects' covariates may be introduced in the framework of CUB models, as successfully experienced by Piccolo and D'Elia (2008); Capecchi and Endrizzi (2015).

In fact, we can jointly consider all K products in a unique CUB model where parameters and subjects' and objects' characteristics are linked by means of:

$$\begin{cases} \pi_{ik} = \dfrac{1}{1+e^{-\boldsymbol{x}_i^{(\pi)}\boldsymbol{\beta}-\boldsymbol{z}_k\,\boldsymbol{\delta}}}; \\ \xi_{ik} = \dfrac{1}{1+e^{-\boldsymbol{x}_i^{(\xi)}\boldsymbol{\gamma}-\boldsymbol{z}_k\,\boldsymbol{\eta}}}; \end{cases} \qquad i=1,2,\dots,n;\; k=1,2,\dots,K. \tag{4}$$

Here, $\boldsymbol{\delta} = (\delta_1,\dots,\delta_K)'$ and $\boldsymbol{\eta} = (\eta_1,\dots,\eta_K)'$ are parameter vectors which measure the impact of the characteristics of the product on uncertainty and feeling components, respectively. More specifically, according to (4), $1-\pi_{ik}$ $(1-\xi_{ik})$ is related to *uncertainty* (*feeling*) expressed by the i-th subject, whose profile is specified by $\boldsymbol{x}_i^{(\pi)}$ (by $\boldsymbol{x}_i^{(\xi)}$) when faced to the k-th object, whose physical, chemical and organoleptic characteristics are specified by $\boldsymbol{z}_k$. It should be noted that the "intercepts" β_0 and γ_0 of the model (4) contain the joint level effect of the i-th subject and k-th object with regard to *uncertainty* and *feeling*, respectively.

Then, with the same software, it is possible to build CUB models when both subjects and objects' covariates are present, but some preliminary analysis is necessary. First of all, we require conditional independence of the subjects' responses given the objects' covariates. Then, let us assume that ratings are in the vectors `item1,item2, ...` and they are vectorized into `ITEMS`. In addition, subjects' covariates and objects' covariates are expanded into `Xpaitilde` and `Ztilde` for *uncertainty* and into `Xcsitilde` and `Ztilde` for *feeling*, respectively. Finally, the following command:

```
> CUB(ITEMS,Y=cbind(Xpaitilde,Ztilde),W=cbind(Xcsitilde,Ztilde))
```

will generate ML estimates of $\boldsymbol{\theta}$ parameters and related statistics. A correct interpretation of these models should be more immediate if one explicitly considers the estimated relationships for varying subjects' and objects' covariates.

The same commands may be implemented in case of contexts' covariate (Iannario and Piccolo, 2014).

5 CUBE models

To take a possible overdispersion of ordinal data into account, CUBE models (*C*ombination of a discrete *U*niform and a shifted *BE*ta-Binomial random variable) have been proposed (Iannario, 2012b, 2014a,b). The specification and implementation of these models require some extra efforts (especially when covariates are considered) in order to avoid the extremely slow convergence of the EM algorithm towards the ML estimates. In fact, effective initial values are necessary and studies are in progress to improve the current rate of convergence.

Let $\boldsymbol{\theta} = (\pi, \xi, \phi)'$. For a given $m > 4$, we define R a CUBE random variable if its probability mass function is defined by:

$$Pr\,(R = r \mid \boldsymbol{\theta}) = \pi\, g_r(\xi, \phi) + (1-\pi)\,\frac{1}{m}\,, \qquad r = 1, 2, \ldots, m\,, \tag{5}$$

where $g_r(\xi, \phi) = Pr\,(X = r)$ is the distribution of a (shifted) Beta-Binomial random variable X defined on the same support. A CUBE model for a given m is fully specified by $\boldsymbol{\theta} = (\pi, \xi, \phi)'$.

It is convenient to parameterize the distribution $g_r(\xi, \phi)$ as:

$$Pr\,(X = r) = \binom{m-1}{r-1} \frac{\displaystyle\prod_{k=1}^{r} [1 - \xi + \phi(k-1)] \prod_{k=1}^{m-r+1} [\xi + \phi(k-1)]}{[1 - \xi + \phi(r-1)]\,[\xi + \phi(m-r)] \displaystyle\prod_{k=1}^{m-1} [1 + \phi(k-1)]}\,, \tag{6}$$

for $r = 1, 2, \ldots, m$. In this way, if $\phi = 0$ a CUBE model reduces to a CUB model and thus CUB are nested into CUBE models. The selection between a CUB and a CUBE model can be solved by an accurate use of Likelihood Ratio Tests (LRT), since the null hypothesis lies on the borderline of the parameter space (Molenberghs and Verbeke, 2007; Self and Liang, 2003; Vu and Zhou, 1997).

The simplest command to build a CUBE model is:

```
> CUBE(ordinal)                          # essential version
```

Alternatively, the complete version of the command is:

```
> CUBE(ordinal,starting,maxiter,toler,makeplot,expinform)
                                         # complete version
```

With the complete versions, users may suggest better starting values, increase the tolerance to reach a faster convergence and then repeat the running with more accurate starting values. In the default options, the *observed* information matrix is computed to get the asymptotic standard errors of ML estimates; the *expected* version of this matrix is also available (by letting `expinform=TRUE`). In addition, the plot of the observed and fitted distributions may be omitted (by letting `makeplot=FALSE`).

It is possible to introduce subjects' covariates for all components of CUBE models (Piccolo, 2015). More precisely, the *stochastic* and the *systematic components* of a CUBE model with covariates are defined by:

$$\begin{cases} Pr\,(R_i = j \mid \boldsymbol{\theta}) = \pi_i\, h_j(\xi_i, \phi_i) + (1-\pi_i)\,\dfrac{1}{m}\,; \\ \\ logit\,(\pi_i) = \boldsymbol{y}_i\boldsymbol{\beta}\,;\;\; logit\,(\xi_i) = \boldsymbol{w}_i\boldsymbol{\gamma}\,;\;\; \log\,(\phi_i) = \boldsymbol{z}_i\boldsymbol{\alpha}\,; \end{cases} \tag{7}$$

for $j = 1, \dots, m$ and $i = 1, \dots, n$.

We let $\boldsymbol{\theta} = (\boldsymbol{\beta}',\, \boldsymbol{\gamma}', \boldsymbol{\alpha}')'$ whereas $h_j(\xi_i, \phi_i)$ is the (shifted) Beta-Binomial distribution of the *feeling* of the i-th subject which has been parameterized as follows:

$$h_j(\xi_i, \phi_i) = \binom{m-1}{j-1} \frac{\displaystyle\prod_{k=1}^{j}\Big[1-\xi_i+\phi_i\,(k-1)\Big] \prod_{k=1}^{m-j+1}\Big[\xi_i+\phi_i\,(k-1)\Big]}{\Big[1-\xi_i+\phi_i\,(j-1)\Big]\Big[\xi_i+\phi_i(m-j)\Big]\displaystyle\prod_{k=1}^{m-1}\Big[1+\phi_i\,(k-1)\Big]}\,, \tag{8}$$

for $j = 1, \dots, m$ and $i = 1, \dots, n$. If $\phi_i \equiv 0\ \forall i$, we get a CUB model; thus, also CUB models with covariates are nested into CUBE models with the corresponding covariates. A CUBE model with covariates, for a given m, is fully specified by $\boldsymbol{\theta} = (\boldsymbol{\beta}',\, \boldsymbol{\gamma}', \boldsymbol{\alpha}')'$.

