Alma Mater Studiorum - Università di Bologna

DOTTORATO DI RICERCA IN

SCIENZE STATISTICHE

Ciclo 35

Settore Concorsuale: 13/D1 - STATISTICA

Settore Scientifico Disciplinare: SECS-S/01 - STATISTICA

SEEMINGLY UNRELATED LINEAR REGRESSION FOR CONTAMINATED DATA BASED ON GAUSSIAN MIXTURES

Presentata da: Gabriele Perrone

Coordinatore Dottorato

Monica Chiogna

Supervisore

Gabriele Soffritti

Esame finale anno 2023

Abstract

In this thesis, new classes of models for multivariate linear regression defined by finite mixtures of seemingly unrelated contaminated normal regression models and seemingly unrelated contaminated normal cluster-weighted models are illustrated. The main difference between such families is that the covariates are treated as fixed in the former class of models and as random in the latter. Thus, in cluster-weighted models the assignment of the data points to the unknown groups of observations depends also by the covariates. These classes provide an extension to mixture-based regression analysis for modelling multivariate and correlated responses in the presence of mild outliers that allows to specify a different vector of regressors for the prediction of each response. Expectation-conditional maximisation algorithms for the calculation of the maximum likelihood estimate of the model parameters have been derived. As the number of free parameters incresases quadratically with the number of responses and the covariates, analyses based on the proposed models can become unfeasible in practical applications. These problems have been overcome by introducing constraints on the elements of the covariance matrices according to an approach based on the eigen-decomposition of the covariance matrices. The performances of the new models have been studied by simulations and using real datasets in comparison with other models. In order to gain additional flexibility, mixtures of seemingly unrelated contaminated normal regressions models have also been specified so as to allow mixing proportions to be expressed as functions of concomitant covariates.

The content of this thesis is organized as follows. In Chapter 1, a brief summary of the state of the art is presented. The general specification of the new models with fixed covariates and including the fully unconstrained parameterisation for the covariance matrices is presented in Chapter 2. In Chapter 3, the latter methodology is

extended to admit more parsimonious parameterisations. The new models developed under the cluster-weighted approach are described in Chapter 4. Chapter 5 contains an illustration of the new models with concomitant variables and a study on housing tension in the municipalities of the Emilia-Romagna region based on different types of multivariate linear regression models.

Every step I take, every move I make, every single day, I will miss you.

To my beloved grandparents, Santa, Cesario and Cosimina.

Contents

1 Introduction **6**

1.1 Overview 6

1.2 Main contributions of the thesis 9

2 Seemingly unrelated clusterwise regression for contaminated data **11**

2.1 Introduction 13

2.2 Seemingly unrelated contaminated Gaussian linear clusterwise regression analysis 17

2.2.1 Seemingly unrelated contaminated Gaussian linear clusterwise regression models 17

2.2.2 Comparisons with other linear clusterwise regression models 19

2.2.3 Identifiability 20

2.2.4 Maximum likelihood estimation 22

2.2.5 Technical details about the ECM algorithm 26

2.2.6 Determining the value of K 27

2.3 Results from Monte Carlo studies 28

2.3.1 Settings 28

2.3.2 Results 30

2.4 Results from the analysis of canned tuna sales 48

2.5 Conclusions 52

3 Parsimonious Mixtures of Seemingly Unrelated Contaminated Normal Regression Models **61**

3.1 Introduction 63

3.2 Parsimonious SU contaminated normal regression mixtures 64

3.3 Analysis of U.S. canned tuna sales 67

3.4 Conclusions . . . 69

4 Parsimonious seemingly unrelated contaminated normal cluster-weighted models 73

4.1 Introduction . . . 75

4.2 Seemingly unrelated contaminated normal cluster-weighted analysis . . . 79

4.2.1 Seemingly unrelated contaminated normal cluster-weighted models . . . 79

4.2.2 Comparisons with other mixture regression models . . . 81

4.2.3 Identifiability . . . 83

4.2.4 An ECM algorithm for ML estimation . . . 84

4.2.5 Technical details about the ECM algorithm . . . 89

4.2.6 Determining the value of K . . . 91

4.2.7 Parsimonious models . . . 92

4.3 Simulation studies . . . 93

4.3.1 Settings . . . 93

4.3.2 Results . . . 94

4.4 Analysis of canned tuna sales . . . 114

4.5 Conclusions . . . 122

4.6 Data availability . . . 123

5 A study on housing tension in the municipalities of the Emilia-Romagna region 128

5.1 Introduction . . . 130

5.2 Housing deprivation in Italy . . . 131

5.3 The Emilia-Romagna Region (ERR) . . . 134

5.4 Dataset . . . 135

5.4.1 Socio - Demographic (SD) indicators . . . 138

5.4.2 Social Life and Income Condition (SLIC) indicators . . . 141

5.4.3 Housing Supply and Housing Market (HSHM) indicators . . . 144

5.4.4 Housing tension indicators . . . 147

5.5 Aim of this study . . . 152

5.6 Methods . . . 154

5.7 Results . . . 158

5.8 Conclusions 174

A R functions **180**

A.1 MCSUN 180

A.2 SuCNCW 197

Chapter 1

Introduction

1.1 Overview

In the last decades, the amount of scientific publications focused on the management of complex data has increased exponentially. This growth has been motivated by the constant researchers' need to develop faster and more accurate methods capable of eliciting information from this kind of datasets. Many complexities can affect the data depending on the field of the research. In multivariate regression analysis, for example, the interest of the researcher in modelling the dependence of M dependent variables $\mathbf{Y}$ on P given predictors $\mathbf{X}$ can become more difficult in a situation where the population from which the sample $\mathcal{S}$ comes from is heterogeneous (i.e. it is composed of K unknown disjoint and homogeneous sub-populations); thus, the information about the specific sub-population each sample observation belongs to is missing. A useful way to manage the possible presence of K unknown clusters in the sample $\mathcal{S}$ while performing multivariate regression analysis is to suitably embed a mixture of K distributions into the regression model. Another source of complexity can arise from the fact that the covariates are not always actively manipulated by the researchers. In particular, if the covariates are under the control of the researcher, then $\mathbf{X}$ should be treated as fixed; otherwise, both $\mathbf{X}$ and $\mathbf{Y}$ have to be considered as random vectors. Thus, either a conditional density function $f(\mathbf{y}|\mathbf{x})$ or a joint density function $f(\mathbf{x}, \mathbf{y})$ should be utilized for modelling the conditional distribution of $\mathbf{Y}|\mathbf{X}$ or the joint distribution of $(\mathbf{X}, \mathbf{Y})$, respectively, where $f(\mathbf{x}, \mathbf{y}) = f(\mathbf{x})f(\mathbf{y}|\mathbf{x})$. Based on the two above mentioned sources of complexity, the following approaches can be employed to perform multivariate regression analysis:

(a) clusterwise regression analysis,

(b) cluster-weighted analysis.

In particular, the approach (a) is useful when the unobserved heterogeneity affects $\mathbf{Y}|\mathbf{X}$ and the covariates are fixed; in the framework (b), instead, the covariates are treated as random and the missing information about the membership to the K sub-populations affects $(\mathbf{X}, \mathbf{Y})$. Then, a mixture of K different regression models (one for each sub-population) will describe either the distribution of $\mathbf{Y}|\mathbf{X}$ or the distribution of $(\mathbf{X}, \mathbf{Y})$ in the population, respectively. If the vector $\mathbf{Y}$ is composed of M continuous responses, then Gaussian clusterwise linear regression models (Jones and McLachlan, 1992) are generally employed. When all the variables are continuous, a Gaussian cluster-weighted model (Dang et al., 2017) is usually specified within the approach (b). The prediction of the responses in such approaches can become even more difficult in the following situations.

(I) In economics or social sciences, there may be prior information about the regressors expected to be relevant in the prediction of the M responses. In such situations, the multivariate regression model specified by the researcher should be composed of a system of M regression equations (one equation for each response) with equation-dependent vectors of predictors (i.e., vectors which do not necessarily contain the same predictors for all the responses). This means that certain regressors contained in $\mathbf{X}$ are absent from certain regression equations. Furthermore, the M responses contained in $\mathbf{Y}$ may be correlated. This latter feature is typically observed with multivariate longitudinal data, time-series data or repeated measures. A parametric framework able to take into consideration both multivariate correlated responses and systems of regression equations with equation-dependent vectors of predictors is given by the so-called seemingly unrelated regression approach (see, e.g., Srivastava and Giles, 1987; Park, 1993).

(II) The data $\mathcal{S}$ are contaminated by the presence of mildly atypical observations (Ritter, 2015), i.e. observations which, in some way, deviate from the general pattern of the data (Maronna et al., 2006). Several robust methods have been developed in the literature by resorting to heavy-tailed models (e.g., Lange et al. (1989), Kibria and Haq (1999), Lachos et al. (2011)). A solution proposed by Tukey (1960) is based on the use of the contaminated normal distribution. This distribution is a two-component normal mixture in which one component has a larger probability and represents the typical observations; the other component has the same expected vector of the first one but an inflated covariance

matrix, which allows to manage the outliers. In a multivariate regression framework, suitable models able to manage the presence of mildly atypical observations have been obtained by specifying a contaminated normal distribution for $\mathbf{Y}|\mathbf{X}$ within the approach (a) or for both $\mathbf{X}$ and $\mathbf{Y}|\mathbf{X}$ within the approach (b). In particular, clusterwise linear regression models and cluster-weighted models have been specified so as to be able to manage, respectively:

(IIa) outliers in the $\mathbf{y}$-direction (verticals or regression outliers);

(IIb) outliers either in the $\mathbf{y}$-direction or in the $\mathbf{x}$-direction (leverage points), depending on whether they occur in the responses or the predictors, respectively (see, e.g., Rousseeuw and Leroy, 2005); if an observation is both a regression outlier and a leverage point it will be classified as a bad leverage point (Rousseeuw and Leroy, 2005).

Based on all these considerations, until now authors have developed the following classes of models for multivariate linear regression analysis:

(i) contaminated Gaussian clusterwise linear regression models (Mazza and Punzo, 2020), which allow to manage the presence of (IIa) within the approach (a);

(ii) seemingly unrelated Gaussian clusterwise linear regression models (Galimberti and Soffritti, 2020) for data affected by the complexity (I) under the approach (a);

(iii) contaminated Gaussian cluster-weighted models (Punzo and McNicholas, 2017) able to manage (IIb) in the approach (b);

(iv) seemingly unrelated Gaussian cluster-weighted models (Diani et al., 2022) for data affected by the complexity (I) under the approach (b).

On the one hand, limitations of the approaches (i) and (iii) are represented by the fact that the same vector of regressors has to be employed for the prediction of all responses. On the other hand, methods (ii) and (iv) are not robust against the presence of atypical observations in the K sub-populations. The aim of this thesis is to extend such approaches so as to:

- jointly account for the sources of complexity (I) and (IIa) under the approach (a);
- jointly account for the sources of complexity (I) and (IIb) under the approach (b).

1.2 Main contributions of the thesis

In Chapter 2, a new class of models for multivariate regression defined by finite mixtures of seemingly unrelated contaminated normal regressions has been developed. This class provides an extension to mixture-based regression analysis for modelling multivariate and correlated responses in the presence of atypical observations in the $\mathbf{y}$-direction that let the researcher free to use a different vector of covariates for each response. Conditions for the identifiability of such models are provided. An expectation-conditional maximisation (ECM) algorithm for maximum likelihood estimation (MLE) of the model parameters has been derived. The performance of the new models has been studied by simulation in comparison with other clusterwise linear regression models. A comparative evaluation of their effectiveness and usefulness has been provided through the analysis of a real dataset. The main results have been summarized in the following paper:

Perrone G., Soffritti G. (2023). "Seemingly unrelated clusterwise linear regression for contaminated data". *Stat Papers* 64, 883–921. https://doi.org/10.1007/s00362-022-01344-6.

As the number of free parameters incresases quadratically with the number of responses, analyses based on the models illustrated in the first part of this thesis can become unfeasible in practical applications in which M is large. This problem has been overcome by introducing in Chapter 3 constraints on the elements of the covariance matrices according to an approach due to Celeux and Govaert (1995). The resulting parsimonious finite mixtures of seemingly unrelated contaminated normal regressions are illustrated in the second part of this thesis, whose source is the following short paper:

Perrone G., Soffritti G. (2022). "Parsimonious mixtures of seemingly unrelated contaminated normal regression models". In P. Brito, J. G. Dias, B. Lausen, A. Montanari, R. Nugent. *Classification and Data Science in the Digital Age: the 17th Conference of the International Federation of Classification Societies (IFCS 2022), Springer Cham. Series E-ISSN: 2198-3321* (pp. 1-8) https://link.springer.com/book/9783031090332 (in press).

In clusterwise regression analysis, where covariates are treated as fixed, the assignment of the

data points to the K clusters is assumed to not depend on $\mathbf{X}$ (*assignment independence*, Hennig (2000)). This could be inadequate in some practical applications in which the assignment of the sample data points to the K clusters is not independent of the covariates (*assignment dependence*, Hennig (2000)). In order to overcome this limitation, a novel class of cluster-weighted models has been introduced in Chapter 4.

This class allows to manage the presence of atypical observations either in the $\mathbf{x}$-direction or in the $\mathbf{y}$-direction; furthermore, it makes it possible to specify a different vector of covariates for each dependent variable. Conditions for the identifiability of such models are described. Parsimonious models are presented; they have been obtained by constraining some elements of the covariance matrices of both the covariates and responses. A new ECM algorithm for the MLE of the model parameters has been developed. The effectiveness and usefulness of such models are shown through the analysis of simulated and real datasets. The main results have been summarized in the following paper:

Perrone G., Soffritti S. (2022). "Parsimonious seemingly unrelated contaminated normal cluster-weighted models". Under review.

More flexible seemingly unrelated clusterwise linear regression models have been specified so as to allow some covariates to influence the prior probabilities of the K sub-populations. This task has been performed by modelling the mixing weights as a function of some concomitant variables. Such variables can be different of the ones used in the prediction of the dependent variables and in the identification of the clusters. Such models, together with other clusterwise linear regression models, have been employed in Chapter 5 to study housing tension in the municipalities of the Emilia-Romagna region. This research has been carried out thanks to an implementation agreement between the region and the Department of Statistical Sciences of the University of Bologna.

Chapter 2

Seemingly unrelated clusterwise regression for contaminated data[1]

[1]This chapter coincides with the published paper: Perrone G., Soffritti G. (2023). "Seemingly unrelated clusterwise linear regression for contaminated data". *Stat Papers* 64, 883–921. https://doi.org/10.1007/s00362-022-01344-6

Abstract

Clusterwise regression is an approach to regression analysis based on finite mixtures which is generally employed when sample observations come from a population composed of several unknown sub-populations. Whenever the response is continuous, Gaussian clusterwise linear regression models are usually employed. Such models have been recently robustified with respect to the possible presence of mild outliers in the sub-populations. However, in some fields of research, especially in the modelling of multivariate economic data or data from the social sciences, there may be prior information on the specific covariates to be considered in the linear term employed in the prediction of a certain response. As a consequence, covariates may not be the same for all responses. Thus, a novel class of multivariate Gaussian linear clusterwise regression models is proposed. This class provides an extension to mixture-based regression analysis for modelling multivariate and correlated responses in the presence of mild outliers that let the researcher free to use a different vector of covariates for each response. Details about the model identification and maximum likelihood estimation via an expectation-conditional maximisation algorithm are given. The performance of the new models is studied by simulation in comparison with other clusterwise linear regression models. A comparative evaluation of their effectiveness and usefulness is provided through the analysis of a real dataset.

Keywords: Contaminated Gaussian distribution, ECM algorithm, Mild outlier, Mixture of regression models, Model-based cluster analysis, Seemingly unrelated regression.

2.1 Introduction

In multivariate regression analysis, when modelling the dependence of a random vector $\mathbf{Y} = (Y_1, \ldots, Y_m, \ldots, Y_M)'$ of M responses on a given vector $\mathbf{X} = (X_1, \ldots, X_p, \ldots, X_P)'$ of P predictors through a sample $\mathcal{S} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_I, \mathbf{y}_I)\}$ drawn from a certain population, the following sources of complexity could affect the data and make the prediction of the responses a task difficult to perform.

a) With multivariate longitudinal data, time-series data or repeated measures, the M responses contained in $\mathbf{Y}$ are typically correlated. Furthermore, in analyses of economic data or data from the social sciences, it is not unusual that prior information about the phenomenon under study enables the analyst to specify a system of M regression equations (one equation for each response) in which certain regressors contained in $\mathbf{X}$ are absent from certain regression equations. This is especially true for multivariate economic data referring to general theories (i.e., investment equations, production functions) or applications dealing with the explanation of a certain economic activity (i.e., demand of petrol, employment) in different geographical locations (see, e.g., Zellner, 1962; White and Hewings, 1982; Giles and Hampton, 1984). Further examples can be found also in other fields, such as medicine, food quality, tourism economics, quality of life and health (see, e.g., Keshavarzi et al., 2012; Cadavez and Hennningsen, 2012; Keshavarzi et al., 2013; Disegna and Osti, 2016; Heidari et al., 2017). A parametric framework able to take into consideration both multivariate correlated responses and systems of regression equations with equation-dependent vectors of predictors (i.e., vectors which do not necessarily contain the same predictors for all the responses) is given by the so-called seemingly unrelated regression approach (see, e.g., Srivastava and Giles, 1987; Park, 1993). In particular, in this approach the random disturbances associated with the M regression equations are allowed to be correlated with each other; hence, the variance-covariance matrix $\mathbf{\Sigma}$ of the resulting M-dimensional vector of the error terms will have a non-diagonal structure.

b) In general, real data can often be characterised by the presence of atypical observations. In parametric regression analysis, such observations negatively impact on both the estimation of the regression coefficients and the prediction of the responses based on the classical procedures. Such procedures have been widely recognized to be extremely sensitive to even seemingly minor or negligible deviations from some conventional assumptions (see,

e.g., Tukey, 1960). Thus, when the data are contaminated by such observations, it is crucial that robust methods are employed (see, e.g., Maronna et al., 2006). Departures from the Gaussian distribution of the error terms in the regression model caused by some mildly atypical observations can be managed by simply resorting to heavy-tailed distributions for $\mathbf{Y}|\mathbf{X} = \mathbf{x}$. Those observations are also called small or mild outliers (see, e.g., Ritter, 2015). Examples of robust methods against the presence of such outliers have been developed by Lange et al. (1989), Kibria and Haq (1999), Lachos et al. (2011); to this end, the multivariate t distribution or scale mixtures of Gaussian distributions have been exploited. Other distributions, such as the multivariate power-exponential (see, e.g., Gómez et al., 1998; Dang et al., 2015), the multivariate leptokurtic-normal distribution (Bagnato et al., 2017), the multivariate tail-inflated normal distribution (Punzo and Bagnato, 2021), and the multivariate shifted exponential normal distribution (Punzo and Bagnato, 2020), have been employed to cope with the same issue. Another model able to manage the possible presence of mild outliers in a dataset is the contaminated Gaussian distribution (see, e.g., Tukey, 1960; Aitkin and Wilson, 1980). This probabilistic model is defined as a mixture of two Gaussian distributions having the same expected mean value but different variances-covariances. Furthermore, the Gaussian distribution having the smallest mixing weight also has inflated variances-covariances and is employed to represent the mild outliers. Maximum likelihood (ML) estimation can be performed via an expectation-maximisation (EM) algorithm (see Dempster et al., 1977; Aitkin and Wilson, 1980). Once such a model is fitted to the observed data, each sample observation can be classified as either typical or outlier using the maximum a posteriori probability (for further details see, e.g., Aitkin and Wilson, 1980). With an approach based on the use of one of these distributions, robustness can be achieved without suppressing any observation from the sample $\mathcal{S}$.

c) Sometimes the population from which the sample $\mathcal{S}$ comes from is composed of a certain number, say K, of sub-populations. Furthermore, when the information about the value of K and the specific sub-population each sample observation belongs to is not known, $\mathcal{S}$ is characterised by unobserved heterogeneity. If this source of heterogeneity affects the distribution of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$, then a mixture of K different regression models (one for each sub-population) will describe the distribution of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ in the population. This phenomenon can be experienced in many fields, such as economics, marketing, agriculture, education, human genomics, quantitative finance, social sciences and transport systems

(see, e.g., Fair and Jaffe, 1972; Kamakura, 1988; Turner, 2000; Ding, 2006; Qin and Self, 2006; Tashman and Frey, 2009; Dyer et al., 2012; Van Horn et al., 2015; McDonald et al., 2016; Elhenawy et al., 2017). In this case, the sample $\mathcal{S}$ should be analysed in a regression framework able to detect both the number of sub-populations and their regression models. Methods for clusterwise regression analysis play a special role. They exploit clusterwise regression models, which are mixtures of K regression models (see, e.g., Hosmer, 1974; De Sarbo and Cron, 1988; Frühwirth-Schnatter, 2006; Depraetere and Vandebroek, 2014). In these models, the mixing weights can also be expressed as a function of some concomitant variables (Wedel, 2002). With M continuous responses in vector $\mathbf{Y}$, multivariate Gaussian clusterwise linear regression models are generally employed (see, e.g., Jones and McLachlan, 1992). If the P predictors are random and the source of heterogeneity mentioned above affects the distribution of $(\mathbf{X}, \mathbf{Y})$, then Gaussian cluster-weighted models should be employed (see, e.g., Dang et al., 2017).