The commands to build CUBE models with covariates $\boldsymbol{Y}, \boldsymbol{W}, \boldsymbol{Z}$ for explaining *uncertainty*, *feeling* and *overdispersion*, respectively, are the following:

```
> CUBE(ordinal,Y,W,Z)                    # essential version


> CUBE(ordinal,Y,W,Z,starting,maxiter,toler)

                                         # complete version
```

In the current version of the program, it is also possible to insert covariates only for the parameters ξ_i.

As already mentioned (with special regard to CUBE models with covariates), accurate starting values for CUBE model are indeed necessary, given the lengthy convergence process of the EM algorithm. As an effective strategy, the following steps are suggested:

- start the estimation procedure with a small random subset of the whole sample;
- re-start the estimation procedure on the whole data set and give as starting values those obtained in the small subset experiment with a very high tolerance (`toler=0.1`, say);
- plug these new preliminary estimates in the command to search for the final and more efficient estimates (with a more severe value of `toler`).

In some circumstances, it may be also convenient to reiterate the proposed strategy several times to achieve better and faster results.

6 CUB models with shelter effect

In different fields, a proportion of respondents may choose a category $c \in \{1, 2, \dots, m\}$ which represents a sort of "refuge" to avoid a more demanding selection: this circumstance has been named a *shelter effect* and the corresponding option a *shelter category* (Corduas, Iannario and Piccolo, 2009; Iannario, 2012a; Iannario and Piccolo, 2014). In the family of CUB models this component is effectively estimated, for a known c, by introducing a dummy variable $D_r^{(c)}$ which is 0 but for $(R = c)$ where it assumes value 1. It is defined by the indicator function as $D_r^{(c)} = I(R = c)$; thus, $Pr\left(D_r^{(c)} = c\right) = 1$.

Formally, there are three equivalent formulations of a CUB model with a *shelter effect.*

- (*Extended* CUB model)

$$Pr\,(R = r \mid \boldsymbol{\theta}) = \pi_1\, b_r(\xi) + \pi_2\, \frac{1}{m} + (1 - \pi_1 - \pi_2)\, D_r^{(c)}\,, \tag{9}$$

  where $\boldsymbol{\theta} = (\pi_1,\, \pi_2,\, \xi)'$ is the parameter vector characterizing the distribution of this new mixture random variable. For a given order of components, such models are identifiable for $m > 4$ and require $\pi_1 > 0$, $\pi_2 \geq 0$, $\pi_1 + \pi_2 \leq 1$, $0 \leq \xi \leq 1$. Then, the parameter vector is $\boldsymbol{\theta} = (\pi_1,\, \pi_2,\, \xi)'$.

  In this formulation, the quantity $\delta = 1 - \pi_1 - \pi_2$ measures the added relative contribution of the *shelter choice* at $R = c$ with respect to the standard version of the model. Of course, if $\pi_1 + \pi_2 = 1$ the extended CUB model collapses to the standard one. Instead, if $\pi_2 = 0$ we are considering a mixture of a shifted Binomial distribution and a degenerate probability mass function at $(R = c)$. Finally, if $\pi_2 = 0$ and $\pi_1 \to 0$ the extended model is able to account also for the (rare) situation where most of respondents' choices are concentrated on a single intermediate category.

- (Explicit *shelter* effect)

$$Pr\,(R = r \mid \boldsymbol{\theta}) = \delta \left[D_r^{(c)} \right] \,+\, (1-\delta) \left[\pi^* \, b_r(\xi) + (1-\pi^*)\,\frac{1}{m} \right], \;\; r = 1, 2, , \dots, m. \qquad (10)$$

  This model is equivalent to the previous one thanks to the relationships:

$$\begin{cases} \pi^* &= \dfrac{\pi_1}{\pi_1 + \pi_2}; \\ \delta &= 1 - \pi_1 - \pi_2; \end{cases} \quad \Longleftrightarrow \quad \begin{cases} \pi_1 &= \pi^*(1-\delta); \\ \pi_2 &= (1-\pi^*)(1-\delta)\,. \end{cases}$$

  In this formulation the parameter vector is $\boldsymbol{\theta} = (\pi^*,\, \xi,\, \delta)'$ and it is immediate to quantify the *shelter effect* by means of the parameter δ. Moreover, the modification of uncertainty induced by the introduction of such an effect is evaluated by comparing the π parameter in a standard CUB model (1) with the π^* parameter in the modified model (10).

- (*Satisficing* interpretation)

$$Pr\,(R = r \mid \boldsymbol{\theta}) = \lambda\, b_r(\xi) \,+\, (1-\lambda) \left[\eta\, \frac{1}{m} \,+\, (1-\eta)\, D_r^{(c)} \right], \quad r = 1, 2, , \dots, m. \qquad (11)$$

  For the third formulation the parameter vector is $\boldsymbol{\theta} = (\lambda,\, \xi,\, \eta)'$ and a meditated choice and a lazy selection are clearly separated as the first and the second component of the decision process, respectively. Again, this model is equivalent to the previous ones given the one-to-one relationships:

$$\begin{cases} \lambda &= \pi^*(1-\delta) &= \pi_1; \\ \eta &= \dfrac{(1-\pi)(1-\delta)}{1-\pi(1-\delta)} &= \dfrac{\pi_2}{1-\pi_1}; \end{cases} \quad \Longleftrightarrow \quad \begin{cases} \pi^* &= \dfrac{\lambda}{\lambda + \eta(1-\lambda)}; \\ \delta &= (1-\lambda)(1-\eta)\,. \end{cases}$$

Since all these formulations are formally equivalent, picking one of them is a matter of convenience for the interpretation. In the program the code is formulated according to the first formulation but parameters are also presented in the second formulation. The third formulation may be used, after the estimation step, for specific interpretations. Notice that the one-to-one correspondence among the formulations guarantees that each of them gives asymptotically efficient ML estimates for the parameters.

The code to build a CUB model with a *shelter effect* at the `shelter` category is:

```
> m=number_of_ordinal_categories
> CUB(ordinal,shelter)
```

The CUB model with covariates for the *shelter effect* has been named a G*e*CUB model (=*Ge*neralized CUB model) in Iannario and Piccolo (2012b, 2015a) where it is defined, according to the second formulation (10), as:

$$Pr\,(R=r) = \delta_i\left[D_r^{(c)}\right] + (1-\delta_i)\left[\pi_i\, b_r(\xi_i) + (1-\pi_i)\,\frac{1}{m}\right], \quad r=1,2,\ldots,m. \tag{12}$$

Given the knowledge of the matrices $\boldsymbol{Y}$, $\boldsymbol{W}$ and $\boldsymbol{W}$ containing the information of the subjects, the links are established as:

$$logit\,(\pi_i) = \boldsymbol{y}_i\boldsymbol{\beta}\,; \quad logit\,(\xi_i) = \boldsymbol{w}_i\boldsymbol{\gamma}\,; \quad logit\,(\delta_i) = \boldsymbol{x}_i\boldsymbol{\omega}\,; \quad i=1,2,\ldots,n. \tag{13}$$

A full estimation of a G*e*CUB model with covariates is currently available in a program written in $GAUSS^{©}$ language and available from Authors upon request.

A very special G*e*CUB model in case $\pi \equiv 0\ \ \forall i$ has been designed by Capecchi and Piccolo (2015) as a CUSH model (a *C*ombination of a discrete *U*niform random variable with a *SH*elter effect). It is specified by the following probability mass function:

$$Pr(R=r) = \delta\, D_r^{(c)} + (1-\delta)\,\frac{1}{m}\,, \qquad r=1,2,\ldots,m\,, \tag{14}$$

for any $\delta \in [0,1]$. In fact, a CUSH model is just a CUB model with a *shelter effect* and such that $\pi_1 = 0$ or $\pi^* = 0$ or $\lambda = 0$ according to the three previous formulations, respectively; however, in real situations, this model deserves specific analysis and allows useful consideration.

When covariates for the *shelter effect* are significant, a CUSH model with covariates is defined by:

$$Pr(R=r \mid \boldsymbol{x}_i) = \delta_i\, D_r^{(c)} + (1-\delta_i)\,\frac{1}{m}\,; \qquad logit(\delta_i) = \boldsymbol{x}_i\boldsymbol{\omega}\,; \qquad i=1,\ldots,n\,, \tag{15}$$

where information on the subjects' covariates are collected in the matrix $\boldsymbol{X}$.

The main commands are the following:

```
> m=number_of_ordinal_categories
> CUSH(ordinal,shelter)   # CUSH model without covariates
> CUSH(ordinal,shelter,X) # CUSH model with covariates in X
```

The CUSH function has the default option `makeplot=TRUE` (which may be changed to `makeplot=FALSE` if plots have to be dropped).

7 IHG models

In a series of papers, mainly motivated by the marginal analysis of observed ranks, D'Elia (1999, 2001, 2003) considered the Inverse (or Negative) Hypergeometric distribution (IHG) as an useful data generating process for ordinal data in the special cases where the modal preference is located at one of the extreme value of the support. For a given m, this random variable is characterized by a single parameter θ which is a measure of preference, attraction, pleasantness, etc. towards the item. If $\theta = 1/m$, the IHG random variable reduces to the discrete Uniform distribution; as a consequence, to save identifiability, an uncertainty component is never added to this class of models.

Although a different approach has led to IHG models, this family of random variables are considered in this program since IHG random variables may be considered as a specific case of CUBE models.

The *stochastic* and *systematic* components of the IHG model are:

$$\begin{cases} Pr\left(Y_i = j \mid \boldsymbol{\gamma}\right) = \theta_i(1-\theta_i)^{j-1}\,\dfrac{m-1}{m}\,\displaystyle\prod_{s=1}^{j}\dfrac{m-s+1}{m-s+\theta_i(s-1)}\,; \\ logit(\theta_i) = \boldsymbol{u}_i^{(\gamma)}\,\boldsymbol{\gamma}\,; \end{cases}$$

for $j = 1, \dots, m$ and $i = 1, \dots, n$. Here, $\boldsymbol{u}_i^{(\gamma)}$ is the i-th row of the matrix $\boldsymbol{U}$ which includes useful covariates for explaining the responses.

The following commands should be run to estimate IHG models, without and with covariates:

```
> m=number_of_categories
> IHG(ordinal)         # IHG model without covariates
> IHG(ordinal,U)       # IHG model with covariates in U
```

In addition, the command

```
> plotloglikihg(frequencies)
```

plots the log-likelihood function of an IHG model over the whole support of θ for a sample whose absolute frequency distribution is summarized in the `frequencies` of m elements.

Notice that the interpretation of the parameter θ is related to m; thus, some caution should be exerted when comparing the preference parameters estimated from surveys with a different number of categories (Iannario and Piccolo, 2015b).

8 Inferential issues

A full usage of the program implies some knowledge of the selected estimation routines and the several measures computed to test and validate the estimated models. In addition, the availability of many functions allow researchers to set up further tools for their specific needs.

First of all, the ML method is performed thanks to the EM procedure (Piccolo, 2006) which requires accurate initial values to reach convergence in acceptable time. This goal has been effectively and automatically obtained by means of specific functions: `inibest`, `inigrid`, `inibestgama`, `inibestcube` and `iniihg`.

For CUB models and IHG without covariates default are `inibest` and `iniihg`, respectively; for CUB models with covariates for *feeling* the program always implements `inibestgama` whereas for CUBE models (without covariates) `inibestcube` is performed.

If these initial estimates are required, they can be obtained as follows:

```
> inibest(freq)          # automatically computed for CUB models


> iniihg(freq)           # automatically computed for IHG models


> inigrid(freq,x,y)      # given as a reference for CUB models
                         # for selected x and y values
                         # for pai and csi, respectively


> inibestgama(ordinal,W) # automatically computed for CUB models
                         # with covariates W for feeling
```

```
> inibestcube(ordinal)   # automatically computed for CUBE models


> inibestcubecov(ordinal,Y,W,Z)
                             # automatically computed for CUBE models
                             # with covariates Y (uncertainty),
                             # W (feeling), Z (overdispersion)
```

Observe that the functions `inibest`, `iniihg` and `inigrid` are applied to a vector of absolute frequencies (`freq`) whereas the functions `inibestgama`, `inibestcube` and `inibestcubecov` require the vector of sample data (`ordinal`).

Secondly, for each parameter of CUB and CUBE models, the output of the program shows parameter estimates, asymptotic standard errors, Wald-tests and p-values. The variance-covariance and the correlation matrices of estimates are also presented. In this respect, the common caveats apply when testing borderline hypotheses (Molenberghs and Verbeke, 2007); on the contrary, for Wald tests of the parameters of covariates the asymptotic theory may be safely applied.

Third, a list of likelihood-based measures and several (general and specific) fitting indexes are printed: dissimilarity measures, Pearson X^2 and deviance. In addition, to compare non-nested models, also AIC (Akaike, 1974), BIC (Schwarz, 1978) and ICOMP (Bodzogan, 1990) measures are computed.

Then, for CUB and CUBE models without covariates, the program presents a table where -for each category- observed relative frequencies, estimated probabilities, Pearson and relative residuals are listed.

Finally, for CUB models with 1 or 2 discrete covariates for the parameter of feeling (ξ) it is possible to obtain the X^2 index of Pearson according to the command:

```
> chi2cub(m,ordinal,W,pai,gama)
```

Possible modifications when the covariates are relevant for the parameter of uncertainty (ξ) are immediate.

9 Plotting facilities

An added value of CUB models is the easiness of interpretation when estimated models are plotted as points in the parameter space, that is the unit square.

In addition to the standard graphical output of CUB, CUBE, CUSH and IHG models, a simple graphical device is based on the function `cubvisual`: it shows a CUB model for the data vector `ordinal` as a single point in the parameter space with some useful options. If necessary, other estimated models (=points) may be added with the standard commands of the ***R*** environment (as `points`(.) and `lines`(.), for instance). Thus, the code is the following:

```
> m=number-of-categories
> cubvisual(ordinal)                        # minimal information


> cubvisual(ordinal,caption,labordinal, maxiter,toler,xlim,ylim)
                                            # complete options
```

A general opportunity is offered by the function `multicub` which allows to plot, with several options, many estimated CUB models over the same unit square if the ordinal responses to different items are included in a number of columns (greater than 1) of a matrix `matord`.