Recently, Mazza and Punzo (2020) have introduced methods to perform Gaussian clusterwise linear regression analysis which are robust with respect to heavy-tailed departures from Gaussianity due to the presence of mild outliers in the data. By relying on contaminated Gaussian clusterwise linear regression models, their methods are able to produce a simultaneous clustering of the sample observations and the detection of mild outliers in a multivariate regression context. In this way, they allow to manage the sources of complexity *b)* and *c)*; they are also capable of explaining the correlation among responses. A limitation of an approach based on those models is that the same vector of regressors has to be employed for the prediction of all responses. Galimberti and Soffritti (2020) have developed models for Gaussian clusterwise linear regression which make use of seemingly unrelated regression equations. The methods based on these latter models are suitable for the analysis of data affected by complexities *a)* and *c)*; however, they are not insensitive to the possible presence of mild outliers in the K sub-populations. Based on all these considerations, multivariate seemingly unrelated clusterwise linear regression models for data contaminated by mild outliers are introduced here. They are obtained from the models described in Mazza and Punzo (2020) by modifying the definition of the linear terms in the M regression equations so that a different vector of regressors can be employed for each dependent variable. With these new models, the three sources of complexities mentioned above are jointly taken into consideration when predicting the responses in a multivariate linear regression framework. Thus, a more flexible approach for the analysis of linear dependencies in multivariate data

is provided.

The key contributions of this chapter are:

- the specification of a novel class of models able to jointly account for the sources of complexity *a)*, *b)* and *c)* mentioned above;
- a comparison with some other linear clusterwise regression models;
- the description of conditions for the identifiability of the novel models;
- details about ML estimation via an expectation-conditional maximisation (ECM) algorithm (Meng and Rubin, 1993);
- a treatment of the initialisation and convergence of the ECM algorithm and the issue of model selection;
- an investigation of the effectiveness of the new models, based on simulated datasets, in comparison with the models proposed by Galimberti and Soffritti (2020) and Mazza and Punzo (2020);
- an application to a study of the effects of prices and promotional activities on sales for two U.S. brands of canned tuna.

The remainder of this chapter is organised as follows. The novel models are introduced in Section 2.2.1. Section 2.2.2 shows how they relate to some clusterwise linear regression models. Identifiability is treated in Section 2.2.3. Section 2.2.4 and Appendix A provide details on the ECM algorithm. Issues of algorithm initialisation, convergence criterion and model selection are discussed in Sections 2.2.5 and 2.2.6. Section 2.3 contains a summary of the experimental results obtained from the analysis of simulated data. The study of the effects of prices and promotional activities on U.S. canned tuna sales is presented in Section 2.4. Finally, in Section 2.5, some concluding remarks and ideas for future research are illustrated.

2.2 Seemingly unrelated contaminated Gaussian linear clusterwise regression analysis

2.2.1 Seemingly unrelated contaminated Gaussian linear clusterwise regression models

In order to introduce the new model, the following notation is required. Suppose that only P_m of the P covariates contained in $\mathbf{X}$ are considered to be relevant for the prediction of the response Y_m, where $P_m \leq P$. Thus, let $\mathbf{X}_m = (X_{m_1}, X_{m_2}, \ldots, X_{m_{P_m}})'$ be the vector composed of such P_m covariates, and let $\mathbf{X}_m^* = (1, \mathbf{X}_m')'$. Furthermore, let $\boldsymbol{\beta}_{km} = (\beta_{k,m_1}, \beta_{k,m_2}, \ldots, \beta_{k,m_{P_m}})'$ be the vector of the P_m regression coefficients capturing the linear effect of such covariates on the response Y_m in the kth sub-population, and $\boldsymbol{\beta}_{km}^* = (\beta_{0k,m}, \boldsymbol{\beta}_{km}')'$. Then, the vector containing all linear effects on the M responses in the kth sub-population can be obtained by stacking the M regression coefficient vectors specific for the kth sub-population one underneath the other; it can be denoted as $\boldsymbol{\beta}_k^* = (\boldsymbol{\beta}_{k1}^{*\prime}, \ldots, \boldsymbol{\beta}_{km}^{*\prime}, \ldots, \boldsymbol{\beta}_{kM}^{*\prime})'$ and its length is $P^* + M$, where $P^* = \sum_{m=1}^{M} P_m$. Finally, the following $(P^* + M) \times M$ partitioned matrix is required:

$$\tilde{\mathbf{X}}^* = \begin{bmatrix} \mathbf{X}_1^* & \mathbf{0}_{P_1+1} & \ldots & \mathbf{0}_{P_1+1} \\ \mathbf{0}_{P_2+1} & \mathbf{X}_2^* & \ldots & \mathbf{0}_{P_2+1} \\ \vdots & \vdots & & \vdots \\ \mathbf{0}_{P_M+1} & \mathbf{0}_{P_M+1} & \ldots & \mathbf{X}_M^* \end{bmatrix},$$

where $\mathbf{0}_{P_m+1}$ denotes the $(P_m + 1)$-dimensional null vector.

The random vector $\mathbf{Y}$ follows a seemingly unrelated contaminated Gaussian linear clusterwise regression model of order K if the conditional probability density function (p.d.f.) of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ has the form

$$f(\mathbf{y}|\mathbf{x}; \boldsymbol{\psi}) = \sum_{k=1}^{K} \pi_k h\left(\mathbf{y}; \boldsymbol{\theta}_k\right), \ \mathbf{y} \in \mathbb{R}^M, \tag{2.1}$$

where π_k is the mixing weight of the kth sub-population, with $\pi_k > 0$ for $k = 1, \ldots, K$, and $\sum_{k=1}^{K} \pi_k = 1$; $h\left(\mathbf{y}; \boldsymbol{\theta}_k\right)$ is the contaminated Gaussian p.d.f. of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ in the kth sub-population, defined as follows:

$$h\left(\mathbf{y}; \boldsymbol{\theta}_k\right) = \alpha_k \phi_M\left(\mathbf{y}; \boldsymbol{\mu}_k(\mathbf{x}; \boldsymbol{\beta}_k^*), \boldsymbol{\Sigma}_k\right) + (1 - \alpha_k)\phi_M\left(\mathbf{y}; \boldsymbol{\mu}_k(\mathbf{x}; \boldsymbol{\beta}_k^*), \eta_k \boldsymbol{\Sigma}_k\right), \tag{2.2}$$

and $\phi_M(\cdot;\boldsymbol{\mu},\boldsymbol{\Sigma})$ denotes the p.d.f. of an M-dimensional Gaussian distribution with expected mean vector $\boldsymbol{\mu}$ and positive definite covariance matrix $\boldsymbol{\Sigma}$. The term $\boldsymbol{\mu}_k(\mathbf{x};\boldsymbol{\beta}_k^*)$ in equation (2.2) is the conditional expected value of $\mathbf{Y}|\mathbf{X}=\mathbf{x}$ in the kth sub-population; it is defined as follows:

$$\boldsymbol{\mu}_k(\mathbf{x};\boldsymbol{\beta}_k^*) = \tilde{\mathbf{x}}^{*\prime}\boldsymbol{\beta}_k^* = \begin{bmatrix} \mathbf{x}_1^{*\prime}\ \boldsymbol{\beta}_{k1}^* \\ \vdots \\ \mathbf{x}_m^{*\prime}\ \boldsymbol{\beta}_{km}^* \\ \vdots \\ \mathbf{x}_M^{*\prime}\ \boldsymbol{\beta}_{kM}^* \end{bmatrix}, \tag{2.3}$$

where $\tilde{\mathbf{x}}^*$ denotes the realisation of $\tilde{\mathbf{X}}^*$ obtained when $\mathbf{X}=\mathbf{x}$. Thus, $\tilde{\mathbf{x}}^{*\prime}\boldsymbol{\beta}_k^*$ coincides with an M-dimensional vector whose mth element is a linear combination of the realisations of the P_m regressors selected for the prediction of Y_m with weights given by the elements of vector $\boldsymbol{\beta}_{km}^*$. Terms $\alpha_k \in (0,1)$ and $\eta_k > 1$ are the weight of the typical observations in the kth sub-population and the factor contaminating the conditional variances and covariances of $\mathbf{Y}|\mathbf{X}=\mathbf{x}$ for the mild outliers in the kth sub-population, respectively. In robust statistics, it is generally assumed that at least half of the observations are typical (see, e.g., Punzo and McNicholas, 2016; Mazza and Punzo, 2020); thus, it is also possible to consider $\alpha_k \in [0.5,1)$. As a consequence of the constraint $\eta_k > 1$, η_k represents an inflation parameter for the elements of $\boldsymbol{\Sigma}_k$. $\boldsymbol{\theta}_k = (\boldsymbol{\beta}_k^*, \boldsymbol{\Sigma}_k, \alpha_k, \eta_k)$ is the parameter vector of model (2.2). The parameter vector of model (2.1) is given by $\boldsymbol{\psi} = (\boldsymbol{\psi}_1,\ldots,\boldsymbol{\psi}_k,\ldots,\boldsymbol{\psi}_K)$, where $\boldsymbol{\psi}_k = (\pi_k, \boldsymbol{\theta}_k)$; the number of free parameters in this vector is equal to $n_{\boldsymbol{\psi}} = 3K - 1 + K(P^* + M) + K\frac{M(M+1)}{2}$.

In summary, the conditional p.d.f. $f(\mathbf{y}|\mathbf{x};\boldsymbol{\psi})$ in equation (2.1) can be interpreted as a weighted average (namely, a mixture) of K Gaussian regression models with weights π_k, $k = 1,\ldots,K$. The kth component of this mixture represents a multivariate seemingly unrelated contaminated Gaussian linear regression model with intercepts and regression coefficients $\boldsymbol{\beta}_k^*$, symmetric and positive definite covariance matrix $\boldsymbol{\Sigma}_k$, proportion of typical points α_k and inflation parameter η_k. Thanks to the non-diagonal structure of the variance-covariance matrices $\boldsymbol{\Sigma}_k$, $k = 1,\ldots,K$, the proposed model is able to account for correlated random disturbances within each of the K sub-populations associated with the mixture (2.1). Since the contaminated Gaussian distribution (2.2) is a mixture of two Gaussian linear regression models which are both associated with the kth component of the mixture in equation (2.1), the model defined

by this latter equation can also be considered as a mixture of $2K$ seemingly unrelated Gaussian clusterwise linear regression models, whose components can be grouped into K pairs, each of which contains two Gaussian components having the same expected values and proportional covariance matrices.

2.2.2 Comparisons with other linear clusterwise regression models

When specific conditions are met, some special linear regression models can be obtained from model (2.1).

- If $M > 1$ and $\mathbf{X}_m = \mathbf{X}\ \forall m$ (the same vector of predictors is considered for all responses), the following equality holds: $\tilde{\mathbf{x}}^* = \mathbf{I}_M \otimes \mathbf{x}^*$, where $\mathbf{I}_M$ is the identity matrix of order M and $\otimes$ denotes the Kronecker product operator (see, e.g., Magnus and Neudecker, 1988). Equation (2.3) can be rewritten as

$$\boldsymbol{\mu}_k(\mathbf{x}; \boldsymbol{\beta}_k^*) = (\mathbf{I}_M \otimes \mathbf{x}^*)' \boldsymbol{\beta}_k^* = \mathbf{B}_k' \mathbf{x},\ k = 1, \ldots, K, \tag{2.4}$$

  where $\mathbf{B}_k = [\boldsymbol{\beta}_{k1}^* \cdots \boldsymbol{\beta}_{km}^* \cdots \boldsymbol{\beta}_{kM}^*]$. Thus, equation (2.1) reduces to the mixture of multivariate contaminated Gaussian regression models introduced by Mazza and Punzo (2020).

- If $M > 1$, $\alpha_k \to 1$ and $\eta_k \to 1\ \forall k$ (there is no contamination in the data), the resulting model coincides with the mixture of multivariate seemingly unrelated linear regressions described in Galimberti and Soffritti (2020).

- If $\alpha_k \to 1$, $\eta_k \to 1\ \forall k$ and $\mathbf{X}_m = \mathbf{X}\ \forall m$ (there is no contamination in the data and the same vector of predictors is considered for all responses), equation (2.1) reduces to a mixture of either univariate Gaussian linear regression models (see, e.g., De Veaux, 1989; Quandt and Ramsey, 1978; De Sarbo and Cron, 1988) or multivariate Gaussian linear regression models (see Jones and McLachlan, 1992).

- If $\alpha_k \to 1$, $\eta_k \to 1\ \forall k$, $\mathbf{X}_m = \mathbf{X}\ \forall m$ and $\boldsymbol{\beta}_k^* = \boldsymbol{\beta}^*\ \forall k$ (there is no contamination in the data, the same vector of predictors is considered for all responses and their effects are the same across all the sub-populations), the resulting model coincides with a linear regression model with error terms distributed according to a mixture of K either univariate Gaussian distributions (Bartolucci and Scaccia, 2005) or multivariate Gaussian distributions (Soffritti and Galimberti, 2011).

- If $M > 1$, $\alpha_k \to 1$, $\eta_k \to 1$ $\forall k$, $\boldsymbol{\beta}_k^* = \boldsymbol{\beta}^*$ $\forall k$ (there is no contamination in the data and the effects of the predictors are the same across all the sub-populations), a multivariate seemingly unrelated linear regression model whose error terms are assumed to follow a Gaussian mixture model is obtained (Galimberti et al., 2016).

Seemingly unrelated regression models represent multivariate regression models in which prior information about the absence of certain covariates for the prediction of certain responses is explicitly taken into consideration (Srivastava and Giles, 1987). Thus, equation (2.1) can also be seen as a mixture of multivariate contaminated Gaussian regression models in which some regression coefficients are constrained to be a priori equal to zero. To the best of the authors' knowledge, the inclusion of such constraints in these latter models has not been addressed yet. Models obtained from equation (2.1) by embedding different constraints on the regression coefficients could also be employed in any practical application in which the relevant regressors for each response cannot be established from a priori information and, thus, the choice of the regressors to be used for the M responses is questionable. As it will be illustrated in Section 2.4, in such situations strategies based on a joint use of models (2.1) and variable selection techniques could be devised and employed.

2.2.3 Identifiability

A preliminary requirement for the consistency and other asymptotic properties of the ML estimator is represented by identifiability of the model parameters. Thus, before detailing ML estimation of $\boldsymbol{\psi}$, a discussion about identifiability of model (2.1) is provided here. Consider the class of models $\mathfrak{F} = \{\mathfrak{F}_K, K = 1, \ldots, K_{max}\}$, where $\mathfrak{F}_K = \{f(\mathbf{y}|\mathbf{x}; \boldsymbol{\psi}), \boldsymbol{\psi} \in \boldsymbol{\Psi}\}$, $f(\mathbf{y}|\mathbf{x}; \boldsymbol{\psi})$ is the p.d.f. of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ under the seemingly unrelated contaminated Gaussian linear clusterwise regression model of order K defined in (2.1) and K_{max} denotes the maximum order specified by the researcher for that model. This class is said to be identifiable if, for any two models M, $\tilde{M} \in \mathfrak{F}$ with parameters $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_k, \ldots, \boldsymbol{\psi}_K)$ and $\tilde{\boldsymbol{\psi}} = (\tilde{\boldsymbol{\psi}}_1, \ldots, \tilde{\boldsymbol{\psi}}_k, \ldots, \tilde{\boldsymbol{\psi}}_{\tilde{K}})$, respectively,

$$\sum_{k=1}^{K} \pi_k h\left(\mathbf{y}; \boldsymbol{\theta}_k\right) = \sum_{k=1}^{\tilde{K}} \tilde{\pi}_k h\left(\mathbf{y}; \tilde{\boldsymbol{\theta}}_k\right) \;\; \forall \; \mathbf{y} \in \mathbb{R}^M$$

implies that $K = \tilde{K}$ and $\boldsymbol{\psi} = \tilde{\boldsymbol{\psi}}$.

Several types of non-identifiability can affect the model class $\mathfrak{F}$. A first type is due to

invariance to relabeling the components of the mixture (also known as label-switching). Non-identifiability can also be caused by potential overfitting associated with empty components or equal components (see, e.g., Frühwirth-Schnatter, 2006). Imposing suitable constraints on the parameter space $\boldsymbol{\Psi}$ can prevent such sources of non-identifiability for $\mathfrak{F}$. Another type of non-identifiability affecting this class is specifically associated with the use of finite mixtures in linear regression analysis with fixed covariates, which requires an additional constraint on the number of components of the mixture (2.1) (see Hennig, 2000). Non-identifiability due to empty components is avoided by requiring the positivity of all the mixing weights π_k. Conditions specifically devised for ensuring identifiability of mixtures of contaminated Gaussian regression models are provided in Mazza and Punzo (2020). These results have been exploited in Theorem 1 to show that model (2.1) is identifiable if the parameters $(\boldsymbol{\beta}_k^*, \boldsymbol{\Sigma}_k)$, $k = 1, \ldots, K$, are pairwise distinct and the order K is exceeded by the number of distinct $(P_m - 1)$-dimensional hyperplanes required to cover the covariates employed for the prediction of Y_m, for $m = 1, \ldots, M$. In order to state Theorem 1, the following notation is also required: $\|\cdot\|_F$ is the element-wise matrix 2-norm (also known as the Frobenious norm); $H^{P_m-1} = \{\mathbf{x}_m \in \mathbb{R}^{P_m} : \boldsymbol{\lambda}'\mathbf{x}_m = c, \boldsymbol{\lambda} \in \mathbb{R}^{P_m}, \boldsymbol{\lambda} \neq \mathbf{0}\}$ is a $(P_m - 1)$-dimensional hyperplane; J_m is the minimum number of such hyperplanes required to cover the covariates $\mathbf{x}_m$; $\mathcal{H}^{P_m-1}$ is the space of $(P_m - 1)$-dimensional hyperplanes of $\mathbb{R}^{P_m}$.

Theorem 1. *Let $M \in \mathfrak{F}$ and $\tilde{M} \in \mathfrak{F}$ be two models, $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_k, \ldots, \boldsymbol{\psi}_K)$ and $\tilde{\boldsymbol{\psi}} = (\tilde{\boldsymbol{\psi}}_1, \ldots, \tilde{\boldsymbol{\psi}}_k, \ldots, \tilde{\boldsymbol{\psi}}_{\tilde{K}})$ the corresponding parameters and, without loss of generality, $K \geq \tilde{K}$. If*

C1) $K < J_m$ for $m = 1, \ldots, M$, where

$$J_m := \min \left\{ q_m : \{\mathbf{x}_{im}, i \in \mathcal{I}_m\} \subseteq \bigcup_{b=1}^{q_m} H_b^{P_m-1} : H_b^{P_m-1} \in \mathcal{H}^{P_m-1} \right\},$$

with $\mathcal{I}_m$ being an index set associated with the distinct covariate points available for the prediction of Y_m, and

C2) $k \neq l$, with $k, l \in \{1, \ldots, K\}$, implies

$$\|\boldsymbol{\beta}_k^* - \boldsymbol{\beta}_l^*\|_F^2 + \|\boldsymbol{\Sigma}_k - a\boldsymbol{\Sigma}_l\|_F^2 \neq 0 \ \forall a > 0,$$

then the class $\mathfrak{F}$ is identifiable.

Conditions *C1)* and *C2)* are obtained from Mazza and Punzo (2020) after suitable modifications of similar conditions required for the identifiability of their mixtures of contaminated

Gaussian regression models. In particular, condition *C2)* results from a simple substitution of the vector $\boldsymbol{\beta}_k^*$ of model (2.1) for the matrix $\mathbf{B}_k$ introduced in equation (2.4) containing the intercepts and regression coefficients in the kth component of the regression mixture model developed by Mazza and Punzo (2020). The modifications involved in the definition of the condition *C1)* derive from the fact that each $Y_m \in \mathbf{Y}$ may have its own covariates and, thus, M different restrictions on K have to be required, each one involving a (possibly) different minimum number of low-dimensional hyperplanes to cover those covariates. As a consequence, the proof of Theorem 1 can be obtained by exploiting the same arguments illustrated in Mazza and Punzo (2020) for the proof of their theorem about identifiability of mixtures of contaminated Gaussian regression models.