The minimal and complete commands of `multicub` are:

```
> multicub(matord,m)                      # minimal information


> multicub(matord,m,labordinal,caption,colo,symb,
            thickness,xwidth,ywidth)
                                          # complete options
```

Thus, for each point (=estimated CUB model), we may specify a label (with `labordinal`), a title (with `caption`), colours (with `colo`), symbols (with `symb`), the `thickness` of the points and the size of the plot with `xwidth` and `ywidth` (the unit square is the default).

Finally, a visual tool for CUB models with covariates is the *Scatter of Parameter Estimates* (= *SPE*) plot which consists in drawing the estimated $(\hat{\pi}_i,\, \hat{\xi}_i)$, for $i = 1, 2, \ldots, n$, over the unit square. This scatter plot is able to detect peculiar behaviour in subsets of respondents, as we will show in Section 11 with a real case study.

From a general point of view, the results of the estimation of a CUB model (and variants) can be usefully exploited by alternative graphical devices.

- In the parameter space, each estimated model (without covariates) is a single point where *uncertainty* and *feeling* are immediately recognized. If covariates assume discrete categories, a sequence of points is plotted; in the case of a continuous covariate it is possible to plot a parametric curve showing how *uncertainty* and *feeling* change with the selected covariate. When several covariates are involved in the estimated model, some conditional plots are necessary to emphasize the effect of each of them.
- To show the effect of a covariate on the *uncertainty* or *feeling*, it is convenient to plot the logistic link as a function of the selected covariate, respectively.
- To derive the expected profile of the respondents, it is wise to plot the whole probability mass function of the resulting CUB model after conditioning on selected values of the covariates.

10 Simulation routines

If pseudo-random numbers have to be generated within the family of CUB models, the following functions are available.

```
> simcub(n,m,pai,csi)                    # generate n observations from
                                         # a CUB model with parameters
                                         # (pai, csi), for a given m


> simcubshe(n,m,pai,csi,delta,shelter)# generate n observations
                                         # from a CUB model
                                         # with a shelter effect
                                         # and with parameters (pai, csi)


> simcube(n,m,pai,csi,phi)               # generate n observations from
```

```
                                      # a CUBE model with
                                      # parameters (pai, csi, phi)

> simcush(n,m,delta,shelter)          # generate n observations from
                                      # a CUSH model with
                                      # parameters (delta, shelter)

> simihg(n,theta)                     # generate n observations from
                                      # an IHG model with
                                      # parameter (theta)
```

In order to simulate observations from a model with covariates, one should first obtain the parameters (π_i, ξ_i, ...) corresponding to the chosen profiles for the subjects and then generate the sample data by using the previous simulation routines with the given parameters.

When performing simulation experiments on the behaviour of estimates and/or fitting indexes, it is useful to apply routines which are not so elaborate with respect to the presentation of results. Thus, the program includes simplified code of the estimation and testing of CUB, CUBE and CUSH functions, respectively, which have been finalized to perform long run of simulation experiments.

To activate these commands, use the following codes:

```
> cub00forsim(ordinal,maxiter,toler)          # CUB model

> cubeforsim(ordinal,starting,maxiter,toler)  # CUBE model

> cushforsim(ordinal,shelter)                 # CUSH model
```

For a standard CUB model we may modify both `maxiter` (the maximum number of iterations allowed) and `toler` (the criterion for convergence based on the increment of log-likelihood functions). In addition, for CUBE models we may also modify `starting` (the sequence of initial values for the estimates of parameters).

11 Some examples

The previous commands are exemplified in some real and artificial situations with a main emphasis on the graphical outputs.

First of all, given $m = 9$, we generate $n = 500$ ordinal observations from a CUB model with $\pi = 0.7$ and $\xi = 0.2$. Then, we estimate the parameters $(\hat{\pi}, \hat{\xi})$ and plot them in the parameter space with a 95% confidence ellipse. The parameters correlation is also computed and displayed: this is possible since `CUB` function calls for `cub00` which assigns `varmat` to the variance-covariance of the estimates.

In Figure 1, we present the output of CUB model estimation run (upper panel) and the visualization of the estimated CUB model in the parameter space (bottom panel). The code (with detailed comments) is the following.

```
> source("CUB.R")
> m=9; n=500
> pai=0.7; csi=0.2
> ordinal=simcub(n,m,pai,csi)
> ### Split of the screen in two panels
> par(mfrow=c(2,1))
> par(mar=c(5,4,3,2)+0.1)
> ### First plot
>    CUB(ordinal)
> ### Second plot
>    plot(1-pai,1-csi,main="CUB model for ordinal",cex=1.2,cex.main=1,
               las=1,pch=19, xlim=c(0.2,0.4),ylim=c(0.7,0.9),
               font.lab=4,cex.lab=1,
               xlab=expression(paste("Uncertainty  ", (1-pi))),
               ylab=expression(paste("Feeling  ", (1-xi))))
> ### Compute parameter correlations
>   corrpar=varmat[1,2]/sqrt(varmat[1,1]*varmat[2,2])
>   labelcorr=paste("Parameters correlation =",round(corrpar,3))
```