2.2.4 Maximum likelihood estimation

The ML estimation of the parameters $\boldsymbol{\psi}$ is carried out here for a fixed value of K. Given a sample $\mathcal{S}$ of I independent observations drawn from model (2.1), the model log-likelihood is equal to $\ell(\boldsymbol{\psi}) = \sum_{i=1}^{I} \ln\left(\sum_{k=1}^{K} \pi_k h\left(\mathbf{y}_i; \boldsymbol{\theta}_k\right)\right)$. Following Mazza and Punzo (2020), ML estimates $\hat{\boldsymbol{\psi}}$ can be computed by means of an ECM algorithm, which represents a variant of the EM algorithm usually employed for the computation of ML estimates from incomplete data. In the considered situation, the missing information is twofold. On the one hand, there is a classical source of incompleteness of any mixture model associated with the component memberships of the I sample observations. On the other hand, it is not known whether such observations are outliers with reference to any component or not. These two sources can be described by two different types of K-dimensional vectors. For the ith sample observation, they are given by $\mathbf{z}_i$ and $\mathbf{u}_i$, respectively: $\mathbf{z}_i = (z_{i1}, \ldots, z_{iK})'$, with $z_{ik} = 1$ if the ith observation comes from the kth component and $z_{ik} = 0$ otherwise; $\mathbf{u}_i = (u_{i1}, \ldots, u_{iK})'$, with $u_{ik} = 1$ if the ith observation is typical in the kth component and $u_{ik} = 0$ if it is an outlier, for $k = 1, \ldots, K$. Then, the set of complete data would be $\mathcal{S}_c = \{(\mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1, \mathbf{u}_1), \ldots, (\mathbf{x}_I, \mathbf{y}_I, \mathbf{z}_I, \mathbf{u}_I)\}$, and the the complete-data likelihood function is equal to

$$L_c(\boldsymbol{\psi}) = \prod_{i=1}^{I}\prod_{k=1}^{K}\Big\{\pi_k\Big[\alpha_k\phi_M\Big(\mathbf{y}_i; \boldsymbol{\mu}_k(\mathbf{x}; \boldsymbol{\beta}_k^*), \boldsymbol{\Sigma}_k\Big)\Big]^{u_{ik}} \Big[(1-\alpha_k)\phi_M\Big(\mathbf{y}_i; \boldsymbol{\mu}_k(\mathbf{x}; \boldsymbol{\beta}_k^*), \eta_k\boldsymbol{\Sigma}_k\Big)\Big]^{1-u_{ik}}\Big\}^{z_{ik}}.$$

Thus, up to an additive constant, the complete-data log-likelihood function employed in the ECM algorithm for the computation of the parameter estimates can be expressed as follows:

$$\ell_c(\boldsymbol{\psi}) = \sum_{i=1}^{I}\sum_{k=1}^{K} z_{ik}\Big[\ln\pi_k + u_{ik}\ln\alpha_k + (1-u_{ik})\ln(1-\alpha_k) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_k| + \\ -\Big(\frac{M}{2}\ln\eta_k\Big)(1-u_{ik}) - \frac{1}{2}\Big(u_{ik} + \frac{1-u_{ik}}{\eta_k}\Big)\delta^2_{\boldsymbol{\Sigma}_k}\Big(\mathbf{y}_i, \boldsymbol{\mu}_k(\mathbf{x}_i;\boldsymbol{\beta}^*_k)\Big)\Big],$$

where

$$\delta^2_{\boldsymbol{\Sigma}_k}\left(\mathbf{y}_i, \boldsymbol{\mu}_k(\mathbf{x}_i;\boldsymbol{\beta}^*_k)\right) = (\mathbf{y}_i - \boldsymbol{\mu}_k(\mathbf{x}_i;\boldsymbol{\beta}^*_k))'\boldsymbol{\Sigma}_k^{-1}(\mathbf{y}_i - \boldsymbol{\mu}_k(\mathbf{x}_i;\boldsymbol{\beta}^*_k)) \tag{2.5}$$

is the squared Mahalanobis distance between $\mathbf{y}_i$ and $\boldsymbol{\mu}_k(\mathbf{x}_i;\boldsymbol{\beta}^*_k)$ with respect to the matrix $\boldsymbol{\Sigma}_k$.

The hth iteration of the E-step in the ECM algorithm consists in calculating the conditional expectation of $l_c(\boldsymbol{\psi})$ on the basis of the current estimate $\boldsymbol{\psi}^{(h)}$ of the model parameters $\boldsymbol{\psi}$; up to an additive constant, this expected value can be expressed as follows:

$$\begin{aligned} Q\left(\boldsymbol{\psi}|\boldsymbol{\psi}^{(h)}\right) &= \mathbb{E}_{\boldsymbol{\psi}^{(h)}}[l_c(\boldsymbol{\psi})] \\ &= \sum_{i=1}^{I}\sum_{k=1}^{K}\hat{z}_{ik}^{(h)}\Big\{\ln\pi_k^{(h)} + \hat{u}_{ik}^{(h)}\ln\alpha_k^{(h)} + (1-\hat{u}_{ik}^{(h)})\ln(1-\alpha_k^{(h)}) + \\ &+ Q_i\Big(\boldsymbol{\beta}^*_k, \boldsymbol{\Sigma}_k|\boldsymbol{\psi}^{(h)}\Big)\Big\}, \end{aligned}$$

where

$$\begin{aligned} Q_i\Big(\boldsymbol{\beta}^*_k, \boldsymbol{\Sigma}_k|\boldsymbol{\psi}^{(h)}\Big) = -\frac{1}{2}\Big[&\ln|\boldsymbol{\Sigma}_k^{(h)}| + M(1-\hat{u}_{ik}^{(h)})\ln\eta_k^{(h)} + \\ &+\Big(\hat{u}_{ik}^{(h)} + \frac{1-\hat{u}_{ik}^{(h)}}{\eta_k^{(h)}}\Big)\delta^2_{\boldsymbol{\Sigma}_k^{(h)}}\Big(\mathbf{y}_i, \boldsymbol{\mu}_k(\mathbf{x}_i;\boldsymbol{\beta}^{*(h)}_k)\Big)\Big], \end{aligned}$$

$\hat{z}_{ik}^{(h)}$ and $\hat{u}_{ik}^{(h)}$ are the posterior probabilities (evaluated using $\boldsymbol{\psi}^{(h)}$) that the ith observation is generated from the kth component of the mixture (2.1) and that the ith observation is a typical

point of such a component, respectively:

$$\hat{z}_{ik}^{(h)} = \mathbb{E}_{\boldsymbol{\psi}^{(h)}}[Z_{ik}|(\mathbf{x}_i, \mathbf{y}_i)] = \frac{\pi_k^{(h)} h\left(\mathbf{y}_i; \boldsymbol{\theta}_k^{(h)}\right)}{f\left(\mathbf{y}_i|\mathbf{x}_i; \boldsymbol{\psi}^{(h)}\right)}, \tag{2.6}$$

$$\hat{u}_{ik}^{(h)} = \mathbb{E}_{\boldsymbol{\psi}^{(h)}}[U_{ik}|(\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)] = \frac{\alpha_k^{(h)} \phi\left(\mathbf{y}_i; \boldsymbol{\mu}_k(\mathbf{x}_i; \boldsymbol{\beta}_k^{*(h)}), \boldsymbol{\Sigma}_k^{(h)}\right)}{h\left(\mathbf{y}_i; \boldsymbol{\theta}_k^{(h)}\right)}, \tag{2.7}$$

with $\mathbf{Z}_i = (Z_{i1}, \ldots, Z_{iK})'$ denoting a K-dimensional multinomial random vector with probabilities $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)'$, and $U_{ik}|Z_{ik} = 1$ having a Bernoulli distribution with success probability of α_k.

As far as the conditional maximisation is concerned, the update of $\boldsymbol{\psi}^{(h)}$ is carried out by considering the following two parameter sub-vectors: $\boldsymbol{\gamma} = (\boldsymbol{\pi}, \boldsymbol{\beta}^*, \boldsymbol{\Sigma}, \boldsymbol{\alpha})$ and $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_K)'$, where $\boldsymbol{\beta}^* = (\boldsymbol{\beta}_1^*, \ldots, \boldsymbol{\beta}_K^*)$, $\boldsymbol{\Sigma} = (\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K)$, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)$. At the $(h+1)$th iteration of the ECM algorithm, $\boldsymbol{\gamma}^{(h)} = (\boldsymbol{\pi}^{(h)}, \boldsymbol{\beta}^{*(h)}, \boldsymbol{\Sigma}^{(h)}, \boldsymbol{\alpha}^{(h)})$ is updated through the maximisation of $Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(h)})$ with respect to $\boldsymbol{\gamma}$ with $\boldsymbol{\eta}$ fixed at $\boldsymbol{\eta}^{(h)}$ (first CM step); then, the update of $\boldsymbol{\eta}^{(h)}$ is carried out by maximising $Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(h)})$ with respect to $\boldsymbol{\eta}$ with $\boldsymbol{\gamma}$ fixed at $\boldsymbol{\gamma}^{(h+1)}$ (second CM step). The resulting updates of $\pi_k^{(h)}$, $\alpha_k^{(h)}$ and $\eta_k^{(h)}$ are:

$$\begin{aligned}
\pi_k^{(h+1)} &= \frac{1}{I}\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}, \\
\alpha_k^{(h+1)} &= \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{u}_{ik}^{(h)}}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}}, \qquad (2.8) \\
\eta_k^{(h+1)} &= \max\left\{1, \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}(1-\hat{u}_{ik}^{(h)}) \delta^2_{\boldsymbol{\Sigma}_k^{(h+1)}}\left(\mathbf{y}_i, \boldsymbol{\mu}_k(\mathbf{x}_i; \boldsymbol{\beta}_k^{*(h+1)})\right)}{M \sum_{i=1}^{I} \hat{z}_{ik}^{(h)}(1-\hat{u}_{ik}^{(h)})}\right\}. \qquad (2.9)
\end{aligned}$$

Such updates coincide with the ones reported in Mazza and Punzo (2020) for their model. Based on equation (2.9), it is possible to highlight that the update $\eta_k^{(h+1)}$ will be larger when the kth component is highly contaminated by the presence of outliers (i.e., when it is characterised by many observations with a small value of $\hat{u}_{ik}^{(h)}$ and a large value of the squared Mahalanobis distance from $\boldsymbol{\mu}_k(\mathbf{x}_i; \boldsymbol{\beta}_k^{*(h+1)})$). As far as the remaining parameters are concerned, their updates

are (details are reported in the Appendix):

$$\boldsymbol{\beta}_k^{*(h+1)} = \Big(\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{ik}^{(h)} \tilde{\mathbf{x}}_i^* \boldsymbol{\Sigma}_k^{(h)^{-1}} \tilde{\mathbf{x}}_i^{*\prime} \Big)^{-1} \Big(\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{ik}^{(h)} \tilde{\mathbf{x}}_i^* \boldsymbol{\Sigma}_k^{(h)^{-1}} \boldsymbol{y}_i \Big), \tag{2.10}$$

$$\boldsymbol{\Sigma}_k^{(h+1)} = \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{ik}^{(h)} \Big(\mathbf{y}_i - \tilde{\mathbf{x}}_i^{*\prime} \boldsymbol{\beta}_k^{*(h+1)} \Big) \Big(\mathbf{y}_i - \tilde{\mathbf{x}}_i^{*\prime} \boldsymbol{\beta}_k^{*(h+1)} \Big)'}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}}, \tag{2.11}$$

where

$$\hat{w}_{ik}^{(h)} = \hat{u}_{ik}^{(h)} + \frac{1 - \hat{u}_{ik}^{(h)}}{\eta_k^{(h)}}. \tag{2.12}$$

It is worth noting that the matrix $\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{ik}^{(h)} \tilde{\mathbf{x}}_i^* \boldsymbol{\Sigma}_k^{(h)^{-1}} \tilde{\mathbf{x}}_i^{*\prime}$ in (2.10) has to be nonsingular; otherwise, the update $\boldsymbol{\beta}_k^{*(h+1)}$ cannot be computed. Equation (2.10) also highlights that this update can be considered as a generalised least squares estimate with weights depending on $\hat{w}_{ik}^{(h)}$; this latter term also affects the update $\boldsymbol{\Sigma}_k^{(h+1)}$ in (2.11), which represents a weighted sum of squared residuals. Using such weights leads to a reduction in the effects of the outliers on the estimation of $\boldsymbol{\beta}_k^{*(h+1)}$; thus, this approach provides robust estimates of $\boldsymbol{\beta}_k^{*(h+1)}$, for $k = 1, \dots K$. Furthermore, based on (2.12), sample observations with the highest posterior estimated probabilities of being generated from the kth component and of representing typical points in the kth component will have the largest impact on the updates of both the regression coefficients and covariances within that component.

Once the convergence is reached and the ML estimates $\hat{\boldsymbol{\psi}}$ are computed, by exploiting equation (2.6) the ECM algorithm provides estimates of the posterior probabilities $\mathbb{P}_{\hat{\boldsymbol{\psi}}}[Z_{ik} = 1|(\mathbf{x}_i, \mathbf{y}_i)] = \hat{z}_{ik}$, $i = 1, \dots, I$, $k = 1, \dots, K$. Such estimated probabilities can be employed to partition the I sample observations into K clusters, by assigning each observation to the component showing the highest posterior probability; for the ith observation:

$$\text{MAP}(\hat{z}_{ik}) = \begin{cases} 1 & \text{if } \max_h\{\hat{z}_{ih}\} \text{ occurs when } h = k; \\ 0 & \text{otherwise.} \end{cases}$$

Furthermore, equation (2.7) allows to compute the estimated posterior probabilities $\mathbb{P}_{\hat{\boldsymbol{\psi}}}[U_{ik} = 1|(\mathbf{x}_i, \mathbf{y}_i, \hat{\mathbf{z}}_i)] = \hat{u}_{ik}$, and an intra-cluster distinction between typical observations and mild outliers can be defined: the ith observation will be classified as an outlier of the hth cluster, where h is the label of the component for which $\text{MAP}(\hat{z}_{ik}) = 1$, if $\hat{u}_{ih} < 0.5$. From the ML estimates $\hat{\boldsymbol{\psi}}$

and equation (2.5) it is also possible to compute the estimated squared Mahalanobis distances $\hat{d}^2_{ik} = \delta^2_{\hat{\boldsymbol{\Sigma}}_k}\left(\mathbf{y}_i, \hat{\boldsymbol{\mu}}_k(\mathbf{x}_i; \hat{\boldsymbol{\beta}}^*_k)\right)$, $i = 1, \ldots, I$, $k = 1, \ldots, K$, which can be employed as multivariate measures of the outlyingness of the I sample observations with respect to the K clusters detected by the model. From the definition of the squared Mahalanobis distance given in equation (2.5) and the expressions for $\hat{u}^{(h)}_{ik}$ and $\hat{w}^{(h)}_{ik}$ reported in equations (2.7) and (2.12), respectively, it is possible to express both $\hat{u}_{ik}$ and $\hat{w}_{ik}$ as decreasing functions of $\hat{d}^2_{ik}$ (see Mazza and Punzo, 2020, for the explicit expressions). Thus, atypical observations could also be detected and studied by considering the values of $\hat{d}^2_{ik}$ $\forall (i,k) \in \{i \in \{1, \ldots, I\}, k : \text{MAP}(\hat{z}_{ik}) = 1\}$ and by focusing on the largest values obtained in this way (see McLachlan and Peel, 2000, p. 232).

2.2.5 Technical details about the ECM algorithm

A crucial point of any EM-based algorithm is the choice of the starting values for the model parameters (i.e., $\boldsymbol{\psi}^{(0)}$). Multiple executions of the algorithm in association with multiple random initialisations or approaches based on non-random choices of either $\boldsymbol{\psi}^{(0)}$ or the missing information can provide a solution (see, e.g., Biernacki et al., 2003; Karlis and Xekalaki, 2003). As far as the ECM algorithm described above is concerned, the initialisation technique illustrated in Mazza and Punzo (2020) could be modified so as to be employed also for model (2.1). This task would require setting the initial values $\hat{z}^{(0)}_{ik}$, $i = 1, \ldots, I$, $k = 1, \ldots, K$, equal to the posterior probabilities from the EM algorithm for the estimation of the seemingly unrelated Gaussian clusterwise linear regression models, which are nested in model (2.1) when $\alpha_k \to 1^-$ and $\eta_k \to 1^+$, $k = 1, \ldots, K$; furthermore, $\hat{u}^{(0)}_{ik} = 0.999$, $i = 1, \ldots, I$, $k = 1, \ldots, K$. Another strategy for the initialisation of $\boldsymbol{\psi}$ which exploits the relationship between model (2.1) and seemingly unrelated Gaussian clusterwise linear regression models (see Section 2.2.2) could be composed of the following three steps. Firstly, a Gaussian mixture model with K components is fitted to the sample residuals of a seemingly unrelated linear regression model (Srivastava and Giles, 1987); this allows to obtain the starting values $\pi^{(0)}_k$ and $\boldsymbol{\Sigma}^{(0)}_k$. Secondly, the starting values $\boldsymbol{\beta}^{*(0)}_k$ are obtained from the fitting of K different seemingly unrelated linear regression models, one for each cluster of the partition associated with the Gaussian mixture model considered in the previous step. Thirdly, $\alpha^{(0)}_k$ and $\eta^{(0)}_k$, $k = 1 \ldots, K$, are set equal to 0.999 and 1.001, respectively. Models involved in the first two steps can be estimated through the packages `mclust` (Scrucca et al., 2017) and `systemfit` (Henningsen and Hamann, 2007) in the `R` environment (R Core Team, 2021). In the analyses of Sections 2.3 and 2.4, the ECM algorithm has been initialised

using this latter strategy. Furthermore, since $(1 - \alpha_k)$ in model (2.1) can be considered as the proportion of outliers in the kth sub-population, when this model is employed for outlier detection, a reasonable requirement is that in each cluster the number of typical observations cannot be smaller than the number of outliers, that is $\alpha_k \in [0.5, 1)\ \forall k$. To guarantee this result, constraints on the estimation of α_k, $k = 1, \ldots, K$, have been included in the ECM algorithm; namely, equation (2.8) has been modified as follows: $\alpha_k^{(h+1)} = \max\left\{0.5, \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{u}_{ik}^{(h)}}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}}\right\}$.

In order to avoid premature stops of the ECM algorithm associated with the use of lack of progress stopping criteria, such as the one based on the difference between the log-likelihood values at two consecutive steps, a convergence criterion based on the Aitken acceleration (Aitken, 1926) has been adopted. It consists in stopping the algorithm when $|\ell_A^{(h+1)} - \ell(\boldsymbol{\psi}^{(h)})| < \epsilon$, where $0 < \epsilon < +\infty$, $\ell_A^{(h+1)}$ is $(h+1)$th Aitken accelerated estimate of the log-likelihood limit, and $\ell(\boldsymbol{\psi}^{(h)})$ is the incomplete log-likelihood evaluated at $\boldsymbol{\psi}^{(h)}$ (see, e.g., McNicholas, 2010). Furthermore, a criterion based on a maximum number of iterations for the ECM algorithm has been employed. In the analyses of Sections 2.3 and 2.4, the maximum number of iterations and ϵ have been set equal to 500 and 10^{-6}, respectively. Furthermore, in order to circumvent the possible issue of unbounded likelihood associated with a degenerate model, the ECM algorithm has been developed by embedding some constraints on the eigenvalues of $\boldsymbol{\Sigma}_k^{(h)}$ for $k = 1, \ldots, K$. Namely, for all estimated covariance matrices, the ratio between the smallest and the largest eigenvalues is required to be not lower than 10^{-10}.