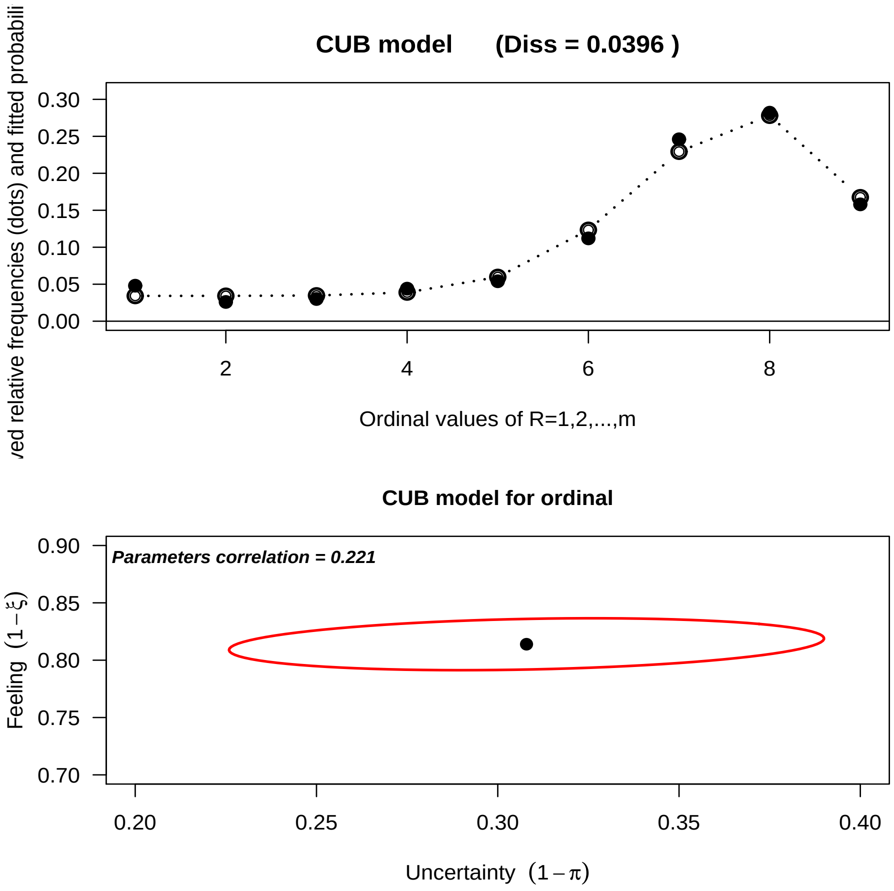

Figure 1: Observed and estimated distribution (upper) and visualization of the CUB model in the parameter space (bottom)

```
>   text(0.23,0.89,labels=labelcorr,font=4,cex=0.8)
> ### Draw ellipse
>    library(ellipse)
>    plot(1-pai,1-csi,main="CUB model for ordinal",
         cex=1.2,cex.main=1, font.lab=4,cex.lab=1,
         pch=19, xlim=c(0.15,0.30),ylim=c(0.75,0.85),
         xlab=expression(paste("Uncertainty  ", (1-pi))),
         ylab=expression(paste("Feeling  ", (1-xi))))
>    lines(ellipse(varmat,centre=c(1-pai,1-csi)),lwd=2,col="red")
> par(mar=c(5,4,4,2)+0.1)
> par(mfrow=c(1,1))
```

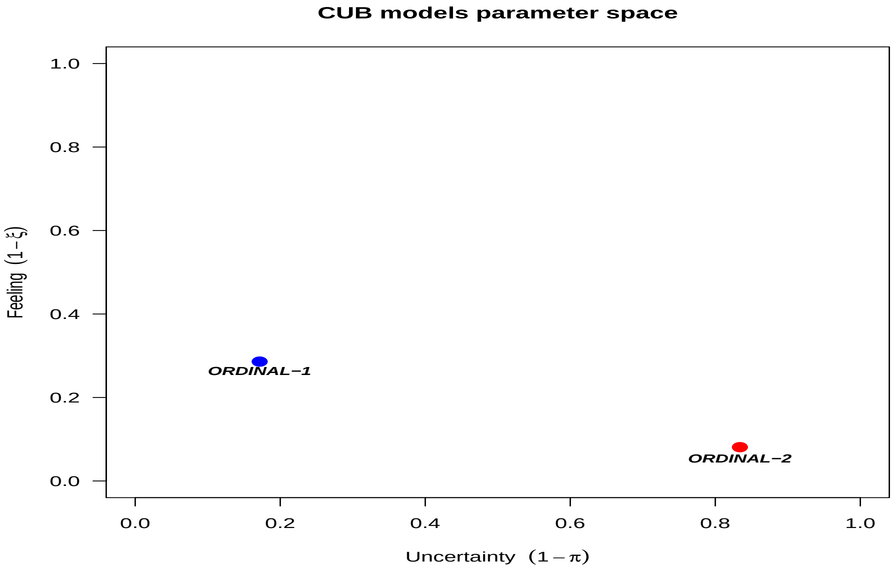

Figure 2: A visual representation of two estimated CUB models in the parameter space

As a second example, we use the command **cubvisual** which plots an estimated CUB model (without covariates) for a single vector **ordinal1** as a point in the parameter space. Then, on the same plot, we add the representation of a further CUB model estimated on a different vector ordinal2 with the ***R*** command **points**(...). Code commands follow and results are shown in Figure 2.

```
> source("CUB.R")
> m=7
> ### First model for ordinal1 ("blue")
> cubvisual(ordinal1,"ORDINAL-1")
> ### Second model for ordinal2 ("red")
> cub00(ordinal2,makeplot=FALSE)
> points(1-pai,1-csi,pch=19,cex=1.5,col="red")
> text(1-pai,1-csi,labels="ORDINAL-2",font=4,pos=1,
offset=0.5,cex=0.8)
```

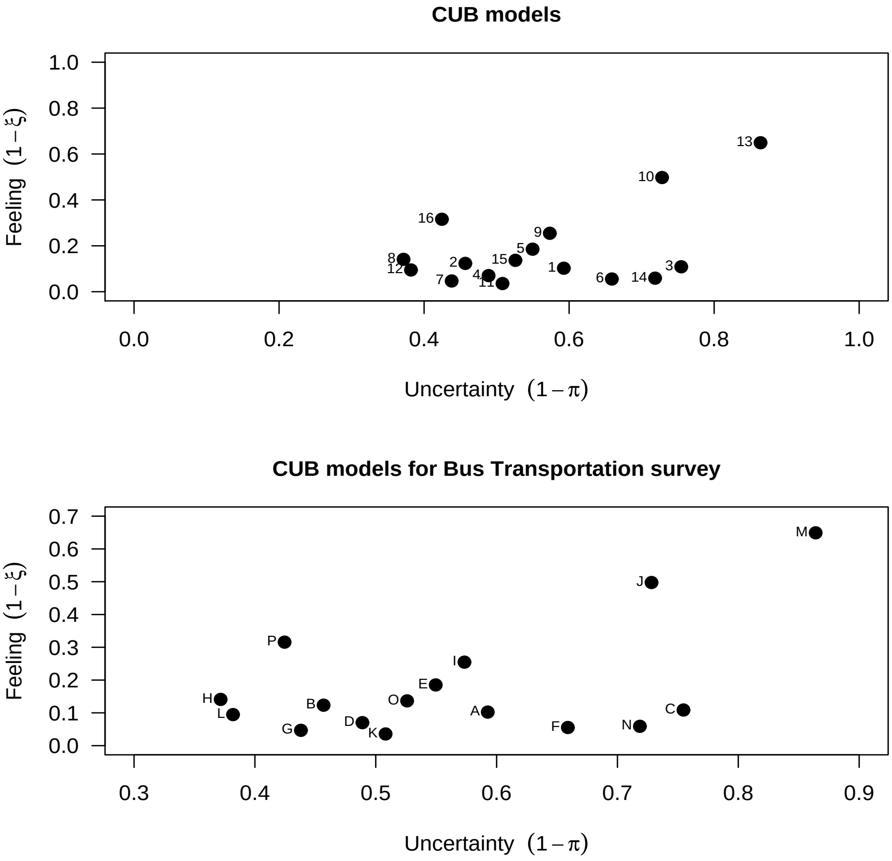

Figure 3: Usage of `multicub` command: standard version (upper) and more elaborated version (bottom)

As a third example, we use the `multicub` command on a real data set concerning several evaluations expressed on a Likert scale with $m = 10$ by users of a public bus transport to/from a metropolitan area. We assume that all ordinal data have been loaded in a matrix `dati`, consisting of $n = 105$ ratings on $nk = 16$ items.