2.2.6 Determining the value of K

As illustrated in Section 2.2.4, the ML estimation of $\boldsymbol{\psi}$ based on the ECM algorithm is carried out for a given number of mixture components. When this number is not known and has to be determined from the data $\mathcal{S}$, it is common practice to employ model selection criteria able to take account of different aspects which are considered relevant when evaluating the adequacy of a model (see, e.g., Frühwirth-Schnatter, 2006; Depraetere and Vandebroek, 2014). For example, the Bayesian Information Criterion (Schwarz, 1978) provides a trade-off between the fit and the model complexity; it can be computed as follows:

$$BIC(K) = 2\ell(\hat{\boldsymbol{\psi}}) - n_{\boldsymbol{\psi}} \ln I.$$

Model selection criteria that also consider the uncertainty of the estimated partition of the sample observations could be employed. An example is represented by the integrated completed likelihood (Biernacki et al., 2000), which can be computed according to different ways of measuring the uncertainty of the estimated partition (see, e.g., Andrews and McNicholas, 2011; Baek and McLachlan, 2011):

$$\begin{aligned} ICL_1(K) &= 2\ell(\hat{\boldsymbol{\psi}}) - n_{\boldsymbol{\psi}} \ln I + 2\sum_{i=1}^{I}\sum_{k=1}^{K} \text{MAP}(\hat{z}_{ik}) \ln \hat{z}_{ik}, \\ ICL_2(K) &= 2\ell(\hat{\boldsymbol{\psi}}) - n_{\boldsymbol{\psi}} \ln I + 2\sum_{i=1}^{I}\sum_{k=1}^{K} \hat{z}_{ik} \ln \hat{z}_{ik}. \end{aligned}$$

These latter criteria penalize complex models more severely than BIC because of the presence of an additional penalty, which represents the estimated mean entropy. Thus, when using these criteria in comparison with the BIC, one cluster should be less likely split into two different components. ICL_1 and ICL_2 differ on whether a soft (i.e., $\hat{z}_{ik}$) or hard (i.e., $\text{MAP}(\hat{z}_{ik})$) clustering is considered in the estimation of the mean entropy. Higher values of these criteria indicate better-fit models; as it will be illustrated in Section 2.4, BIC, ICL_1 and ICL_2 can also be employed to select the predictors to be considered in the linear terms employed in the prediction of the M responses in model (2.1).

2.3 Results from Monte Carlo studies

2.3.1 Settings

This section focuses on the investigation of the effectiveness of models (2.1) (mixtures of contaminated seemingly unrelated Gaussian regressions, hereafter denoted as MCSG) in comparison with other approaches using simulated datasets. This task has been carried out in a multivariate setting with $M = 4$ responses, $P = 4$ covariates and datasets comprising $K = 3$ groups of observations. The additional models considered in the comparison are those described by Mazza and Punzo (2020) and Galimberti and Soffritti (2020). From now on, these latter models have been denoted as MCG (mixtures of contaminated Gaussian regressions) and MSG (mixtures of seemingly unrelated Gaussian regressions), respectively.

The simulated datasets have been generated using three different data generation processes:

(a) MSG;

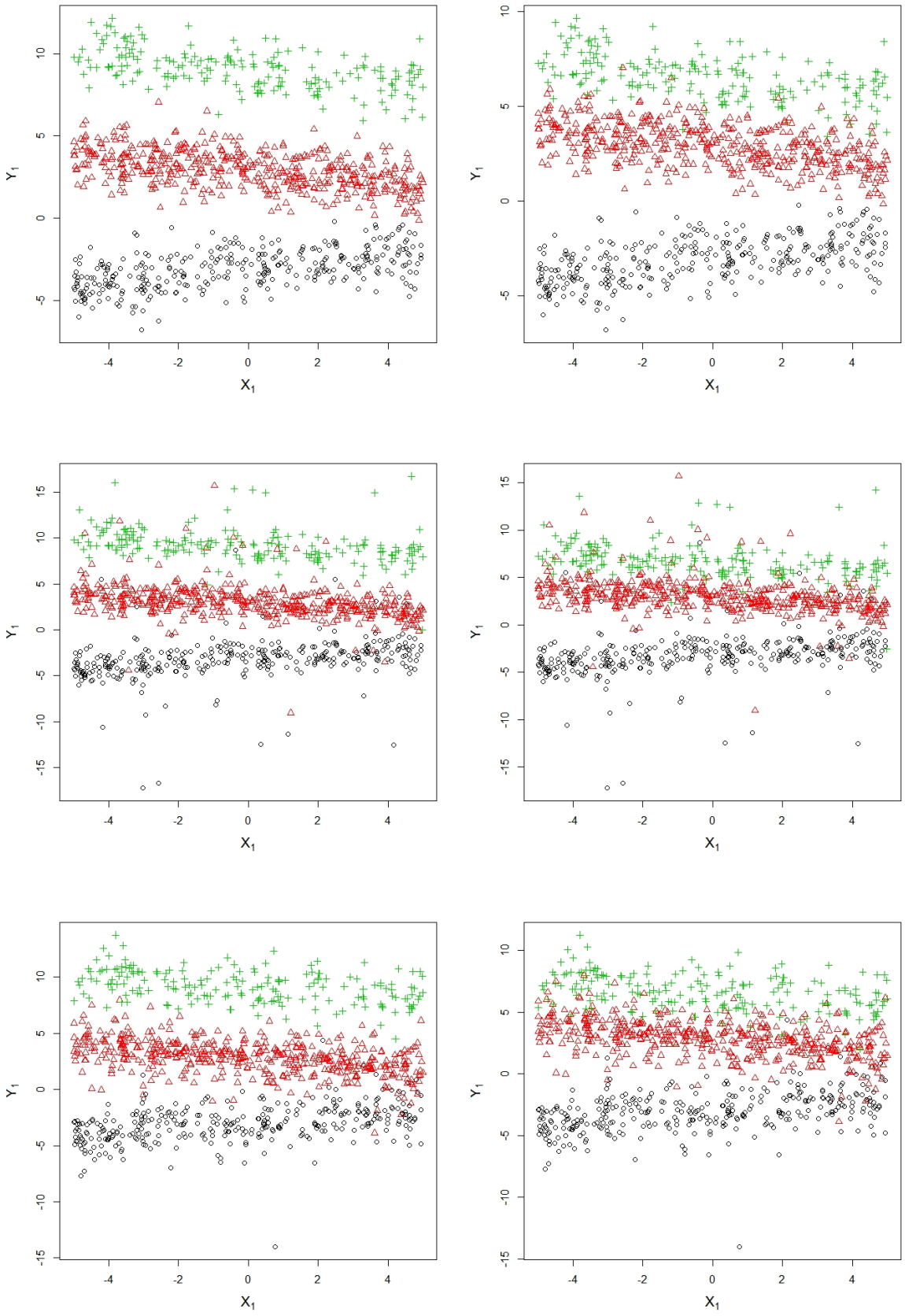

Figure 2.1: Scatterplots of X_1 and Y_1 for samples of size $I = 1000$ generated from the first (upper panel), second (intermediate panel) and third (lower panel) data generation processes under higher ($\epsilon = 9$, left panels) and lower ($\epsilon = 6.5$, right panels) degree of separation. Black circle, red triangle and green plus correspond to $k = 1$, $k = 2$ and $k = 3$, respectively.

(b) MCSG with $\alpha_k = 0.9\ \forall k$, $\eta_1 = 40$, $\eta_2 = \eta_3 = 20$;

(c) mixtures of regression models with seemingly unrelated t-distributed errors (MSt), with $\nu_1 = \nu_2 = \nu_3 = 4$ degrees of freedom.

In all the regression models employed to generate the datasets, the response Y_m has been assumed to depend on X_m, for $m = 1, 2, 3, 4$; thus, $P_m = 1\ \forall m$. With each process, the following parameters have been employed: $\pi_1 = 0.3$, $\pi_2 = 0.5$, $\pi_3 = 0.2$, $\boldsymbol{\beta}_1^* = (-3, 0.2, -3, 0.2, -3, 0.2, -3, 0.2)'$, $\boldsymbol{\beta}_2^* = -\boldsymbol{\beta}_1^*$, $\boldsymbol{\beta}_3^* = (3+\epsilon, -0.2, 3+\epsilon, -0.2, 3+\epsilon, -0.2, 3+\epsilon, -0.2)$,

$$\boldsymbol{\Sigma}_1 = \begin{pmatrix} 1.0 & 0.5 & 0.5 & 0.5 \\ 0.5 & 1.0 & 0.5 & 0.5 \\ 0.5 & 0.5 & 1.0 & 0.5 \\ 0.5 & 0.5 & 0.5 & 1.0 \end{pmatrix}, \boldsymbol{\Sigma}_2 = \boldsymbol{\Sigma}_3 = \begin{pmatrix} 1.00 & 0.75 & 0.75 & 0.75 \\ 0.75 & 1.00 & 0.75 & 0.75 \\ 0.75 & 0.75 & 1.00 & 0.75 \\ 0.75 & 0.75 & 0.75 & 1.00 \end{pmatrix}.$$

It is worth noting that the second and third components only differ in the intercepts of the four regression equations. Covariate values have been generated by a uniform distribution over the interval $(-5, 5)$. As concerns ϵ, two alternatives have been considered in order to produce two different degrees of separation between groups of observations: $\epsilon = 9$ (higher degree), $\epsilon = 6.5$ (lower degree). Figure 2.1 shows the scatterplots of the variables Y_1 and X_1 for a sample of size $I = 1000$ generated using the MSG (upper panel), MCSG (central panel) and MSt (lower panel) processes with $\epsilon = 9$ (on the left) and $\epsilon = 6.5$ (on the right). Due to the values of the regression coefficients employed to model the linear dependencies of Y_m and X_m across the three components, the scatterplots of Y_m and X_m for $m = 2, 3, 4$ are similar. Under each data generating process, 100 random samples of size I have been simulated for each ϵ. As far as the sample size is concerned, the following values have been examined: $I = 500, 1000$. Thus, the degree of separation and the sample size can be considered as experimental factors. This yields a total of 600 generated datasets for each I. The whole analysis has been run on an IBM x3750 M4 server with 4 Intel Xeon E5-4620 processors with 8 cores and 128GB RAM.

2.3.2 Results

A first analysis has been carried out where the MSG, MCG and MCSG models of order $K = 3$ have been fitted to each dataset. It is worth noting that the MCG models have been specified and estimated by assuming that each of the four responses depends on all covariates. Thus, using such models leads to non-parsimonious specifications for all the models that have generated the simulated datasets, as 12 regression coefficients for each component have been estimated

Table 2.1: Estimation of α_k and η_k: averages and standard deviations of the estimates over 100 samples for the fitted MCG and MCSG models of order $K = 3$ ($I = 500$).

	MCG						MCSG					
	$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\alpha}_3$	$\hat{\eta}_2$	$\hat{\eta}_2$	$\hat{\eta}_3$	$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\alpha}_3$	$\hat{\eta}_1$	$\hat{\eta}_2$	$\hat{\eta}_3$
First process - high separation ($\epsilon = 9$)												
average	0.987	0.981	0.983	1.030	1.043	1.017	0.985	0.989	0.989	1.008	1.000	1.000
s.d.	0.027	0.050	0.041	0.268	0.238	0.159	0.049	0.013	0.013	0.077	0.000	0.000
First process - low separation ($\epsilon = 6.5$)												
average	0.967	0.959	0.963	1.060	1.063	1.167	0.980	0.983	0.980	1.033	1.031	1.027
s.d.	0.086	0.100	0.096	0.316	0.219	0.653	0.065	0.038	0.057	0.177	0.156	0.137
Second process - high separation ($\epsilon = 9$)												
average	0.912	0.923	0.901	41.251	18.456	21.903	0.909	0.921	0.899	40.645	17.552	18.886
s.d.	0.040	0.036	0.056	12.373	10.450	15.307	0.043	0.056	0.060	11.610	10.622	8.109
Second process - low separation ($\epsilon = 6.5$)												
average	0.910	0.937	0.895	39.388	16.137	19.439	0.906	0.940	0.892	39.748	14.666	16.705
s.d.	0.038	0.039	0.077	13.692	11.943	16.325	0.049	0.043	0.078	13.112	11.290	9.825
Third process - high separation ($\epsilon = 9$)												
average	0.797	0.746	0.775	7.802	5.331	6.366	0.810	0.742	0.769	9.156	4.386	5.917
s.d.	0.155	0.149	0.161	8.739	6.755	5.281	0.149	0.146	0.171	12.476	2.044	4.457
Third process - low separation ($\epsilon = 6.5$)												
average	0.784	0.761	0.790	7.602	5.672	7.025	0.802	0.751	0.782	7.838	4.444	5.852
s.d.	0.160	0.141	0.171	8.804	10.984	6.709	0.152	0.133	0.155	10.084	1.980	4.283

Table 2.2: Estimation of α_k and η_k: averages and standard deviations of the estimates over 100 samples for the fitted MCG and MCSG models with $K = 3$ ($I = 1000$).

	MCG						MCSG					
	$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\alpha}_3$	$\hat{\eta}_2$	$\hat{\eta}_2$	$\hat{\eta}_3$	$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\alpha}_3$	$\hat{\eta}_1$	$\hat{\eta}_2$	$\hat{\eta}_3$
First process - high separation ($\epsilon = 9$)												
average	0.999	0.999	0.998	1.001	1.001	1.015	0.999	0.999	0.999	1.001	1.001	1.001
s.d.	0.000	0.000	0.012	0.000	0.000	0.145	0.000	0.000	0.000	0.000	0.000	0.000
First process - low separation ($\epsilon = 6.5$)												
average	0.973	0.986	0.983	1.196	1.116	1.155	0.973	0.984	0.988	1.185	1.157	1.090
s.d.	0.080	0.061	0.064	0.444	0.260	0.450	0.079	0.062	0.050	0.394	0.343	0.272
Second process - high separation ($\epsilon = 9$)												
average	0.901	0.909	0.901	41.061	19.267	18.435	0.903	0.912	0.897	40.358	18.882	18.850
s.d.	0.033	0.038	0.042	7.512	6.962	6.380	0.025	0.029	0.045	7.465	6.788	5.862
Second process - low separation ($\epsilon = 6.5$)												
average	0.897	0.917	0.881	37.201	16.182	18.595	0.900	0.924	0.889	37.569	15.547	19.722
s.d.	0.049	0.052	0.092	11.199	9.403	11.045	0.033	0.040	0.071	8.730	9.090	12.401
Third process - high separation ($\epsilon = 9$)												
average	0.860	0.713	0.789	9.317	5.433	6.132	0.849	0.727	0.789	7.846	4.344	6.132
s.d.	0.102	0.138	0.136	16.161	12.798	3.347	0.116	0.134	0.133	9.670	2.569	6.354
Third process - low separation ($\epsilon = 6.5$)												
average	0.812	0.757	0.762	10.637	4.706	10.179	0.812	0.763	0.764	6.883	4.579	6.650
s.d.	0.122	0.124	0.155	33.859	2.597	25.126	0.124	0.113	0.140	8.518	3.895	8.262

Table 2.3: Bias and RMSE for the regression coefficients β_{km} under MSG, MCG and MCSG models of order $K = 3$ in the first process ($I = 500$).

	Bias			RMSE		
	MSG	MCG	MCSG	MSG	MCG	MCSG
High separation						
β_{11}	0.307	0.563	0.312	0.022	0.029	0.022
β_{12}	−0.012	−0.108	−0.013	0.021	0.028	0.021
β_{13}	−0.085	0.047	−0.084	0.024	0.030	0.024
β_{14}	0.145	0.147	0.148	0.023	0.029	0.023
β_{21}	−0.027	0.014	−0.027	0.014	0.021	0.014
β_{22}	−0.119	−0.028	−0.119	0.010	0.024	0.010
β_{23}	0.111	0.205	0.111	0.013	0.022	0.013
β_{24}	−0.256	−0.165	−0.256	0.013	0.023	0.013
β_{31}	−0.112	−0.141	−0.112	0.021	0.038	0.021
β_{32}	0.239	0.439	0.239	0.021	0.036	0.021
β_{33}	−0.257	−0.576	−0.257	0.021	0.036	0.021
β_{34}	0.094	0.060	0.094	0.021	0.034	0.021
Low separation						
β_{11}	0.307	0.571	0.309	0.022	0.029	0.022
β_{12}	−0.012	−0.106	−0.016	0.021	0.028	0.021
β_{13}	−0.085	0.049	−0.089	0.024	0.030	0.024
β_{14}	0.145	0.147	0.153	0.023	0.029	0.023
β_{21}	0.010	0.107	0.005	0.014	0.022	0.014
β_{22}	−0.098	0.153	−0.097	0.010	0.026	0.010
β_{23}	0.107	0.204	0.117	0.013	0.025	0.013
β_{24}	−0.252	−0.047	−0.252	0.014	0.025	0.014
β_{31}	−0.224	−0.034	−0.219	0.021	0.046	0.021
β_{32}	0.195	0.820	0.190	0.023	0.042	0.023
β_{33}	−0.244	−0.512	−0.251	0.022	0.041	0.022
β_{34}	0.094	0.166	0.092	0.021	0.040	0.021

Biases have been multiplied by 100 to facilitate presentation.

Table 2.4: Bias and RMSE for the regression coefficients β_{km} under MSG, MCG and MCSG models of order $K = 3$ in the second process ($I = 500$).

	Bias			RMSE		
	MSG	MCG	MCSG	MSG	MCG	MCSG
High separation						
β_{11}	6.732	0.515	0.288	0.103	0.029	0.023
β_{12}	6.819	−0.054	−0.008	0.105	0.029	0.023
β_{13}	6.728	0.062	−0.083	0.105	0.029	0.023
β_{14}	6.816	0.297	0.246	0.104	0.031	0.025
β_{21}	−0.833	0.104	0.049	0.061	0.022	0.015
β_{22}	−0.983	−0.058	−0.113	0.057	0.025	0.012
β_{23}	−0.852	0.131	0.086	0.064	0.023	0.014
β_{24}	−1.165	−0.195	−0.288	0.060	0.025	0.014
β_{31}	−1.220	−0.260	−0.272	0.044	0.041	0.023
β_{32}	−0.441	0.295	0.270	0.034	0.036	0.021
β_{33}	−0.917	−0.593	−0.241	0.041	0.039	0.021
β_{34}	−0.248	0.261	0.184	0.034	0.036	0.021
Low separation						
β_{11}	7.440	0.900	0.306	0.118	0.052	0.022
β_{12}	7.583	0.331	0.025	0.118	0.046	0.023
β_{13}	7.517	0.418	−0.104	0.118	0.045	0.023
β_{14}	7.421	0.527	0.189	0.117	0.050	0.025
β_{21}	−1.508	0.368	0.030	0.074	0.024	0.014
β_{22}	−1.791	0.140	−0.070	0.079	0.025	0.012
β_{23}	−1.611	−0.008	0.123	0.081	0.026	0.014
β_{24}	−1.890	0.010	−0.266	0.079	0.026	0.013
β_{31}	−3.674	−0.764	−0.089	0.129	0.137	0.034
β_{32}	−3.169	−3.185	0.174	0.101	0.200	0.052
β_{33}	−3.644	−1.903	−0.536	0.145	0.177	0.077
β_{34}	−2.049	−1.250	0.325	0.101	0.201	0.044

Biases have been multiplied by 100 to facilitate presentation.

Table 2.5: Bias and RMSE for the regression coefficients β_{km} under MSG, MCG and MCSG models of order $K = 3$ in the third process ($I = 500$).

	Bias			RMSE		
	MSG	MCG	MCSG	MSG	MCG	MCSG
High separation						
β_{11}	0.786	0.090	0.296	0.034	0.034	0.029
β_{12}	0.861	0.224	0.411	0.035	0.043	0.029
β_{13}	0.674	0.300	0.254	0.033	0.041	0.030
β_{14}	0.532	0.108	−0.157	0.035	0.043	0.027
β_{21}	0.145	−0.014	0.055	0.018	0.037	0.016
β_{22}	0.109	−0.810	−0.003	0.017	0.045	0.014
β_{23}	−0.082	−0.211	−0.152	0.020	0.041	0.018
β_{24}	0.162	−0.023	0.027	0.015	0.032	0.014
β_{31}	−0.206	−1.520	−0.273	0.029	0.056	0.027
β_{32}	−0.384	−0.092	−0.319	0.031	0.061	0.027
β_{33}	0.784	0.293	0.425	0.027	0.063	0.026
β_{34}	0.060	0.326	0.384	0.026	0.049	0.025
Low separation						
β_{11}	0.312	−0.218	0.101	0.032	0.032	0.026
β_{12}	0.411	0.024	0.264	0.029	0.035	0.028
β_{13}	0.354	0.011	0.182	0.033	0.035	0.029
β_{14}	−0.019	−0.297	−0.246	0.029	0.034	0.026
β_{21}	0.026	0.124	0.048	0.017	0.038	0.017
β_{22}	−0.117	−0.536	0.155	0.018	0.039	0.016
β_{23}	0.105	0.232	−0.108	0.022	0.043	0.018
β_{24}	0.371	−0.038	0.156	0.017	0.038	0.016
β_{31}	−0.336	−3.023	0.052	0.056	0.138	0.034
β_{32}	0.334	−2.051	−1.141	0.057	0.166	0.066
β_{33}	1.120	0.634	−1.330	0.169	0.110	0.128
β_{34}	−0.296	−1.419	−0.377	0.059	0.151	0.047

Biases have been multiplied by 100 to facilitate presentation.

although in fact they are equal to zero. The average execution times (over the 100 datasets with $I = 500$) for the MCSG models have ranged between 2.499 and 55.020 seconds, depending on the process and the specific value of ϵ employed to generate the datasets. Concerning the other two models, the minimum and maximum average execution times have resulted to be equal to 1.722 and 24.580 seconds with MSG models, 7.765 and 58.520 seconds with MCG models. It is worth noting that, since the implementation of the ECM algorithm has not been carried out with the goal of being efficient from a computational point of view, these CPU times should be regarded as merely illustrative and can be reduced using more efficient implementations. In the first analysis, the performances of the three competing models have been evaluated with respect to the following aspects: textit(i) the estimation of the proportions of typical observations and the degrees of contamination (proper estimation of α_k and η_k); textit(ii) the ability to recover the true values of the unknown parameters (parameter recovery); *(iii)* the ability to recover the true partition of the sample observations (classification recovery). When evaluating properties of the parameter estimators using simulation studies under mixture models, there may be label switching issues. Several labeling methods have been proposed. For the models examined here, as in Bai et al. (2012), Yao (2014) and Mazza and Punzo (2020), labels have been chosen by minimising the Euclidean distance to the true parameter values.