The first command of the following code is the standard version (upper panel of Figure 3) whereas the second one is a more elaborated version of the same command (bottom panel of Figure 3).

```
> source("CUB.R")
> dati=read.table("C:/.../Transports.R",header=TRUE)
> m=10
> par(mfrow=c(2,1))
> par(mar=c(5,4,3,2)+0.1)
### First plot (upper)
> multicub(dati,m)
### Second plot (bottom)
> multicub(dati,m,etich=LETTERS[1:ncol(dati)],
          titolo="CUB models for Bus Transportation survey",
          colori="black",simboli=19,thickness=1.5,
          xwidth=c(0.3,0.9),ywidth=c(0,0.7))
> par(mar=c(5,4,4,2)+0.1)
> par(mfrow=c(1,1))
```

As a fourth example, we consider a large data set called `nes96` –discussed by Faraway (2006, pp.106-112) in the context of a multinomial logit model, among others– which consists in the evaluation of 944 respondents with respect to the political Left-Right orientation of Bill Clinton (=`ClinLR`). This evaluation is examined as a function of party identification (=`PID`), age in years (=`Age`) and education level (=`Educ`) of the respondents. The ordinal variables `ClinLR` and `PID` are expressed on a Likert scale ranging from *Surely Left* = 1 up to *Surely Right* = 7. We report the commands for the best CUB model obtained to explain the responses `ClinLR`.

Then, we plot the *SPE* diagram for all respondents, by introducing a new variable to re-define `PID` with the following simplified recoding scheme:

$$RePID = \begin{cases} -1 & \text{if} \quad PID = 1, 2\,; \\ \ \ 0 & \text{if} \quad PID = 3, 4, 5\,; \\ \ \ 1 & \text{if} \quad PID = 6, 7\,. \end{cases}$$

In this way, `RePID` is a rough classification of the political orientation of the respondent as Democrat (`RePID`$= -1$), Intermediate (`RePID`$= 0$) and Republican (`RePID`$= +1$), respectively.

```
### Read data and define variables
```

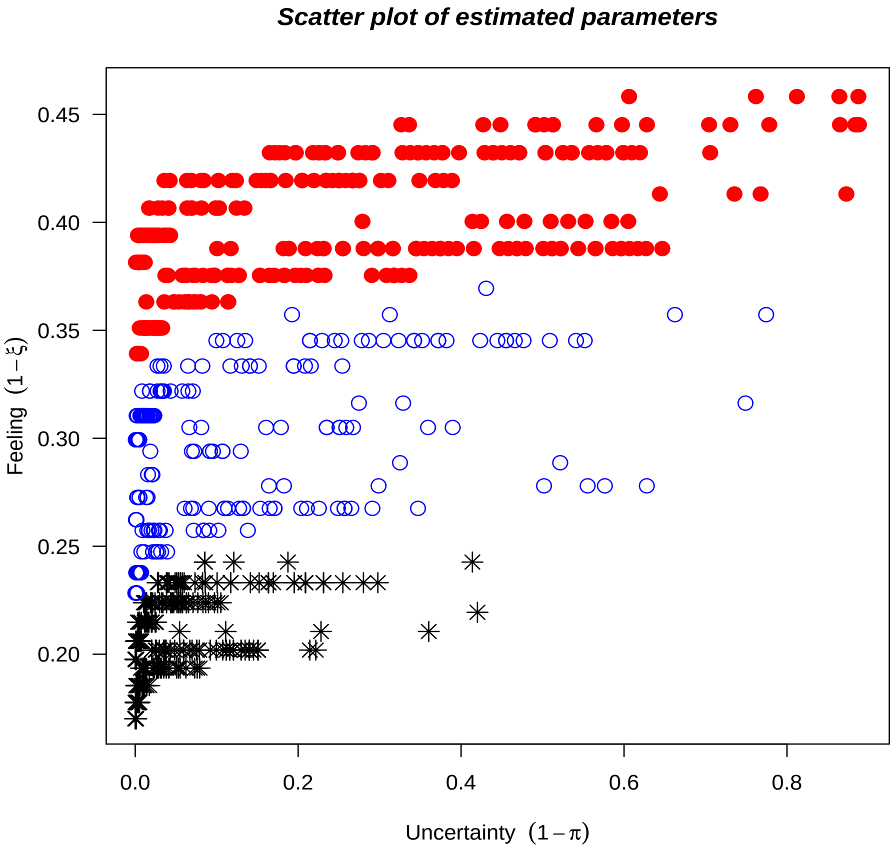

Figure 4: A typical Scatter Plot of Estimates (*SPE*)

```
> dati=read.table("C:/.../nes_96.txt",header=T)
> ClinLR=dati$ClinLR                                  # 1...7
> PID=dati$PID                                        # 1...7
> Age=dati$age                                        # 19...91
> Educ=dati$educ                                      # 1...7
> n=length(ClinLR)                                    # n=944
### Estimate CUB model with covariates
> source("CUB.R")
> m=7
> Y=cbind(PID,Age,Educ); W=cbind(PID,Educ)
> CUB(ClinLR,Y,W)
### Numerical estimates of the parameters
#####################
### bet=c(-5.985787, 0.355111, 0.043023, 1.284132)
### gama=c(-0.068924, 0.183626, 0.052654)
#####################
### Recode PID by means of RePID
> RePID=rep(NA,n)
> RePID[PID==1 | PID==2]=-1
> RePID[PID==3 | PID==4 | PID==5]=0
> RePID[PID==6 | PID==7]=1
##################
### make the SPE plot
> paivet=logis(Y,bet)
> csivet=logis(W,gama)
### Figure 3
> caption="Scatter plot of estimated parameters"
> symb=rep(19,n); symb[repid==0]=1; symb[repid==1]=8;
> colo=rep("red",n); colo[repid==0]="blue";
      colo[repid==1]="black"
> plot(1-paivet,1-csivet,xlim=c(0,0.89),ylim=c(0.17,0.46),
      cex=1.5,pch=symb,col=colo,
      xlab=expression(1-pi),ylab=expression(1-xi),
      main=caption,font.main=4)
```

With adequate codes for symbols and colours of the points, Figure 4 clearly visualizes a different behaviour of the respondents as function of `RePID` and make easier the interpretation of the results. The estimated points $(\hat{\pi}_i,\, \hat{\xi}_i)$, for $i = 1, 2, \ldots, n$ are easily obtained as functions of the values of the subjects' covariates via a logit link which is computed with the function `logis()` included in the main program. Thus, to obtain the *SPE* plot new functions are not required.

12 Empirical evidence for CUB models

Ordinal data arise in several applied and scientific fields; thus, the applications of related models are pervasive in the statistical literature. With regard to the application of CUB models and their variants we list the topics where they have been successfully applied to the best of our knowledge (ranking* and rating analysis are reported). Most of the data sets here quoted are available at www.labstat.it website.

- *Preferences*:
  Colours* (young people, children, air force cadets). Cities where to live*. Professions for students of Political Sciences graduates*. Olive oils preference. Coffee preference. Sensometric analysis and consumers' behaviours. Typical agri-products of South of Italy. Italian newspapers*. Political affairs: Left/Right self-placement.

- *Evaluations*:
  Orientation services at University. University teaching and structures. Services for E-bay users. Importance and performance of products. Repeatability and reproducibility in Measurement System Analysis. Characteristics of bus transports towards a metropolitan area*. Degree of preference for buying equo-solidal agricultural products. Quality of services in a protected area. Customers' satisfaction of European consumers towards salmon. Judgment of a city administration. Final degree of University graduates. Questionnaire validation for patient satisfaction.