A second analysis has been carried out so as to obtain an evaluation of the three approaches without exploiting the knowledge of the true number of components. Thus, in addition to the models already examined in the first analysis, also models of order $K = 1, 2, 4, 5$ have been fitted to each dataset. All the obtained results have been employed to collect information on the following aspects: *(iv)* the capability to reach the best trade-off between the fit and model complexity; *(v)* the ability of BIC, ICL_1 and ICL_2 to detect the true value of K (comparison among information criteria).

Estimation of α_k and η_k

The aspect *(i)* has been studied for the fitted MCG and MCSG models with $K = 3$. Under the first two data generation processes, the averages of the estimated proportions of good points ($\hat{\alpha}_k$) and the estimated inflation parameters ($\hat{\eta}_k$) are close to their true values under both MCG and MCSG models, regardless of the level of separation and the sample size (see the upper part of Tables 2.1 and 2.2). However, it is worth noting that slightly lower standard deviations of such estimates have been registered under the first process, thus giving an indication of a

Table 2.6: Bias and RMSE for the regression coefficients β_{km} under MSG, MCG and MCSG models of order $K = 3$ in the first process ($I = 1000$).

	Bias			RMSE		
	MSG	MCG	MCSG	MSG	MCG	MCSG
High separation						
β_{11}	0.162	0.128	0.162	0.016	0.020	0.016
β_{12}	−0.066	0.009	−0.066	0.017	0.022	0.017
β_{13}	0.127	0.478	0.127	0.015	0.020	0.015
β_{14}	0.070	0.084	0.070	0.017	0.020	0.017
β_{21}	−0.126	−0.314	−0.126	0.008	0.014	0.008
β_{22}	−0.042	−0.080	−0.042	0.008	0.015	0.008
β_{23}	0.081	0.077	0.081	0.010	0.016	0.010
β_{24}	−0.057	0.080	−0.057	0.008	0.014	0.008
β_{31}	0.075	0.161	0.075	0.014	0.025	0.014
β_{32}	−0.153	−0.073	−0.153	0.015	0.026	0.015
β_{33}	0.091	0.158	0.091	0.015	0.024	0.015
β_{34}	−0.124	−0.452	−0.124	0.014	0.025	0.014
Low separation						
β_{11}	0.159	0.122	0.161	0.016	0.020	0.016
β_{12}	−0.065	0.012	−0.060	0.017	0.022	0.017
β_{13}	0.129	0.474	0.127	0.015	0.020	0.015
β_{14}	0.070	0.077	0.073	0.017	0.020	0.017
β_{21}	−0.008	0.276	−0.008	0.009	0.015	0.009
β_{22}	−0.008	−0.045	−0.007	0.009	0.016	0.009
β_{23}	0.059	−0.071	0.056	0.010	0.016	0.010
β_{24}	0.028	−0.149	0.031	0.008	0.016	0.008
β_{31}	−0.034	−0.027	−0.032	0.014	0.028	0.014
β_{32}	−0.067	−0.248	−0.067	0.014	0.031	0.014
β_{33}	0.069	−0.238	0.070	0.016	0.031	0.016
β_{34}	−0.031	0.013	−0.030	0.015	0.031	0.015

Biases have been multiplied by 100 to facilitate presentation.

Table 2.7: Bias and RMSE for the regression coefficients β_{km} under MSG, MCG and MCSG models of order $K = 3$ in the second process ($I = 1000$).

	Bias			RMSE		
	MSG	MCG	MCSG	MSG	MCG	MCSG
High separation						
β_{11}	6.928	0.092	0.217	0.086	0.020	0.015
β_{12}	7.415	0.161	0.116	0.094	0.385	0.019
β_{13}	6.835	−0.304	−0.269	0.088	0.390	0.015
β_{14}	6.101	−0.221	−0.219	0.081	0.021	0.018
β_{21}	−0.277	−0.140	−0.102	0.033	0.016	0.010
β_{22}	−0.100	0.045	0.077	0.031	0.015	0.011
β_{23}	−0.246	−0.003	−0.055	0.033	0.386	0.010
β_{24}	−0.276	−0.178	−0.103	0.034	0.016	0.009
β_{31}	−0.906	−0.264	−0.185	0.030	0.027	0.015
β_{32}	−0.218	0.214	−0.036	0.026	0.389	0.015
β_{33}	−0.916	−0.396	−0.233	0.031	0.029	0.016
β_{34}	−0.502	−0.099	−0.157	0.027	0.026	0.014
Low separation						
β_{11}	6.911	−0.051	0.147	0.092	0.020	0.015
β_{12}	7.924	0.014	0.299	0.105	0.023	0.019
β_{13}	7.733	0.175	−0.075	0.101	0.018	0.014
β_{14}	6.543	−0.234	−0.239	0.090	0.023	0.017
β_{21}	−0.713	0.223	−0.126	0.049	0.018	0.010
β_{22}	−0.354	0.219	0.198	0.048	0.019	0.010
β_{23}	−0.668	−0.148	−0.084	0.050	0.018	0.009
β_{24}	−0.286	0.143	0.252	0.044	0.016	0.009
β_{31}	−2.667	0.019	−0.876	0.081	0.085	0.116
β_{32}	−1.447	−0.236	−0.033	0.072	0.080	0.068
β_{33}	−2.959	0.591	0.184	0.092	0.087	0.035
β_{34}	−2.173	0.732	−1.039	0.081	0.111	0.091

Biases have been multiplied by 100 to facilitate presentation.

Table 2.8: Bias and RMSE for the regression coefficients β_{km} under MSG, MCG and MCSG models of order $K = 3$ in the third process ($I = 1000$).

	Bias			RMSE		
	MSG	MCG	MCSG	MSG	MCG	MCSG
High separation						
β_{11}	0.325	−0.128	−0.022	0.022	0.027	0.019
β_{12}	0.412	0.057	−0.011	0.024	0.026	0.022
β_{13}	0.686	0.268	0.160	0.022	0.025	0.019
β_{14}	0.326	−0.049	0.091	0.027	0.028	0.024
β_{21}	0.006	−0.199	−0.027	0.011	0.020	0.011
β_{22}	0.217	0.330	0.035	0.012	0.020	0.011
β_{23}	−0.011	−0.280	−0.131	0.012	0.019	0.011
β_{24}	−0.324	−0.406	−0.233	0.013	0.018	0.012
β_{31}	−0.049	0.125	−0.083	0.021	0.033	0.019
β_{32}	0.118	0.154	0.003	0.018	0.032	0.017
β_{33}	−0.170	0.052	−0.190	0.020	0.036	0.018
β_{34}	−0.271	−0.516	−0.251	0.020	0.033	0.018
Low separation						
β_{11}	0.197	0.035	0.052	0.022	0.028	0.018
β_{12}	−0.075	−0.289	−0.160	0.021	0.038	0.019
β_{13}	0.540	0.430	0.407	0.023	0.028	0.020
β_{14}	0.257	0.081	0.130	0.019	0.027	0.018
β_{21}	0.084	0.140	0.063	0.013	0.023	0.012
β_{22}	0.137	−0.142	−0.049	0.013	0.026	0.011
β_{23}	0.140	0.279	0.213	0.014	0.021	0.012
β_{24}	−0.143	−0.130	−0.117	0.012	0.024	0.012
β_{31}	−0.911	−1.273	0.050	0.057	0.104	0.019
β_{32}	−1.822	−2.135	0.061	0.085	0.162	0.021
β_{33}	−1.087	−1.037	0.041	0.077	0.107	0.021
β_{34}	−0.408	−0.881	0.156	0.069	0.083	0.022

Biases have been multiplied by 100 to facilitate presentation.

higher stability of the obtained estimates; furthermore, the estimation of η_1, η_2 and η_3 under the second process appears to be characterised by a certain instability, which results to reduce as the sample size I increases using both MCG and MCSG models. As far as the results from the analyses of the datasets generated using the third process are concerned (lower part of Tables 2.1 and 2.2), the estimated values of α_k and η_k, $k = 1, 2, 3$, are far from 1, regardless of the values of ϵ and I. Thus, the departure from a four-dimensional Gaussian distribution for the errors of the regression model has been detected within each of the three mixture components of both MCG and MCSG models for both sample sizes. The standard deviations of $\hat{\eta}_k$, $k = 1, 2, 3$ are high, and this result holds true particularly with MCG models and $I = 1000$.

Parameter recovery

The evaluation of the aspect *(ii)* has been focused on the regression coefficients β_{km} and has been carried out by computing the following quantities:

$$\begin{aligned}
\text{Bias}\left(\hat{\beta}_{km}\right) &= \frac{\sum_{r=1}^{100} \hat{\beta}_{km}^{(r)}}{100} - \beta_{km}, \ k = 1, 2, 3, \ m = 1, 2, 3, 4, \\
\text{RMSE}\left(\hat{\beta}_{km}\right) &= \sqrt{\frac{\sum_{r=1}^{100} \left(\beta_{km} - \hat{\beta}_{km}^{(r)}\right)^2}{100}}, \ k = 1, 2, 3, \ m = 1, 2, 3, 4,
\end{aligned}$$

where $\hat{\beta}_{km}^{(r)}$ is the ML estimate of β_{km} obtained from the rth dataset ($r = 1, \ldots, 100$) using models of order $K = 3$. With $I = 500$ and under the first data generating process (Table 2.3), MSG and MCSG models show the same performance in terms of recovering the true values of the regression coefficients with both degrees of separation. The good performance of MCSG models is consistent with the proper estimation of α_k and η_k associated with these models under the first process (see the previous aspect). On the contrary, the inclusion of irrelevant predictors in the four regression equations (MCG models) leads to a slight increase in the RMSEs. With contaminated datasets of size $I = 500$, as expected, the lowest (absolute) biases and RMSEs are obtained using the MCSG model (see Table 2.4); there also seems to be a tendency for MCG models to perform slightly better than MSG models for the majority of the regression coefficients. When the datasets are generated with $I = 500$ and according to the third process, the highest accuracy in the estimation of the regression coefficients is obtained using MCSG models (see Table 2.5). It is also worth noting that, in spite of their ability to detect a departure from the Gaussian distribution within each component, MCG models show the lowest accuracy. Similar results have been obtained with $I = 1000$ (see Tables 2.6-2.8).

Table 2.9: Classification recovery of the fitted MSG, MCG and MCSG models of order $K = 3$: average values (standard deviations) of the ARI index over 100 samples ($I = 500$).

Process	ϵ	MSG	MCG	MCSG
I	9	0.999 (0.003)	0.999 (0.003)	0.999 (0.003)
I	6.5	0.946 (0.018)	0.937 (0.028)	0.946 (0.018)
II	9	0.818 (0.024)	0.911 (0.027)	0.910 (0.031)
II	6.5	0.723 (0.094)	0.806 (0.100)	0.821 (0.087)
III	9	0.931 (0.033)	0.936 (0.037)	0.937 (0.040)
III	6.5	0.721 (0.147)	0.745 (0.145)	0.776 (0.129)

Classification recovery

To obtain information on the aspect *(iii)*, the partitions of the sample units associated with the models of order $K = 3$ under each competing model class have been compared with the true partition; the agreement with this latter partition has been measured by resorting to the adjusted Rand index (ARI) (Hubert and Arabie, 1985). When the datasets are generated using the first process and the highest level of separation (see the upper part of Tables 2.9 and 2.10), an almost perfect classification recovery ($ARI = 0.999$) is obtained by each of the three models regardless of the sample size. When the level of separation is low ($\epsilon = 6.5$), a slight decrease in the ability to recover the true partition of the sample observations is registered for all models and, in particular, for the MCG ones when $I = 500$ ($ARI = 0.937$). When there are outliers in the data and $\epsilon = 9$, the best performance is obtained using either MCG models or MCSG models with both sample sizes ($ARI = 0.91$); these latter models slightly outperform MCG models when $\epsilon = 6.5$. As far as MSG models are concerned, due to their inability to manage the presence of mild outliers in the data, the classification recovery appears to be markedly lower, especially with the lowest level of separation ($ARI = 0.723$ with $I = 500$, $ARI = 0.716$ with $I = 1000$). Under the third process and the highest level of separation, good performances are obtained by all models with both sample sizes ($ARI > 0.93$). When the level of separation is reduced, a general decrease in the capability to reconstruct the true partition is registered; MCSG models appear to be less affected by this tendency, regardless of the sample size.

Trade-off between fit and complexity

In order to study the aspect *(iv)*, for each dataset and each model class, the models of order $\hat{K}_{IC}$ have been selected, where IC denotes an information criterion ($IC \in \{BIC, ICL_1, ICL_2\}$) and $\hat{K}_{IC} = \arg\max IC(K)$ for $K \in \{1, 2, 3, 4, 5\}$. Then, the average values of the 100 resulting values

Table 2.10: Classification recovery of the fitted MSG, MCG and MCSG models of order $K = 3$: average values (standard deviations) of the ARI index over 100 samples ($I = 1000$).

Process	ϵ	MSG	MCG	MCSG
I	9	0.999 (0.002)	0.999 (0.002)	0.999 (0.002)
I	6.5	0.951 (0.011)	0.949 (0.012)	0.951 (0.011)
II	9	0.803 (0.015)	0.914 (0.023)	0.916 (0.021)
II	6.5	0.716 (0.088)	0.823 (0.092)	0.831 (0.082)
III	9	0.941 (0.016)	0.943 (0.013)	0.944 (0.014)
III	6.5	0.706 (0.147)	0.814 (0.095)	0.814 (0.102)

of $BIC(\hat{K}_{BIC})$, $ICL_1(\hat{K}_{ICL_1})$ and $ICL_2(\hat{K}_{ICL_2})$ have been computed within the three model classes. As expected, when datasets of $I = 500$ observations are generated without outliers (first process), the best trade-off between the fit and model complexity is reached by MSG models, regardless of the level of separation and the criterion employed to select the best model (see the upper part of Table 2.11). With these datasets, MCSG models slightly outperform MCG models. When there are outliers in the data (second process) or the error terms of the K regression models have tails heavier than the Gaussian ones (third process), MCSG shows the best performance in terms of capability to reach the best trade-off between fit and complexity, regardless of the level of separation and the criterion employed to select the best model (see the lower part of Table 2.11). Interestingly, when the outliers are generated using a MCSG model (second process), MSG models slightly outperform MCG models, regardless of the value of ϵ. Similar conclusions can be drawn also from the results obtained when $I = 1000$ (see Table 2.12).

Comparison among information criteria

As far as the aspect *(v)* is concerned, the attention has been focused on the number of times each value of K has been selected by each examined criterion. With datasets generated using the first process and the highest level of separation, all the examined information criteria always recognize the presence of three clusters, regardless of the fitted model and the sample size (see the upper part of Tables 2.13 and 2.14). If the level of separation is reduced ($\epsilon = 6.5$), the BIC still tends to correctly identify the presence of three clusters regardless of the fitted model only with the largest sample size. If $I = 500$, the same tendency is slightly weaker with MSG and MCSG models; the order of the models employed to generate the datasets is always underestimated by the BIC when MCG models are employed. ICL_1 and ICL_2 show a clear preference for $K = 3$ components only when models embedding the information on the relevant regressors (e.g., MSG

Table 2.11: Average values of $BIC(\hat{K}_{BIC})$, $ICL_1(\hat{K}_{ICL_1})$ and $ICL_2(\hat{K}_{ICL_2})$ over 100 samples ($I = 500$).

$BIC(\hat{K}_{BIC})$			$ICL_1(\hat{K}_{ICL_1})$			$ICL_2(\hat{K}_{ICL_2})$		
MSG	MCG	MCSG	MSG	MCG	MCSG	MSG	MCG	MCSG
First process - high separation ($\epsilon = 9$)								
-5776	-6002	-5807	-5776	-6003	-5808	-5777	-6003	-5809
First process - low separation ($\epsilon = 6.5$)								
-5731	-5894	-5753	-5740	-5895	-5759	-5748	-5898	-5764
Second process - high separation ($\epsilon = 9$)								
-6650	-6756	-6558	-6657	-6776	-6577	-6674	-6802	-6603
Second process - low separation ($\epsilon = 6.5$)								
-6601	-6667	-6508	-6621	-6682	-6531	-6655	-6702	-6555
Third process - high separation ($\epsilon = 9$)								
-6979	-7065	-6886	-6995	-7076	-6898	-7012	-7096	-6919
Third process - low separation ($\epsilon = 6.5$)								
-6866	-6895	-6775	-6892	-6906	-6787	-6906	-6925	-6806

Table 2.12: Average values of $BIC(\hat{K}_{BIC})$, $ICL_1(\hat{K}_{ICL_1})$ and $ICL_2(\hat{K}_{ICL_2})$ over 100 samples ($I = 1000$).

$BIC(\hat{K}_{BIC})$			$ICL_1(\hat{K}_{ICL_1})$			$ICL_2(\hat{K}_{ICL_2})$		
MSG	MCG	MCSG	MSG	MCG	MCSG	MSG	MCG	MCSG
First process - high separation ($\epsilon = 9$)								
−11298	−11552	−11339	−11299	−11553	−11340	−11300	−11554	−11341
First process - low separation ($\epsilon = 6.5$)								
−11217	−11469	−11257	−11253	−11492	−11293	−11296	−11507	−11334
Second process - high separation ($\epsilon = 9$)								
−13116	−13202	−13000	−13131	−13251	−13049	−13167	−13313	−13111
Second process - low separation ($\epsilon = 6.5$)								
−12989	−13107	−12923	−13039	−13159	−13002	−13119	−13209	−13070
Third process - high separation ($\epsilon = 9$)								
−13699	−13760	−13541	−13773	−13786	−13568	−13833	−13829	−13611
Third process - low separation ($\epsilon = 6.5$)								
−13495	−13510	−13346	−13611	−13536	−13401	−13681	−13575	−13444

Table 2.13: Comparison among information criteria: number of selections over 100 samples for MSG, MCG and MCSG models of order $K \in \{1, 2, 3, 4, 5\}$ ($I = 500$).

K	$BIC(K)$			$ICL_1(K)$			$ICL_2(K)$		
	MSG	MCG	MCSG	MSG	MCG	MCSG	MSG	MCG	MCSG
First process - high separation ($\epsilon = 9$)									
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	100	100	100	100	100	100	100	100	100
4	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0
First process - low separation ($\epsilon = 6.5$)									
1	0	0	0	0	0	0	0	0	0
2	25	100	51	52	100	72	76	100	85
3	75	0	49	48	0	28	24	0	15
4	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0
Second process - high separation ($\epsilon = 9$)									
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	0	100	98	0	100	98	0	100	99
4	99	0	2	99	0	2	99	0	1
5	1	0	0	1	0	0	1	0	0
Second process - low separation ($\epsilon = 6.5$)									
1	0	0	0	0	0	0	0	0	0
2	0	99	50	0	99	75	0	100	94
3	11	1	50	15	1	25	19	0	6
4	89	0	0	85	0	0	81	0	0
5	0	0	0	0	0	0	0	0	0
Third process - high separation ($\epsilon = 9$)									
1	0	0	0	0	0	0	0	0	0
2	0	2	0	0	2	1	0	2	1
3	52	98	99	70	98	98	77	98	96
4	39	0	1	25	0	1	22	0	3
5	9	0	0	5	0	0	1	0	0
Third process - low separation ($\epsilon = 6.5$)									
1	0	0	0	0	0	0	0	0	0
2	40	100	89	82	100	100	93	100	100
3	24	0	11	7	0	0	4	0	0
4	27	0	0	10	0	0	3	0	0
5	9	0	0	1	0	0	0	0	0

Table 2.14: Comparison among information criteria: number of selections over 100 samples for MSG, MCG and MCSG models of order $K \in \{1, 2, 3, 4, 5\}$ ($I = 1000$).