- *Perceptions*:
  Urban audit surveys about city emergencies*. Perceived risk in a printing factory. Chronic pain threshold in TMD. Synonyms and semantic space of words*. Ethnical identity of immigrants by cohorts*. European Union objectives and policies*. Perception of Economic Security and Job satisfaction in SHIW surveys. Measure of Happiness. Job satisfaction of Italian graduates. Work related problems in Eurofound surveys. Subjective survival probability to 75 and 90 years. Importance-Performance analysis in marketing research. Coffee tasting. Consumer perception of wine attributes. Level of teachers' stress. Intention to Human Papilloma Virus vaccination. Intention to seasonal influenza vaccination. Conflict with job environment.

13 Bibliographic notes

The CUB model framework started with Piccolo (2003) and it has been mainly exploited for rank data by D'Elia and Piccolo (2005). Inferential issues for estimation and testing purposes have been established in Piccolo (2006). The identifiability of CUB models has been proved by Iannario (2010).

The modern approach is presented in Iannario (2012a,b) and Iannario and Piccolo (2012a). Preliminary estimates have been repeatedly tested (Iannario, 2008, 2009b, 2012c) and specific fitting measures for ordinal data models have been proposed (Iannario, 2009a).

The extension of CUB models with subjects' covariates has been obtained by Piccolo (2006); Iannario and Piccolo (2010). The analysis with both subjects' and objects' covariates has been firstly performed in Piccolo and D'Elia (2008); further examples are in Capecchi and Endrizzi (2015).

The consideration of a *shelter effect* has been studied by Iannario (2012a) and successfully applied in Corduas, Iannario and Piccolo (2009); Iannario and Piccolo (2014). The introduction of covariates in a CUB model with a *shelter effect* has been introduced by Iannario and Piccolo (2012b) with the definition of G*e*CUB models: for these models a program written in $GAUSS^{©}$ language is currently available.

CUBE models have been proposed and estimated by Iannario (2012b, 2014a,b) whereas the development of CUBE models with covariates is due to Piccolo (2015).

Specialized extensions of the CUB family of statistical models are:

- Hierarchical CUB models (HCUB): Iannario (2012d)
- CUB models in case of complex designs: Gambacorta, Iannario and Vaillant (2014)
- Latent class of CUB models (LC-CUB): Grilli, Iannario, Piccolo and Rampichini (2013)
- CUB models with varying uncertainty (VCUB): Gottard, Iannario and Piccolo (2015)
- Nonlinear CUB models (NL-CUB): Manisera and Zuccolotto (2013, 2014b)
- Generalized mixture models with uncertainty: Iannario and Piccolo (2015b)
- CUB models with "don't know" option: Manisera and Zuccolotto (2014a)

These bibliographic notes are not exhaustive since they have been limited to the main methodological papers which originated the framework of CUB models and their extensions. More specific contributions have been added by several researchers on specific topics, *e.g.* clustering ordinal data (Corduas, 2011a; Deldossi and Zappa, 2014), missing values (Cugnata and Salini, 2014), measurement errors (Deldossi and Zappa, 2011), importance-performance analysis (Cugnata and Salini, 2013; Cugnata, Guglielmetti and Salini, 2014), relationships with multivariate analysis (Iannario and Maravalle, 2011). In addition, alternative inferential approaches have been pursued, as permutation test (Bonnini, Piccolo, Salmaso and Solmi, 2012) and Bayesian analysis (Deldossi and Paroli, 2014, 2015). Further analyses concerning multivariate CUB models are in progress (Corduas, 2011b, 2015; Andreis and Ferrari, 2013; Colombi and Giordano, 2015).

According to the paradigm of CUB models, a recent extension of cumulative models with an uncertainty component has been suggested by Tutz, Schneider, Iannario and Piccolo (2014). Finally, a unified approach which includes both the family of CUB models and the cumulative ones has been proposed Iannario and Piccolo (2015b) whereas a critical comparison among CUB and proportional odd models has been advanced by Iannario and Piccolo (2015c).

Acknowledgements. The version 4.0 of the CUB program in **R** has been thoroughly tested and substantially improved by Dr. Rosaria Simone at University of Naples Federico II. This work has been supported by FIRB2012 project at University of Perugia (code RBFR12SHVV) and by the SHAPE project within the frame of Programme STAR (CUP E68C13000020003) at University of Naples Federico II, financially supported by UniNA and Compagnia di San Paolo.

References

Agresti A. (2010). *Analysis of Ordinal Categorical Data*, 2^{nd} edition. J.Wiley & Sons, Hoboken.

Akaike H. (1974). A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, **AC-19**, 716–723.

Anderson J.A. (1984). Regression and ordered categorical variables. *Journal of the Royal Statistical Society, Series B*, **46**, 1–30.

Andreis F., Ferrari P.A. (2013). On a copula model with CUB margins. *Quaderni di Statistica*, **15**, 33–51.

Bodzogan H. (1990). On the information-based measure of covariance complexity and its application to the evaluation of multivariate linear models. *Communications in Statistics. Theory and Methods*, **19**, 221–278.

Bonnini S., Piccolo D., Salmaso L., Solmi F. (2012). Permutation Inference for a Class of Mixture Models. *Communications in Statistics. Theory and Methods*, **41**, 2879–2895.

Capecchi S., Endrizzi I. (2015). A multi-product approach for detecting subjects' and objects' covariates in consumer preferences, *Preliminary Report*.

Capecchi S., Piccolo D. (2015). Dealing with large heterogeneity/uncertainty in sample survey with ordinal data. *IFCS Proceedings*, University of Bologna.

Colombi R., Giordano S. (2015). *Accounting for two types of uncertainty in a multidimensional* CUB *model*, paper presented at the 'Conference on Statistical Models for Complex Data', University of Naples Federico II, 26 June.

Corduas M. (2011a). Assessing similarity of rating distributions by Kullback-Leibler divergence. In: Fichet, B. *et al.* (eds.) *Classification and Multivariate Analysis for Complex Data Structures* (pp.221–228). Physica-Verlag, Springer, Berlin Heidelberg.

Corduas M. (2011b). Modelling correlated bivariate ordinal data with CUB marginals. *Quaderni di Statistica*, *13*, 109–119.

Corduas M. (2015). Analyzing bivariate ordinal data with CUB margins. *Statistical Modelling: an International Journal*, **15**, DOI:10.1177/1471082X14558770

Corduas M., Iannario M., Piccolo D. (2009). A class of statistical models for evaluating services and performances, in: M.Bini *et al.* (eds.): *Statistical methods for the evaluation of educational services and quality of products*, Contribution to Statistics (pp.99–117). Berlin Heidelberg: Physica-Verlag, Springer.

Cugnata F., Guglielmetti C., Salini S. (2014). Model-based approach to validate a measuring instrument for assessing a quality of care. *QdS - Journal of Methodological and Applied Statistics*, **15**, 203–217.

Cugnata F., Salini S. (2013). Model-based approach for importance-performance analysis. *Quality & Quantity*, **48**, 3053–3064.

Cugnata F., Salini S. (2014). Comparison of alternative imputation methods for ordinal data. *Communications in Statistics. Simulation and Computation*, DOI: 10.1080/03610918.2014.963611

Deldossi L., Paroli R. (2014). Bayesian covariate selection in CUB model: some considerations. *Quaderni di Statistica*, **14**, 85–88.