K	$BIC(K)$			$ICL_1(K)$			$ICL_2(K)$		
	MSG	MCG	MCSG	MSG	MCG	MCSG	MSG	MCG	MCSG
First process - high separation ($\epsilon = 9$)									
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	100	100	100	100	100	100	100	100	100
4	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0
First process - low separation ($\epsilon = 6.5$)									
1	0	0	0	0	0	0	0	0	0
2	0	13	0	0	49	0	17	84	24
3	100	87	100	100	51	100	83	16	76
4	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0
Second process - high separation ($\epsilon = 9$)									
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	0	99	100	0	99	100	0	100	100
4	100	1	0	100	1	0	100	0	0
5	0	0	0	0	0	0	0	0	0
Second process - low separation ($\epsilon = 6.5$)									
1	0	0	0	0	0	0	0	0	0
2	0	19	4	0	80	17	0	93	68
3	0	81	91	1	20	81	8	7	31
4	100	0	5	99	0	2	92	0	1
5	0	0	0	0	0	0	0	0	0
Third process - high separation ($\epsilon = 9$)									
1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0
3	14	100	98	48	100	99	69	100	99
4	69	0	2	49	0	1	31	0	1
5	17	0	0	3	0	0	0	0	0
Third process - low separation ($\epsilon = 6.5$)									
1	0	0	0	0	0	0	0	0	0
2	1	88	12	44	100	88	81	100	100
3	19	12	87	13	0	12	10	0	0
4	67	0	1	40	0	0	9	0	0
5	13	0	0	3	0	0	0	0	0

and MCSG) are employed and the sample size is $I = 1000$. Otherwise, they generally underestimate the true number of clusters. Under the second process, when MSG models are fitted to the data, all the examined information criteria show a clear tendency to select $K = 4$ components an additional component accommodating outliers is typically selected), regardless of the level of separation and the sample size (see also Mazza and Punzo, 2020). On the contrary, with both MCG and MCSG models, the three criteria almost always correctly identify three components, regardless of the sample size, provided that the degree of separation is high. When $\epsilon = 6.5$, the same result is obtained by the BIC in association with MCG and MCSG models and by ICL_1 in association with MCSG models only with the largest sample size; otherwise, due to both a low separation between two clusters and a low sample size, the examined criteria generally underestimate the true value of K. This behaviour is particularly evident when the selection of K is based on ICL_2. A possible explanation for this is that the penalty employed by ICL_2 (a function of the uncertainty of the estimated posterior probabilities $\hat{z}_{ik}$) is the most severe and is also expected to be particularly large whenever the analysed dataset contains true clusters which are not well separated. When the datasets are generated using the third process and the smallest sample size, the obtained results show that, if $\epsilon = 9$, the three criteria generally detect the true value of K (see the lower part of Table 2.13). This tendency appears to be stronger when MCG and MCSG models are employed. These results hold true also with $I = 1000$ except when MSG models are fitted to the data and K is selected using either the BIC or the ICL_1; in these latter situations the true K is overestimated. On the contrary, when the degree of separation is low, models of order $K = 2$ are generally selected from each examined model class according to ICL_1 and ICL_2, regardless of the sample size. Also this result could be due to the role played by the penalties employed by these two latter criteria in the presence of true clusters which are not well separated. As far as the BIC is concerned, it allows to detect the true number of components only when MCSG models are fitted to samples of size $I = 1000$. It also shows a tendency to underestimate the true K both with MCSG models fitted to smaller samples and with MCG models regardless of the sample size. Finally, a slight preference with MSG models of order $K = 2$ and $K = 4$ emerges in association with samples of size $I = 500$ and $I = 1000$, respectively.

2.4 Results from the analysis of canned tuna sales

The practical usefulness and effectiveness of the proposed models have been evaluated through the analysis of a dataset containing the volume of weekly sales (`Move`) for seven of the top 10 U.S. brands in the canned tuna product category for $I = 338$ weeks between September 1989 and May 1997 (Chevalier et al., 2003). Measures of the display activity (`Nsale`) and the log price (`Lprice`) of each brand in each week are also available. This dataset is included in the `R` package `bayesm` (Rossi, 2012). The analysis here considers two products: Star Kist 6 oz. (SK) and Bumble Bee Solid 6.12 oz. (BBS). In order to study the dependence of canned tuna sales on prices and promotional activites for these two brands, the analysis has been carried out starting from the following vectors of variables: $\mathbf{Y} = (Y_1 = \texttt{Lmove SK}, Y_2 = \texttt{Lmove BBS})$, $\mathbf{X} = (X_1 = \texttt{Nsale SK}, X_2 = \texttt{Lprice SK}, X_3 = \texttt{Nsale BBS}, X_4 = \texttt{Lprice BBS})$, where `Lmove` denotes the logarithm of `Move`; thus, $M = 2$ and $P = 4$. Previous studies focused on other brands are illustrated in Galimberti et al. (2016) and Galimberti and Soffritti (2020).

The analysis has been carried out through MSG, MCG and MCSG models. The additional class comprising mixtures of linear Gaussian regression models (Jones and McLachlan, 1992) has been included in the comparison; the notation employed for this model class is MRM. Models from each of these four classes have been estimated for $K \in \{1, 2, 3, 4\}$. Furthermore, since prices and promotional activities for one product could have an impact on the sales of the other product, models from MSG and MCSG classes have been specified and fitted by considering all possible sub-vectors of $\mathbf{X}$ as vectors $\mathbf{X}_m$, $m = 1, 2$, for each K. Thus, the analysis has also included an exhaustive search of the relevant regressors for both `Lmove SK` and `Lmove BBS`. For each K, $2^{P \cdot M} = 256$ different mixtures of regression models have been estimated either with contamination or without contamination; the overall number of estimated models is 2048. It is worth noting that none of the models employed in this analysis explicitly accounts for serial dependencies that may characterise this dataset.

Figure 2.2 shows the values of BIC, ICL_1 and ICL_2 for the fitted MCSG, MSG, MCG and MRM models which maximise each of these model selection criteria by K. An analysis based on a single linear regression model without contamination (MSG and MRM models with $K = 1$) is clearly inadequate according to all criteria. The best trade-off among the fit, the model complexity and the uncertainty of the estimated partition of the weeks is reached by models of order $K = 2$ for each of the four examined model classes. If model selection is only

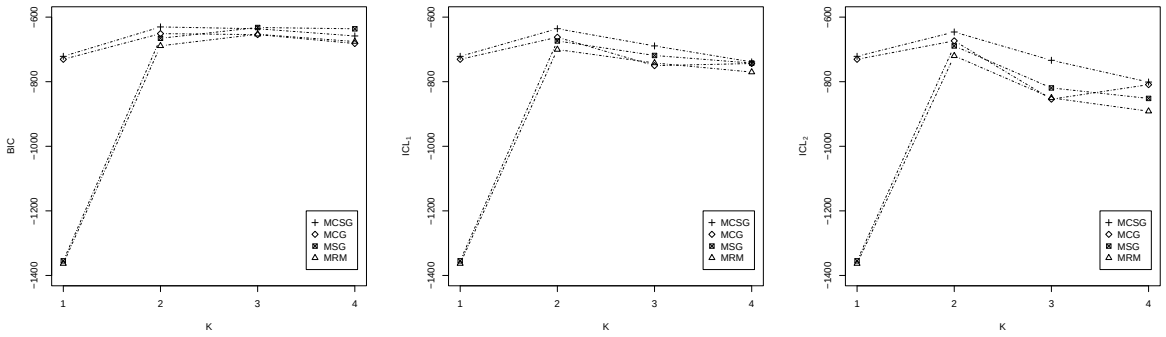

Figure 2.2: Values of BIC, ICL_1 and ICL_2 for the best MCG, MCSG, MSG and MRM models by number of components in the analysis of tuna sales.

Table 2.15: Maximised log-likelihood and values of BIC, ICL_1 and ICL_2 for six models selected from the classes MCSG, MCG, MSG and MRM in the analysis of tuna sales.

Class	K	$\mathbf{X}_1$	$\mathbf{X}_2$	$\ell(\hat{\boldsymbol{\psi}})$	n_ψ	BIC	ICL_1	ICL_2
MCSG	2	X_1, X_2	X_2, X_3, X_4	-242.5	25	-630.5	-636.0	-646.1
MCG	2	X_2, X_3, X_4	X_2, X_3, X_4	-247.0	27	-651.1	-662.3	-673.5
MSG	2	X_1, X_2	X_3, X_4	-277.5	19	-665.6	-673.8	-689.2
MRM	2	X_2, X_4	X_2, X_4	-289.2	19	-688.9	-700.5	-719.9
MSG	3	X_2	X_3, X_4	-240.4	26	-632.2	-737.4	-865.7
MRM	3	X_2, X_3, X_4	X_2, X_3, X_4	-224.6	35	-653.0	-750.0	-877.9

based on the fit and the model complexity, the best MCSG and MCG models still have $K = 2$ components, while MSG and MRM models of order $K = 3$ should be preferred.

Table 2.15 reports more detailed information about the six models which best fit the analysed dataset according to the three model selection criteria over the five examined values of K within each model class. All the examined criteria select a seemingly unrelated contaminated Gaussian linear clusterwise regression model of order $K = 2$ as the overall best model for studying the effect of prices and promotional activities on sales for the two brands. In this model, the log unit sales of SK canned tuna are regressed on the log prices and the promotional activities of the same brand; as far as the regressors for the BBS log unit sales are concerned, the selected regressors are the log prices of both brands and the promotional activities of BBS. From the parameter estimates (see Table 2.16) it emerges that the analysed dataset is characterised both by heterogeneity over time and by the presence of atypical observations. This latter feature seems to characterise the two clusters of weeks detected by the model almost in the same way (the estimated weights of the typical observations are $\hat{\alpha}_1 = 0.827$ and $\hat{\alpha}_2 = 0.829$); however, the strength of the contaminating effect on the conditional variances and covariances of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ results to be stronger in the first cluster, where the estimated inflation parameter for the elements

Table 2.16: Parameter estimates of the overall best model for the analysis of tuna sales.

$\hat{\boldsymbol{\psi}}$	$k=1$	$k=2$
$\hat{\pi}_k$	0.062	0.938
$\hat{\alpha}_k$	0.827	0.829
$\hat{\eta}_k$	13.44	6.80
$\hat{\boldsymbol{\beta}}_{k1}^{\prime *}$	$(8.86, 0.59, -4.68)$	$(8.65, 0.27, -3.11)$
$\hat{\boldsymbol{\beta}}_{k2}^{\prime *}$	$(15.09, 3.91, 2.77, -17.84)$	$(9.98, 0.25, 0.12, -3.82)$
$\hat{\boldsymbol{\Sigma}}_k$	$\begin{pmatrix} 0.043 & -0.022 \\ -0.022 & 0.126 \end{pmatrix}$	$\begin{pmatrix} 0.118 & 0.011 \\ 0.011 & 0.028 \end{pmatrix}$

of $\boldsymbol{\Sigma}_1$ is larger ($\hat{\eta}_1 = 13.44$). Heterogeneity over time appears to emerge both in some effects of the selected regressors and in the conditional expected variances and covariances of log sales for the typical observations. From the estimates of the regression equation for `Lmove SK` it emerges that sales of SK canned tuna are negatively affected by prices and positively affected by promotional activities of the same brand within both clusters detected by the model. However, the estimated effects of these two variables in the first cluster result to be stronger than those in the second cluster. Similar results have been obtained with reference to the regression equation for `Lmove BBS`, from which it also emerges that the log prices of SK canned tuna positively affect the log unit sales of the other brand, especially in the first cluster of weeks. As far as the estimated conditional variances and covariances are concerned, typical weeks in the first cluster appear to be characterised by values of `Lmove SK` which are more homogeneous than those of `Lmove BBC`; the opposite holds true for the typical weeks belonging to the second cluster. Heterogeneity over time appears to emerge also in the correlation between log sales of SK and BBS products, which is slightly positive (0.191) within the largest cluster of weeks, while a mild negative correlation (-0.299) between `Lmove SK` and `Lmove BBC` is estimated in the weeks belonging to the first cluster.

The first cluster determined according to the highest estimated posterior probabilities of the selected model is composed of 20 weeks; 17 of these weeks are consecutive (from week no. 58 to week no. 74) and correspond to a period (from mid-October 1990 to mid-February 1991) characterised by a worldwide boycott campaign encouraging consumers not to buy Bumble Bee tuna because Bumble Bee was found to be buying yellow-fin tuna caught by dolphin-unsafe techniques (Baird and Quastel, 2011). The selected model seems to suggest that such events may be one of the sources of the unobserved heterogeneity detected by the analysis. The fact that the estimated effects of all the selected regressors on the log prices of both products are stronger in the first cluster of weeks and weaker in the second cluster could be associated with

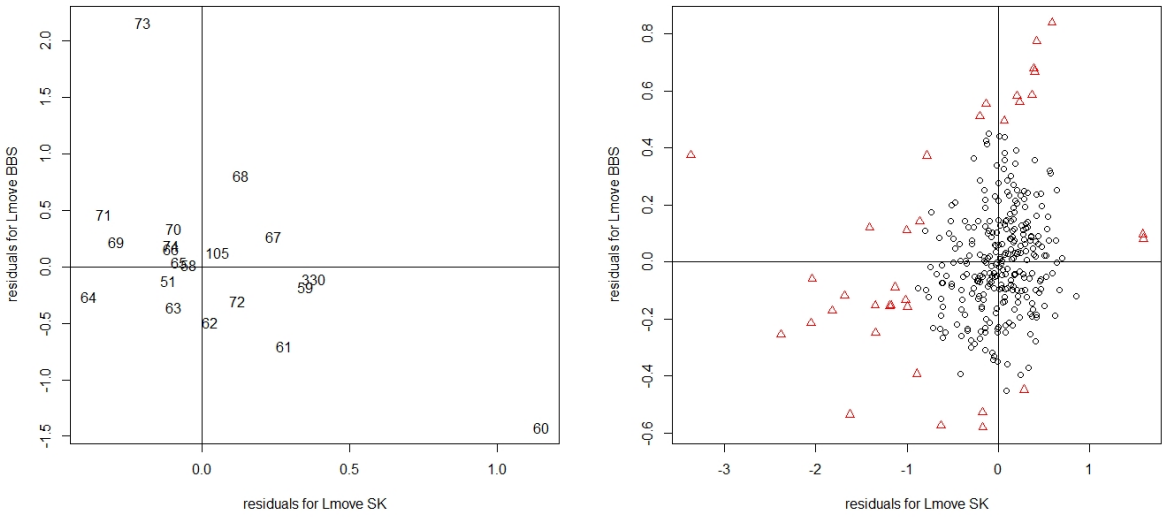

Figure 2.3: Scatterplots of the estimated residuals for the weeks assigned to the first (left) and second (right) clusters detected by the overall best model for the analysis of tuna sales. Points of the first scatterplot are labelled with the number of the corresponding weeks. Black circle and red triangle in the second scatterplot correspond to typical and outlying weeks, respectively.

those events. According to the rule for the intra-class distinction between typical observations and mild outliers illustrated in Section 2.2.4, some weeks have been classified as mild outliers within both clusters. As far as the first cluster is concerned, this has happened for week no. 60 (immediately after Halloween 1990) and week no. 73 (two weeks immediately before Presidents day 1999). For these weeks, the estimated squared Mahalanobis distances $\hat{d}^2_{i1}$, equal to 36.68 and 37.82, respectively, appear to be extremely higher than those of the other 18 weeks of the same cluster, which are comprised between 0.05 and 7.05. From the estimated sample residuals $\mathbf{y}_i - \hat{\boldsymbol{\mu}}_1(\mathbf{x}_i; \hat{\boldsymbol{\beta}}^*_1)$ for the 20 weeks belonging to the first cluster (see the scatterplot on the left side of Figure 2.3) it emerges that week no. 60 noticeably deviates from the other weeks because log unit sales of SK tuna are slightly lower than the predicted value, while an opposite result characterises the log unit sales of BBS tuna. On the contrary, the selected model identifies week no. 73 as a mild outlier mainly because of a large overestimation of the sales of BBS tuna. Among the 318 weeks of the second cluster, 35 have resulted to be mild outliers, most of which are associated with holidays and special events that took place between September 1989 and mid-October 1990 or between mid-February and May 1997. The scatterplot with the estimated sample residuals $\mathbf{y}_i - \hat{\boldsymbol{\mu}}_2(\mathbf{x}_i; \hat{\boldsymbol{\beta}}^*_2)$ for all the weeks of the second cluster (see the right side of Figure 2.3) shows that, for the majority of the 35 mild outlying weeks, the reason for the outlyingness detected by the model has been an overestimation or an underestimation of

the sales for either brands. The values of the estimated distances $\hat{d}_{i2}^2$ for the weeks that have been classified as typical are between 0.003 and 7.993; the minimum and maximum of the same distances for the outlying weeks are 8.20 and 114.95, respectively.

2.5 Conclusions

A new family of seemingly unrelated clusterwise linear regression models for possibly contaminated data has been introduced. Such models can account for heterogeneous regression data with mild outliers and multivariate correlated responses, each one depending on its own vector of covariates. This latter feature represents the main novelty of the models proposed here in reference with the ones described in Mazza and Punzo (2020). The new family encompasses several other types of Gaussian mixture-based linear regression models previously proposed in the literature. It also provides a more flexible framework for modelling data in applications where sample observations could be atypical and different covariates are expected to be relevant in the prediction of different responses, based on some prior information to be conveyed in the analysis. The new family could be made more flexible by exploiting the approach illustrated in Celeux and Govaert (1995), which allows to introduce constraints on the elements of the covariance matrices $\boldsymbol{\Sigma}_k$, $k = 1, \ldots, K$, so that models with a lower number of variances and covariances of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ in the K sub-populations are obtained. Monte Carlo studies have shown that the choice of the number of components and the reconstruction of the true classification of the sample observations can be negatively affected by the inclusion of irrelevant regressors in a clusterwise linear regression model, especially with overlapping clusters of observations. Whenever the choice of the regressors to be considered in the specification of the linear predictor of each response is questionable, models introduced here can be employed in conjunction with techniques for variable selection (e.g., genetic algorithms, stepwise strategies) in a multivariate regression setting in order to detect the relevant predictors for each regression equation. Since the ECM algorithm for the ML estimation of the model parameters does not automatically produce any estimate of the covariance matrix of the ML estimator, additional computations are necessary to obtain an assessment of the sample variability of model parameter estimates. This task could be carried out by means of some approaches commonly employed under finite mixture models (see, e.g., McLachlan and Peel, 2000). We are currently developing an extension of the methods proposed herein to some mixtures of Gaussian linear regression models with random covariates

(Punzo and McNicholas, 2017). Another avenue of future research is represented by the study of seemingly unrelated clusterwise regression models explicitly accounting for contaminated data and space/time-dependent observations.

Appendix A - Update of $\boldsymbol{\beta}_k^*$ and $\boldsymbol{\Sigma}_k$

The updates of the model parameters $\boldsymbol{\beta}_k^*$ and $\boldsymbol{\Sigma}_k$ at the $(h+1)$th first CM-step of the ECM algorithm, as illustrated in equations (2.10) and (2.11), can be obtained as follows.

$$\begin{aligned}\frac{\partial}{\partial \boldsymbol{\beta}_k^{*\prime}} Q\left(\boldsymbol{\psi}|\boldsymbol{\psi}^{(h)}\right) &= \frac{\partial}{\partial \boldsymbol{\beta}_k^{*\prime}} \sum_{i=1}^{I}\sum_{k=1}^{K} \hat{z}_{ik}^{(h)} Q_i\left(\boldsymbol{\beta}_k^*, \boldsymbol{\Sigma}_k|\boldsymbol{\psi}^{(h)}\right) = \\ &= \frac{\partial}{\partial \boldsymbol{\beta}_k^{*\prime}} \sum_{i=1}^{I}\sum_{k=1}^{K} \frac{\hat{z}_{ik}^{(h)}}{2}\Big[-\ln|\boldsymbol{\Sigma}_k| - M(1-\hat{u}_{ik}^{(h)})\ln\hat{\eta}_k^{(h)} - \hat{w}_{ik}^{(h)}\delta^2_{\boldsymbol{\Sigma}_k}(\mathbf{y}_i, \boldsymbol{\mu}_k(\mathbf{x}_i;\boldsymbol{\beta}_k^*))\Big]. \end{aligned} \tag{2.13}$$

Focusing on the squared Mahalanobis distance $\delta^2_{\boldsymbol{\Sigma}_k}(\mathbf{y}_i, \boldsymbol{\mu}_k(\mathbf{x}_i;\boldsymbol{\beta}_k^*))$ and using properties of trace and transpose, it follows that

$$\begin{aligned}\delta^2_{\boldsymbol{\Sigma}_k}(\mathbf{y}_i, \boldsymbol{\mu}_k(\mathbf{x}_i;\boldsymbol{\beta}_k^*)) &= \boldsymbol{y}_i'\boldsymbol{\Sigma}_k^{-1}\boldsymbol{y}_i - \boldsymbol{y}_i'\boldsymbol{\Sigma}_k^{-1}\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^* - \boldsymbol{\beta}_k^{*\prime}\tilde{\mathbf{x}}_i^*\boldsymbol{\Sigma}_k^{-1}\boldsymbol{y}_i + \boldsymbol{\beta}_k^{*\prime}\tilde{\mathbf{x}}_i^*\boldsymbol{\Sigma}_k^{-1}\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^* \\ &= \boldsymbol{y}_i'\boldsymbol{\Sigma}_k^{-1}\boldsymbol{y}_i - 2tr(\boldsymbol{\beta}_k^{*\prime}\tilde{\mathbf{x}}_i^*\boldsymbol{\Sigma}_k^{-1}\boldsymbol{y}_i) + \boldsymbol{\beta}_k^{*\prime}\tilde{\mathbf{x}}_i^*\boldsymbol{\Sigma}_k^{-1}\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*. \end{aligned} \tag{2.14}$$

Deriving (2.14) respect to $\boldsymbol{\beta}_k^{*\prime}$ and then replacing the so obtained result in (2.13) leads to

$$\begin{aligned}\frac{\partial}{\partial \boldsymbol{\beta}_k^{*\prime}} Q\left(\boldsymbol{\psi}|\boldsymbol{\psi}^{(h)}\right) &= \sum_{i=1}^{I} -\frac{\hat{z}_{ik}^{(h)}}{2}\hat{w}_{ik}^{(h)}\Big(-2\boldsymbol{y}_i'\boldsymbol{\Sigma}_k^{-1}\tilde{\mathbf{x}}_i^{*\prime} + 2\boldsymbol{\beta}_k^{*\prime}\tilde{\mathbf{x}}_i^*\boldsymbol{\Sigma}_k^{-1}\tilde{\mathbf{x}}_i^{*\prime}\Big) \\ &= \sum_{i=1}^{I}\hat{z}_{ik}^{(h)}\hat{w}_{ik}^{(h)}\boldsymbol{y}_i'\boldsymbol{\Sigma}_k^{-1}\tilde{\mathbf{x}}_i^{*\prime} - \sum_{i=1}^{I}\hat{z}_{ik}^{(h)}\hat{w}_{ik}^{(h)}\boldsymbol{\beta}_k^{*\prime}\tilde{\mathbf{x}}_i^*\boldsymbol{\Sigma}_k^{-1}\tilde{\mathbf{x}}_i^{*\prime}. \end{aligned} \tag{2.15}$$

Setting (2.15) equal to the null vector, solving the so obtained system with respect to $\boldsymbol{\beta}_k^{*\prime}$ and using properties of transpose results in the solution reported in equation (2.10). Finally,

$$
\begin{aligned}
&\frac{\partial}{\partial \boldsymbol{\Sigma}_k^{-1}} Q\left(\boldsymbol{\psi}|\boldsymbol{\psi}^{(h)}\right) = \frac{\partial}{\partial \boldsymbol{\Sigma}_k^{-1}} \sum_{i=1}^{I}\sum_{k=1}^{K} \hat{z}_{ik}^{(h)} Q_i\left(\boldsymbol{\beta}_k^*, \boldsymbol{\Sigma}_k|\boldsymbol{\psi}^{(h)}\right) \\
=&\frac{\partial}{\partial \boldsymbol{\Sigma}_k^{-1}} \sum_{i=1}^{I}\sum_{k=1}^{K} \frac{\hat{z}_{ik}^{(h)}}{2}\Big[-\ln|\boldsymbol{\Sigma}_k| - M(1-u_{ik})\ln\eta_k + \\
&- \hat{w}_{ik}^{(h)}\left(\boldsymbol{y}_i - \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^{*(h+1)}\right)'\boldsymbol{\Sigma}_k^{-1}\left(\boldsymbol{y}_i - \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^{*(h+1)}\right)\Big] \\
=&\frac{\partial}{\partial \boldsymbol{\Sigma}_k^{-1}} \sum_{i=1}^{I}\sum_{k=1}^{K} \frac{\hat{z}_{ik}^{(h)}}{2}\Big[\ln|\boldsymbol{\Sigma}_k^{-1}| - M(1-u_{ik})\ln\eta_k + \\
&- \hat{w}_{ik}^{(h)} tr\left(\boldsymbol{\Sigma}_k^{-1}(\boldsymbol{y}_i - \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^{*(h+1)})(\boldsymbol{y}_i - \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^{*(h+1)})'\right)\Big]. \\
=&\sum_{i=1}^{I} \frac{\hat{z}_{ik}^{(h)}}{2}\Big[\boldsymbol{\Sigma}_k - \hat{w}_{ik}^{(h)}\left(\boldsymbol{y}_i - \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^{*(h+1)})(\boldsymbol{y}_i - \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^{*(h+1)}\right)'\Big], \qquad (2.16)
\end{aligned}
$$

where the second and third equalities are obtained using properties of trace and transpose and differentiation rules of functions of matrices. Setting (2.16) equal to the null matrix and solving the resulting system with respect to $\boldsymbol{\Sigma}_k$ gives the update in equation (2.11).

Bibliography

Aitken AC (1926) A series formula for the roots of algebraic and transcendental equations. Proc R Soc Edinb 45(1):14–22

Aitkin M, Wilson TG (1980) Mixture models, outliers, and the EM algorithm. Technometrics 22(3):325–331

Andrews JL, McNicholas PD (2011) Extending mixtures of multivariate t-factor analyzers. Stat Comput 21(3):361–373

Baek J, McLachlan GJ (2011) Mixtures of common t-factor analyzers for clustering high-dimensional microarray data. Bioinformatics 27(9):1269–1276

Bagnato L, Punzo A, Zoia MG (2017) The multivariate leptokurtic-normal distribution and its application in model-based clustering. Can J Stat 45(1):95–119

Bai X, Yao W, Boyer JE (2012) Robust fitting of mixture regression models. Comput Stat Data Anal 56(7):2347–2359

Baird IG, Quastel N (2011) Dolphin-safe tuna from California to Thailand: localisms in environmental certification of global commodity networks. Ann Assoc Am Geogr 101(2):337–355

Bartolucci F, Scaccia L (2005) The use of mixtures for dealing with non-normal regression errors. Comput Stat Data Anal 48(4):821–834

Biernacki C, Celeux G, Govaert G (2000) Assessing a mixture model for clustering with the integrated completed likelihood. IEEE Trans Pattern Anal Mach Intell 22(7):719–725

Biernacki C, Celeux G, Govaert G (2003) Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixture models. Comput Stat Data Anal 41(3–4):561–575

Cadavez VAP, Hennningsen A (2012) The use of seemingly unrelated regression (SUR) to predict the carcass composition of lambs. Meat Sci 92(4):548–553

Celeux G, Govaert G (1995) Gaussian parsimonious clustering models. Pattern Recognit 28(5):781–793

Chevalier JA, Kashyap AK, Rossi PE (2003) Why don't prices rise during periods of peak demand? Evidence from scanner data. Am Econ Rev 93(1):15–37

Dang UJ, Browne RP, McNicholas PD (2015) Mixtures of multivariate power exponential distributions. Biom 71(4):1081–1089

Dang UJ, Punzo A, McNicholas PD, Ingrassia S, Browne RP (2017) Multivariate response and parsimony for Gaussian cluster-weighted models. J Classif 34(1):4–34

De Sarbo WS, Cron WL (1988) A maximum likelihood methodology for clusterwise linear regression. J Classif 5(2):249–282

De Veaux RD (1989) Mixtures of linear regressions. Comput Stat Data Anal 8(3):227–245

Dempster A, Laird N, Rubin D (1977) Maximum likelihood for incomplete data via the EM algorithm. J Roy Stat Soc: Ser B 39(1):1–38

Depraetere N, Vandebroek M (2014) Order selection in finite mixtures of linear regressions. Stat Pap 55(3):871–911

Ding C (2006) Using regression mixture analysis in educational research. Pract Assess Res Eval 11(1):1–11

Disegna M, Osti L (2016) Tourists' expenditure behaviour: the influence of satisfaction and the dependence of spending categories. Tour Econ 22(1):5–30

Dyer WJ, Pleck J, McBride B (2012) Using mixture regression to identify varying effects: a demonstration with paternal incarceration. J Marriage Fam 74(5):1129–1148

Elhenawy M, Rakha H, Chen H (2017) An automatic traffic congestion identification algorithm based on mixture of linear regressions. In: Helfert M, Klein C, Donnellan B, Gusikhin O (eds) Smart cities, green technologies, and intelligent transport systems. Springer, Cham, pp 242–256

Fair RC, Jaffe DM (1972) Methods of estimation for markets in disequilibrium. Econometrica 40:497–514

Frühwirth-Schnatter S (2006) Finite mixture and Markov switching models. Springer, New York

Galimberti G, Scardovi E, Soffritti G (2016) Using mixtures in seemingly unrelated linear regression models with non-normal errors. Stat Comput 26(5):1025–1038

Galimberti G, Soffritti G (2020) Seemingly unrelated clusterwise linear regression. Adv Data Anal Classif 14(2):235–260

Giles S, Hampton P (1984) Regional production relationships during the industrialization of New Zealand, 1935–1948. Reg Sci 24(4):519–532

Gómez E, Gómez-Viilegas MA, Marin JM (1998) A multivariate generalization of the power exponential family of distributions. Commun Stat Theory Methods 27(3):589–600

Heidari S, Keshavarzi S, Mirahmadizadeh A (2017) Application of seemingly unrelated regression (SUR) in determination of risk factors of fatigue and general health among the employees of petrochemical companies. J Health Sci Surveillance Sys 5(4):1–8

Hennig C (2000) Identifiability of models for clusterwise linear regression. J Classif 17:273–296

Henningsen A, Hamann JD (2007) **systemfit**: a package for estimating systems of simultaneous equations in `R`. J Stat Softw 23(4):1–40

Hosmer DW (1974) Maximum likelihood estimates of the parameters of a mixture of two regression lines. Commun Stat Theory Methods 3(10):995–1006

Hubert L, Arabie P (1985) Comparing partitions. J Classif 2(1):193–218

Jones PN, McLachlan GJ (1992) Fitting finite mixture models in a regression context. Aust J Stat 34(2):233–240

Kamakura W (1988) A least squares procedure for benefit segmentation with conjoint experiments. J Mark Res 25(2):157–167

Karlis D, Xekalaki E (2003) Choosing initial values for the EM algorithm for finite mixtures. Comput Stat Data Anal 41(3–4):577–590

Keshavarzi S, Ayatollahi SMT, Zare N, Pakfetrat M (2012) Application of seemingly unrelated regression in medical data with intermittently observed time-dependent covariates. Comput Math Methods Med 2012, 821643

Keshavarzi S, Ayatollahi SMT, Zare N, Sharif F (2013) Quality of life of childbearing age women and its associated factors: an application of seemingly unrelated regression (SUR) models. Qual Life Res 22(6):1255–1263

Kibria BMG, Haq MS (1999) The multivariate linear model with multivariate t and intra-class covariance structure. Stat Pap 40(3):263–276

Lachos VH, Angolini T, Abanto-Valle CA (2011) On estimation and local influence analysis for measurement errors models under heavy-tailed distributions. Stat Pap 52(3):567–590

Lange KL, Little RJA, Taylor JMG (1989) Robust statistical modeling using the t distribution. J Am Stat Assoc 84(408):881–896

Magnus JR, Neudecker H (1988) Matrix differential calculus with applications in statistics and econometrics. Wiley, New York

Maronna RA, Martin RD, Yohai VJ (2006) Robust statistics: theory and methods. Wiley, Chichester

Mazza A, Punzo A (2020) Mixtures of multivariate contaminated normal regression models Stat Pap 61(2):787–822

McDonald SE, Shin S, Corona R et al (2016) Children exposed to intimate partner violence: identifying differential effects of family environment on children's trauma and psychopathology symptoms through regression mixture models. Child Abus Negl 58:1–11

McLachlan GJ, Peel D (2000) Finite mixture models. Wiley, New York

McNicholas PD (2010) Model-based classification using latent Gaussian mixture models. J Stat Plan Inference 140(5):1175–1181

Meng XL, Rubin DB (1993) Maximum likelihood estimation via the ECM algorithm: A general framework. Biometrika 80(2):267–278

Park T (1993) Equivalence of maximum likelihood estimation and iterative two-stage estimation for seemingly unrelated regression models. Commun Stat Theory Methods 22(8):2285–2296

Punzo A, Bagnato L (2020) Allometric analysis using the multivariate shifted exponential normal distribution. Biometr J 62(6):1525–1543

Punzo A, Bagnato L (2021) The multivariate tail-inflated normal distribution and its application in finance. J Stat Comput Simul 91(1):1–36

Punzo A, McNicholas PD (2016) Parsimonious mixtures of multivariate contaminated normal distributions. Biometr J 58(6):1506–1537

Punzo A, McNicholas PD (2017) Robust clustering in regression analysis via the contaminated Gaussian cluster-weighted model. J Classif 34(2):249–293

Qin LX, Self SG (2006) The clustering of regression models method with applications in gene expression data. Biometrics 62(2):526–533

Quandt RE, Ramsey JB (1978) Estimating mixtures of normal distributions and switching regressions. J Am Stat Assoc 73(364):730–738

R Core Team (2021) `R`: a language and environment for statistical computing. `R` Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org

Ritter G (2015) Robust cluster analysis and variable selection. Chapman and Hall, Boca Raton

Rossi PE (2012) **bayesm**: Bayesian inference for marketing/micro-econometrics. `R` package version 2.2-5. http://CRAN.R-project.org/package=bayesm

Schwarz G (1978) Estimating the dimension of a model. Ann Stat 6(2):461–464

Scrucca L, Fop M, Murphy TB, Raftery AE (2017) **mclust 5**: clustering, classification and density estimation using Gaussian finite mixture models. R J 8(1):205–223

Soffritti G, Galimberti G (2011) Multivariate linear regression with non-normal errors: a solution based on mixture models. Stat Comput 21(4):523–536

Srivastava VK, Giles DEA (1987) Seemingly unrelated regression equations models. Marcel Dekker, New York

Tashman A, Frey RJ (2009) Modeling risk in arbitrage strategies using finite mixtures. Quant Finance 9(5):495–503

Turner TR (2000) Estimating the propagation rate of a viral infection of potato plants via mixtures of regressions. Appl Stat 49(3):371–384

Tukey JW (1960) A survey of sampling from contaminated distributions. In: Olkin I (ed) Contributions to probability and statistics: essays in honor of Harold Hotelling, Stanford studies in mathematics and statistics. Stanford University Press, California, pp 448–485

Van Horn ML, Jaki T, Masyn K et al (2015) Evaluating differential effects using regression interactions and regression mixture models. Educ Psychol Meas 75(4):677–714

Wedel M (2002) Concomitant variables in finite mixture models. Stat Neerl 56(3):362–375

White EN, Hewings GJD (1982) Space-time employment modelling: some results using seemingly unrelated regression estimators. J Reg Sci 22(3):283–302

Yao W, Wei Y, Yu C (2014) Robust mixture regression using the t-distribution. Comput Stat Data Anal 71:116–127

Zellner A (1962) An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. J Am Stat Assoc 57(298):348–368

Chapter 3

Parsimonious Mixtures of Seemingly Unrelated Contaminated Normal Regression Models[1]

[1]This chapter coincides with the published paper: Perrone G., Soffritti G. (2022). "Parsimonious mixtures of seemingly unrelated contaminated normal regression models". In P. Brito, J. G. Dias, B. Lausen, A. Montanari, R. Nugent. *Classification and Data Science in the Digital Age: the 17th Conference of the International Federation of Classification Societies (IFCS 2022), Springer Cham. Series E-ISSN: 2198-3321* (pp. 1-8) https://link.springer.com/book/9783031090332 (in press).

Abstract

In recent years, the research into linear multivariate regression based on finite mixture models has been intense. With such an approach, it is possible to perform regression analysis for a multivariate response by taking account of the possible presence of several unknown latent homogeneous groups, each of which is characterised by a different linear regression model. For a continuous multivariate response, mixtures of normal regression models are usually employed. However, in real data, it is not unusual to observe mildly atypical observations that can negatively affect the estimation of the regression parameters under a normal distribution in each mixture component. Furthermore, in some fields of research, a multivariate regression model with a different vector of covariates for each response should be specified, based on some prior information to be conveyed in the analysis. To take account of all these aspects, mixtures of contaminated seemingly unrelated normal regression models have been recently developed. A further extension of such an approach is presented here so as to ensure parsimony, which is obtained by imposing constraints on the group-covariance matrices of the responses. A description of the resulting parsimonious mixtures of seemingly unrelated contaminated regression models is provided together with the results of a numerical study based on the analysis of a real dataset, which illustrates their practical usefulness.

Keywords: Contaminated normal distribution, ECM algorithm, mixture of regression models, model-based cluster analysis, seemingly unrelated regression.

3.1 Introduction

Seemingly unrelated (SU) regression equations are usually employed in a multivariate regression analysis whenever the dependence of a vector $\mathbf{Y} = (Y_1, \ldots, Y_m, \ldots, Y_M)'$ of M continuous variables on a vector $\mathbf{X} = (X_1, \ldots, X_p, \ldots, X_P)'$ of P regressors has to be modelled by allowing the error terms in the different equations to be correlated and, thus, the regression parameters of the M equations have to be jointly estimated (Srivastava and Giles, 1987). With such an approach, the researcher is also enabled to convey prior information on the phenomenon under study into the specification of the regression equations by defining a different vector of regressors for each dependent variable. This latter feature is particularly useful in any situation in which different regressors are expected to be relevant in the prediction of different responses, such as in White and Hewings (1982); Cadavez and Hennningsen (2012); Disegna and Osti (2016). This approach has been recently embedded into the framework of Gaussian mixture models, leading to multivariate SU normal regression mixtures Galimberti and Soffritti (2020). In these models, the effect of the regressors on the dependent variables changes with some unknown latent sub-populations composing the population that has generated the sample of observations to be analysed. Thus, when the sample is characterised by unobserved heterogeneity, model-based cluster analysis is simultaneously carried out.

Another source of complexity which could affect the data and make the prediction of $\mathbf{Y}$ a difficult task to perform is represented by mildly atypical observations (Ritter, 2015). Robust methods of parameter estimation insensitive to the presence of such observations in a sample characterised by unobserved heterogeneity have been introduced in Mazza and Punzo (2020), where the conditional distribution $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ is modelled through a mixture of K multivariate contaminated normal models, where K is the number of the latent sub-populations. A limitation associated with these latter models is that the same vector of regressors has to be specified for the prediction of all the dependent variables. To overcome this limitation while preserving all the features mentioned above, a more flexible approach which employs mixtures of multivariate SU contaminated normal regression models has been recently introduced in Perrone and Soffritti (2023). These latter models are able to capture the linear effects of the regressors on the dependent variables from sample observations coming from heterogeneous populations. The researcher is also enabled to specify a different vector of regressors for each dependent variable. Finally, a robust estimation of the regression parameters and the detection of mild outliers in

the data are ensured.

In the presence of many responses and many latent sub-populations, analyses based on these latter models can become unfeasible in practical applications because of a large number of model parameters. In order to keep this number as low as possible, an approach due to Celeux and Govaert (1995), based on the spectral decompositions of the K covariance matrices of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$, is exploited here so as to obtain fourteen different covariance structures. The resulting parsimonious mixtures of SU contaminated regression models are described in Section 3.2. The usefulness of these new models is illustrated through a study aiming at determining the effect of prices and promotional activities on sales of canned tuna in the US market. A summary of the obtained results is provided in Section 3.3 .