Deldossi L., Paroli R. (2015). Bayesian variable selection in a class of mixture models for ordinal data: a comparative study. *Journal of Statistical Computation and Simulation*, **85**, 1926–1944.

Deldossi L., Zappa D. (2011). Measurement errors and uncertainty: a statistical perspective, in: Ingrassia S., Rocci S., Vichi M. (eds.), *New Perspectives in Statistical Modeling and Data Analysis* (pp.145–153). Springer, Berlin.

Deldossi L., Zappa D. (2014). Evaluating R&R of ordinal classifications with CUB model. *Quaderni di Statistica*, **14**, 89–92.

D'Elia A. (1999). A Proposal for Ranks Statistical Modelling. In: Friedl, H. *et al.* (eds.) *Statistical Modelling - Proceedings of the 14th International Workshop on Statistical Modelling* (pp. 468–471). Graz, Austria.

D'Elia A. (2000). Il meccanismo dei confronti appaiati nella modellistica per graduatorie: sviluppi statistici ed aspetti critici. *Quaderni di Statistica*, **2**, 173–203.

D'Elia A. (2001). A comparison between two asymptotic tests for analysing preferences. *Quaderni di Statistica*, **3**, 127–143.

D'Elia A. (2003). Modelling ranks using the Inverse Hypergeometric distribution. *Statistical Modelling: an International Journal*, **3**, 65–78.

D'Elia A., Piccolo D. (2005). A mixture model for preference data analysis. *Computational Statistics & Data Analysis*, **49**, 917–934.

Faraway J.J. (2006). *Extending the linear model with R.* Chapman & Hall/CRC, Boca Raton, FL.

Gambacorta R., Iannario M., Vaillant R. (2014). Design-based inference in a mixture model for ordinal variables for a two stage stratified design. *Australian and New Zealand Journal of Statistics*, **56**, 125–143.

Gottard A., Iannario M., Piccolo D. (2015). Varying uncertainty in CUB models. *Submitted.*

Grilli L., Iannario M., Piccolo D., Rampichini C. (2014). Latent Class CUB Models. *Advances in Data Analysis and Classification*, **8**, 105–119.

Honaker J., King G., Blackwell M. (2011). Amelia II: A Program for Missing Data. *Journal of Statistical Software*, **45**, 1–47.

Iannario M. (2008). Selecting feeling covariates in rating surveys. *Rivista di Statistica Applicata*, **20**, 103–116.

Iannario M. (2009a). Fitting measures for ordinal data models. *Quaderni di Statistica*, **11**, 39–72.

Iannario M. (2009b). A comparison of preliminary estimators in a class of ordinal data models. *Statistica & Applicazioni*, **VII**, 25–44.

Iannario M. (2010). On the identifiability of a mixture model for ordinal data. *Metron*, **LXVIII**, 87–94.

Iannario M. (2012a). Modelling *shelter* choices in a class of mixture models for ordinal responses. *Statistical Methods and Applications*, **21**, 1–22.

Iannario M. (2012b). CUBE models for interpreting ordered categorical data with overdispersion. *Quaderni di Statistica*, **14**, 137–140.

Iannario M. (2012c). Preliminary estimators for a mixture model of ordinal data. *Advances in Data Analysis and Classification*, **6**, 163–184.

Iannario M. (2012d). Hierarchical CUB models for ordinal variables. *Communications in Statistics. Theory and Methods*, **41**, 3110–3125.

Iannario M. (2014a). Modelling Uncertainty and Overdispersion in Ordinal Data. *Communications in Statistics. Theory and Methods*, **43**, 771–786.

Iannario M. (2014b). Detecting latent components in ordinal data with overdispersion by means of a mixture distribution. *Quality & Quantity*, **49**, 977–987.

Iannario M., Maravalle M. (2011). Parametric and nonparametric approaches to the semantic relationship among synonymy. *Quaderni di Statistica*, **11**, 83–108.

Iannario M., Piccolo D. (2010). A New Statistical Model for the Analysis of Customer Satisfaction. *Quality Technology & Quantitative Management*, **7**, 149–168.

Iannario M., Piccolo D. (2012a). CUB models: Statistical methods and empirical evidence, in: Kenett R.S. and Salini S. (eds.). *Modern Analysis of Customer Surveys: with applications using R* (pp. 231–258). J. Wiley & Sons, Chichester.

Iannario M., Piccolo D. (2012b). A Framework for Modelling Ordinal Data in Rating Surveys. *Proceedings of the Joint Statistical Meetings*, American Statistical Association, Section on Statistics in Marketing, pp.3308-3322. San Diego, California.

Iannario M., Piccolo D. (2014). A new paradigm for modelling ordinal responses in sample surveys. Paper presented at the *Conference of European Statistics Stakeholders*, Sapienza University of Rome.

Iannario M., Piccolo D. (2015a). A generalized framework for modelling ordinal data. *Statistical Methods and Applications*, forthcoming.

Iannario M., Piccolo D. (2015b). A comprehensive framework for regression models of ordinal Data. *Submitted.*

Iannario M., Piccolo D. (2015c). A comparative analysis of alternative frameworks for modelling ordinal data. *Preliminary report.*

Manisera M., Zuccolotto P. (2013). Nonlinear CUB models: some stylized facts. *Quaderni di Statistica*, **15**, 111-130.

Manisera M., Zuccolotto P. (2014a). Modeling “Don’t know” responses in rating scales. *Pattern Recognition Letters*, **45**, 226–234.

Manisera M., Zuccolotto P. (2014b). Modeling rating data with Nonlinear CUB models. *Computational Statistics and Data Analysis*, **78**, 100–118.

Manisera M., Zuccolotto P. (2015). Identifiability of a model for discrete frequency distributions with a multidimensional parameter space. *Journal of Multivariate Analysis*, forthcoming.

Molenberghs G., Verbeke G. (2007). Likelihood ratio, score, and Wald tests in a constrained parameter space. *The American Statistician*, **61**, 22–27.

Piccolo D. (2003). On the moments of a mixture of uniform and shifted binomial random variables. *Quaderni di Statistica*, **5**, 85–104.

Piccolo D. (2006). Observed information matrix for MUB models. *Quaderni di Statistica*, **8**, 33–78.

Piccolo D. (2015). Inferential issues for CUBE models with covariates. *Communications in Statistics. Theory and Methods*, **44**, DOI: 10.1080/03610926.2013.821487

Piccolo D., D'Elia A. (2008). A new approach for modelling consumers' preferences. *Food Quality and Preference*, **19**, 247–259.

Self S.G., Liang K.Y. (2003). Asymptotic properties of maximum likelihood estimators and likelihood ratio test under nonstandard conditions. *Journal of the American Statistical Association*, **82**, 605–610.

Schwarz G. (1978). Estimating the Dimension of a Model. *The Annals of Statistics*, **6**, 461–464.

Tutz G. (2012). *Regression for Categorical Data.* Cambridge University Press, Cambridge.

Tutz G., Schneider M., Iannario M., Piccolo D. (2014). *Mixture Models for Ordinal Responses to Account for Uncertainty of Choice.* Technical Report Number 175, Department of Statistics, University of Munich.

Vu H.T.V., Zhou S. (1997). Generalization of likelihood ratio tests under nonstandard conditions. *The Annals of Statistics*, **25**, 897–916.