3.2 Parsimonious SU contaminated normal regression mixtures

In a system of M SU regression equations for modelling the linear dependence of $\mathbf{Y}$ on $\mathbf{X}$, let $\mathbf{X}_m = (X_{m_1}, X_{m_2}, \ldots, X_{m_{P_m}})'$ be the P_m-dimensional sub-vector of $\mathbf{X}$ composed of the P_m regressors expected to be relevant for the explanation of Y_m, for $m = 1, \ldots, M$. Furthermore, let $\mathbf{X}^*_m = (1, \mathbf{X}'_m)'$. The mixture of K SU normal regression models described in Galimberti and Soffritti (2020) can be defined as follows:

$$\mathbf{Y} = \begin{cases} \tilde{\mathbf{X}}^{*\prime}\boldsymbol{\beta}^*_1 + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N_M(\mathbf{0}_M, \boldsymbol{\Sigma}_1) \text{ with probability } \pi_1, \\ \quad \ldots \\ \tilde{\mathbf{X}}^{*\prime}\boldsymbol{\beta}^*_K + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim N_M(\mathbf{0}_M, \boldsymbol{\Sigma}_K) \text{ with probability } \pi_K, \end{cases} \tag{3.1}$$

where π_k is the prior probability of the kth latent sub-population, with $\pi_k > 0$ for $k = 1, \ldots, K$; $\sum_{k=1}^K \pi_k = 1$; $\tilde{\mathbf{X}}^*$ is the following $(P^* + M) \times M$ partitioned matrix:

$$\tilde{\mathbf{X}}^* = \begin{bmatrix} \mathbf{X}^*_1 & \mathbf{0}_{P_1+1} & \ldots & \mathbf{0}_{P_1+1} \\ \mathbf{0}_{P_2+1} & \mathbf{X}^*_2 & \ldots & \mathbf{0}_{P_2+1} \\ \vdots & \vdots & & \vdots \\ \mathbf{0}_{P_M+1} & \mathbf{0}_{P_M+1} & \ldots & \mathbf{X}^*_M \end{bmatrix},$$

with $\mathbf{0}_{P_m+1}$ denoting the $(P_m + 1)$-dimensional null vector; $P^* = \sum_{m=1}^M P_m$;
$\boldsymbol{\beta}^*_k = (\boldsymbol{\beta}^{*\prime}_{k1}, \ldots, \boldsymbol{\beta}^{*\prime}_{km}, \ldots, \boldsymbol{\beta}^{*\prime}_{kM})'$ is the $(P^* + M)$-dimensional vector containing all the linear

effects on the M responses in the kth latent sub-population, with $\boldsymbol{\beta}^*_{km} = (\beta_{0k,m}, \boldsymbol{\beta}'_{km})'$, for $m = 1, \ldots, M$; $\boldsymbol{\epsilon} = (\epsilon_1, \ldots, \epsilon_M)'$ is the vector of the errors, which are supposed to be independent and identically distributed; $N_M(\mathbf{0}_M, \boldsymbol{\Sigma}_k)$ denotes the M-dimensional normal distribution with mean vector $\mathbf{0}_M$ and positive-definite covariance matrix $\boldsymbol{\Sigma}_k$. From now on, this mixture regression model is denoted as MSUN. When $\mathbf{X}_m = \mathbf{X}\ \forall m$ (the P regressors are employed in all the M equations), model (3.1) reduces to the mixtures of K normal (MN) regression models (see Jones and McLachlan (1992)).

When the data are contaminated by the presence of mild outliers, departures from the normal distribution could be observed within any of the K latent sub-populations. A model able to manage this situation has been recently introduced in Perrone and Soffritti (2023). It has been obtained from equation (3.1) by replacing the normal distribution with the contaminated normal distribution. Under this latter distribution, the probability density function (p.d.f.) of $\boldsymbol{\epsilon}$ within the kth sub-population is equal to $h(\boldsymbol{\epsilon}; \boldsymbol{\vartheta}_k) = \alpha_k \phi_M(\boldsymbol{\epsilon}; \mathbf{0}_M, \boldsymbol{\Sigma}_k) + (1 - \alpha_k)\phi_M(\boldsymbol{\epsilon}; \mathbf{0}_M, \eta_k \boldsymbol{\Sigma}_k)$, where $\phi_M(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the p.d.f. of the distribution $N_M(\mathbf{0}_M, \boldsymbol{\Sigma}_k)$, $\alpha_k \in (0.5, 1)$ and $\eta_k > 1$ are the proportion of typical observations within the kth sub-population and a parameter that inflates the elements of $\boldsymbol{\Sigma}_k$, respectively, and $\boldsymbol{\vartheta}_k = (\alpha_k, \eta_k, \boldsymbol{\Sigma}_k)$. As a consequence, a mixture of K SU contaminated normal (MSUCN) regression models is given by:

$$\mathbf{Y} = \begin{cases} \tilde{\mathbf{X}}^{*\prime} \boldsymbol{\beta}^*_1 + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim CN_M(\alpha_1, \eta_1, \mathbf{0}_M, \boldsymbol{\Sigma}_1) \text{ with probability } \pi_1, \\ \quad \ldots \\ \tilde{\mathbf{X}}^{*\prime} \boldsymbol{\beta}^*_K + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim CN_M(\alpha_K, \eta_K, \mathbf{0}_M, \boldsymbol{\Sigma}_K) \text{ with probability } \pi_K, \end{cases} \tag{3.2}$$

where $CN_M(\alpha_k, \eta_k, \mathbf{0}_M, \boldsymbol{\Sigma}_k)$ denotes the M-dimensional contaminated normal distribution described by the p.d.f. $h(\boldsymbol{\epsilon}; \boldsymbol{\vartheta}_k)$. The parameter vector of model (3.2) is $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_k, \ldots, \boldsymbol{\psi}_K)$, where $\boldsymbol{\psi}_k = (\pi_k, \boldsymbol{\theta}_k)$, $\boldsymbol{\theta}_k = (\boldsymbol{\beta}^*_k, \boldsymbol{\vartheta}_k)$. The number of free elements of $\boldsymbol{\psi}$ is $n_{\boldsymbol{\psi}} = 3K - 1 + K(P^* + M) + n_{\boldsymbol{\sigma}}$, where $n_{\boldsymbol{\sigma}}$ denotes the total number of free variances and covariances, with $n_{\boldsymbol{\sigma}} = K n_{\boldsymbol{\Sigma}}$ and $n_{\boldsymbol{\Sigma}} = \frac{M(M+1)}{2}$. When $\mathbf{X}_m = \mathbf{X}\ \forall m$, model (3.2) coincides with the mixture of K contaminated normal (MCN) regression models described in Mazza and Punzo (2020). For $\alpha_k \to 1$ or $\eta_k \to 1\ \forall k$, model (3.2) reduces to model (3.1). Conditions ensuring identifiability of models (3.2) are provided in Perrone and Soffritti (2023). The ML estimation of $\boldsymbol{\psi}$ in equation (3.2) can be carried out by means of a sample $\mathcal{S} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_I, \mathbf{y}_I)\}$ of I independent observations drawn from model (3.2) and an expectation-conditional max-

imisation (ECM) algorithm Meng and Rubin (1993). Details about this algorithm, including strategies for the initialisation of $\boldsymbol{\psi}$ and convergence criteria, are illustrated in Perrone and Soffritti (2023). In practical applications, the value of K is generally unknown and has to be properly chosen. This task can be carried out by resorting to model selection criteria, such as the Bayesian information criterion Schwarz (1978): $BIC = 2\ell(\hat{\boldsymbol{\psi}}) - n_{\boldsymbol{\psi}} \ln I$, where $\hat{\boldsymbol{\psi}}$ is the maximum likelihood estimator of $\boldsymbol{\psi}$. Another commonly used information criterion is the integrated completed likelihood Biernacki et al. (2000), which admits two slightly different formulations: $ICL_1 = BIC + 2\sum_{i=1}^{I}\sum_{k=1}^{K} \text{MAP}(\hat{z}_{ik}) \ln \hat{z}_{ik}$ and $ICL_2 = BIC + 2\sum_{i=1}^{I}\sum_{k=1}^{K} \hat{z}_{ik} \ln \hat{z}_{ik}$, where $\hat{z}_{ik}$ is the estimated posterior probability that the ith sample observation come from the kth sub-population (for further details see Perrone and Soffritti (2023)), $\text{MAP}(\hat{z}_{ik}) = 1$ if $\max_h\{\hat{z}_{ih}\}$ occurs when $h = k$ ($\text{MAP}(\hat{z}_{ik}) = 0$ otherwise). Whenever the specification of the subvectors $\mathbf{X}_m$, $m = 1, \ldots, M$, to be considered in the M equations of the multivariate regression model is questionable, such criteria can also be employed to perform subset selection.

As the number of free parameters $n_{\boldsymbol{\psi}}$ incresases quadratically with M, analyses based on model (3.2) can become unfeasible in real applications. A way to manage this problem can be based on the introduction of suitable constraints on the elements of $\boldsymbol{\Sigma}_k$, $k = 1, \ldots, K$, based on the following eigen-decomposition Celeux and Govaert (1995): $\boldsymbol{\Sigma}_k = \lambda_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k'$, where $\lambda_k = |\boldsymbol{\Sigma}_k|^{1/M}$, $\mathbf{A}_k$ is a diagonal matrix with entries (sorted in decreasing order) proportional to the eigenvalues of $\boldsymbol{\Sigma}_k$ (with the constraint $|\mathbf{A}_k| = 1$) and $\mathbf{D}_k$ is a $M \times M$ orthogonal matrix of the eigenvectors of $\boldsymbol{\Sigma}_k$ (ordered according to the eigenvalues). This decomposition allows to obtain variances and covariances in $\boldsymbol{\Sigma}_k$ from λ_k, $\mathbf{A}_k$ and $\mathbf{D}_k$. From a geometrical point of view, λ_k determines the volume, $\mathbf{A}_k$ the shape and $\mathbf{D}_k$ the orientation of the kth cluster of sample observations detected by the fitted model. By constraining λ_k, $\mathbf{A}_k$ and $\mathbf{D}_k$ to be equal or variable across the K clusters, a class of fourteen mixtures of K SUCN regression models is obtained (see Table 3.1). With variable volumes, shapes and orientations (`VVV` in Table 3.1), the resulting model coincides with (3.2). When $K > 1$, the other covariance structures allow to obtain thirteen different parsimonious mixtures of K SUCN regression models (i.e.: with a reduced $n_{\boldsymbol{\sigma}}$). When $K = 1$, the possible covariance structures for $\boldsymbol{\Sigma}_1$ are: diagonal with different entries, diagonal with the same entries and fully unconstrained. The ML estimation of $\boldsymbol{\psi}$ under model (3.2) with any of these parameterisations can be carried out through an ECM algorithm in which the CM-step update for $\boldsymbol{\Sigma}_k$ can be computed either in closed form or using iterative procedures, depending on the parameterisation to be employed (see Celeux and Govaert (1995)).

Table 3.1: Features of the parameterisations for the covariance matrices $\boldsymbol{\Sigma}_k$, $k = 1, \ldots, K$ $(K > 1)$.

Acronym	Covariance structure	Volume	Shape	Orientation	CM step	$n_{\boldsymbol{\sigma}}$
`EEE`	$\lambda \mathbf{D}\mathbf{A}\mathbf{D}'$	Equal	Equal	Equal	Closed	$n_{\boldsymbol{\Sigma}}$
`VVV`	$\lambda_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k'$	Variable	Variable	Variable	Closed	$Kn_{\boldsymbol{\Sigma}}$
`EII`	$\lambda \mathbf{I}$	Equal	Spherical	–	Closed	1
`VII`	$\lambda_k \mathbf{I}$	Variable	Spherical	–	Closed	K
`EEI`	$\lambda \mathbf{A}$	Equal	Equal	Axis-aligned	Closed	M
`VEI`	$\lambda_k \mathbf{A}$	Variable	Equal	Axis-aligned	Iterative	$M + K - 1$
`EVI`	$\lambda \mathbf{A}_k$	Equal	Variable	Axis-aligned	Closed	$MK - (K - 1)$
`VVI`	$\lambda_k \mathbf{A}_k$	Variable	Variable	Axis-aligned	Closed	MK
`EEV`	$\lambda \mathbf{D}_k \mathbf{A} \mathbf{D}_k'$	Equal	Equal	Variable	Iterative	$Kn_{\boldsymbol{\Sigma}} - (K - 1)M$
`VEV`	$\lambda_k \mathbf{D}_k \mathbf{A} \mathbf{D}_k'$	Variable	Equal	Variable	Iterative	$Kn_{\boldsymbol{\Sigma}} - (K - 1)(M - 1)$
`EVE`	$\lambda \mathbf{D}\mathbf{A}_k\mathbf{D}'$	Equal	Variable	Equal	Iterative	$n_{\boldsymbol{\Sigma}} - (K - 1)(M - 1)$
`VVE`	$\lambda_k \mathbf{D}\mathbf{A}_k\mathbf{D}'$	Variable	Variable	Equal	Iterative	$n_{\boldsymbol{\Sigma}} - (K - 1)M$
`VEE`	$\lambda_k \mathbf{D}\mathbf{A}\mathbf{D}'$	Variable	Equal	Equal	Iterative	$n_{\boldsymbol{\Sigma}} - (K - 1)$
`EVV`	$\lambda \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k'$	Equal	Variable	Variable	Iterative	$Kn_{\boldsymbol{\Sigma}} - (K - 1)$

3.3 Analysis of U.S. canned tuna sales

The models illustrated in Section 3.2 have been fitted to a dataset Chevalier et al. (2003) containing the volume of sales (`Move`), a measures of the display activity (`Nsale`) and the log price (`Lprice`) for seven of the top 10 U.S. brands in the canned tuna product category in the $I = 338$ weeks between September 1989 and May 1997. The goal of the analysis is to study the dependence of canned tuna sales on prices and promotional activites for two products: Star Kist 6 oz. (SK) and Bumble Bee Solid 6.12 oz. (BBS). To this end, the following vectors have been considered: $\mathbf{Y}' = (Y_1 = \texttt{Lmove SK}, Y_2 = \texttt{Lmove BBS})$, $\mathbf{X}' = (X_1 = \texttt{Nsale SK}, X_2 = \texttt{Lprice SK}, X_3 = \texttt{Nsale BBS}, X_4 = \texttt{Lprice BBS})$, where `Lmove` denotes the logarithm of `Move`. The analysis has been carried out using all the parameterisations of the MSUN, MN, MCSUN and MCN models for each $K \in \{1, 2, 3, 4, 5, 6\}$. Furthermore, MSUN and MCSUN models have been fitted by considering all possible subvectors of $\mathbf{X}$ as vectors $\mathbf{X}_m$, $m = 1, 2$, for each K. In this way, best subset selections for `Lmove SK` and `Lmove BBS` have been included in the analysis both with and without contamination. The overall number of fitted models is 37376, including the fully unconstrained models (i.e., with the `VVV` parameterisation) previously employed in Perrone and Soffritti (2023) to perform the same analysis.

Table 3.2 reports some information about the nine models which best fit the analysed dataset according to the three model selection criteria over the six examined values of K within each model class. An analysis based on a single linear regression model ($K = 1$), both with and without contamination, appears to be inadequate according to all criteria. All the examined criteria indicate that the overall best model for studying the effect of prices and promotional

activities on sales of SK and BBS tuna is a parsimonious mixture of two SU contaminated Gaussian linear regression models with the `EVE` parameterisation for the covariance matrices in which the log unit sales of SK tuna are regressed on the log prices and the promotional activites of the same brand, while the regressors selected for the BBS log unit sales are the log prices of both brands and the promotional activites of BBS. Thus, the analysis suggests that two sources of complexity affect the analysed dataset: unobserved heterogeneity over time ($K = 2$ clusters of weeks have been detected) and the presence of mildly atypical observations. Since the two estimated proportions of typical observations are quite similar (see the values of $\hat{\alpha}_k$ in Table 3.3), contamination seems to characterise the two clusters of weeks detected by the model almost in the same way. As far as the strength of the contaminating effects on the conditional variances and covariances of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$ is concerned, it appears to be stronger in the first cluster, where the estimated inflation parameter is larger ($\hat{\eta}_1 = 15.70$). By focusing the attention on the other estimates, it appears that also some of the estimated regression coefficients, variances and covariances are affected by heterogeneity over time. Sales of SK tuna results to be negatively affected by prices and positively affected by promotional activites of the same brand within both clusters detected by the model, but with effects which are sligthly stronger in the first cluster of weeks. A similar behavior is detected for the estimated regression equation for `Lmove BBS`, which also highlights that `Lmove BBS` are positively affected by the log prices of SK tuna, especially in the first cluster of weeks. Furthermore, typical weeks in the first cluster show values of `Lmove SK` which are more homogeneous than those of `Lmove BBC`; the opposite holds true for the typical weeks belonging to the second cluster. Also the correlation between log sales of SK and BBS products results to be affected by heterogeneity over time: while in the largest cluster of weeks this correlation has been estimated to be slightly positive (0.200), the first cluster is characterised by a mild estimated negative correlation (-0.151). An interesting feature of this latter cluster is that 17 out of the 20 weeks which have been assigned to this cluster are consecutive from week no. 58 to week no. 74, which correspond to the period from mid-October 1990 to mid-February 1991 characterised by a worldwide boycott campaign encouraging consumers not to buy Bumble Bee tuna because Bumble Bee was found to be buying yellow-fin tuna caught by dolphin-unsafe techniques Baird and Quastel (2011). Such events could represent one of the sources of the unobserved heterogeneity detected by the model. According to the overall best model, some weeks have beed detected to be mild outliers. In the first cluster, this has happened for week no. 60 (immediately after Halloween 1990) and week no. 73 (two weeks immediately before

Table 3.2: Maximised log-likelihood $\ell(\hat{\boldsymbol{\psi}})$ and values of BIC, ICL_1 and ICL_2 for nine models selected from the classes MSUCN, MCN, MSUN and MN in the analysis of tuna sales.

Model class	K	Acronym	$\mathbf{X}_1$	$\mathbf{X}_2$	$\ell(\hat{\boldsymbol{\psi}})$	n_ψ	BIC	ICL_1	ICL_2
MSUCN	2	`EVE`	X_1, X_2	X_2, X_3, X_4	-242.9	23	-619.8	-625.7	-635.8
MCN	2	`EVI`	$\mathbf{X}$	$\mathbf{X}$	-239.6	28	-642.2	-648.9	-663.2
MCN	2	`EEV`	$\mathbf{X}$	$\mathbf{X}$	-240.8	29	-650.6	-650.8	-652.0
MCN	3	`EVI`	X_1, X_2, X_4	X_1, X_2, X_4	-214.2	36	-638.0	-703.1	-788.6
MSUN	2	`VEV`	X_1, X_2	X_3, X_4	-279.3	18	-663.4	-673.1	-692.1
MSUN	3	`EEV`	X_2, X_3	X_2, X_3, X_4	-259.8	28	-682.7	-684.7	-688.0
MSUN	5	`VVV`	X_2, X_3	X_1, X_4	-167.4	49	-620.0	-701.1	-780.3
MN	3	`EEV`	X_2, X_3, X_4	X_2, X_3, X_4	-258.7	31	-697.9	-699.6	-702.1
MN	4	`VVE`	X_2, X_4	X_2, X_4	-216.6	36	-642.9	-725.3	-832.9

Table 3.3: Parameter estimates of the overall best model for the analysis of tuna sales.

$\hat{\boldsymbol{\psi}}$	$k=1$	$k=2$
$\hat{\pi}_k$	0.062	0.938
$\hat{\alpha}_k$	0.810	0.844
$\hat{\eta}_k$	15.70	6.94
$\hat{\boldsymbol{\beta}}'^*_{k1}$	$(8.87, 0.56, -4.70)$	$(8.64, 0.27, -3.09)$
$\hat{\boldsymbol{\beta}}'^*_{k2}$	$(15.04, 3.92, 2.83, -17.76)$	$(9.98, 0.25, 0.12, -3.83)$
$\hat{\boldsymbol{\Sigma}}_k$	$\begin{pmatrix} 0.034 & -0.009 \\ -0.009 & 0.105 \end{pmatrix}$	$\begin{pmatrix} 0.121 & 0.012 \\ 0.012 & 0.030 \end{pmatrix}$

Presidents day 1999). The analysis of the estimated sample residuals $\mathbf{y}_i - \hat{\boldsymbol{\mu}}_1(\mathbf{x}_i; \hat{\boldsymbol{\beta}}_1^*)$ for the 20 weeks belonging to the first cluster (see the scatterplot on the left side of Figure 3.1) clearly show that weeks 60 and 73 noticeably deviates from the other weeks. Among the 318 weeks of the second cluster, 32 have resulted to be mild outliers, most of which are associated with holidays and special events that took place between September 1989 and mid-October 1990 or between mid-February and May 1997 (see the scatterplot on the right side of Figure 3.1). These results are almost equal to those obtained using the best overall fully unconstrained fitted model in the analysis presented in Perrone and Soffritti (2023). However, the `EVE` parameterisation for the MSUCN model has allowed to obtain a better trade-off among the fit, the model complexity and the uncertainty of the estimated partition of the weeks; furthermore, it has led to a slightly lower number of mild outliers in the second cluster of weeks.

3.4 Conclusions

The parsimonious mixtures of seemingly unrelated linear regression models for contaminated data introduced here can account for heterogeneous regression data both in the presence of mild outliers and multivariate correlated dependent variables, each of which is regressed on a

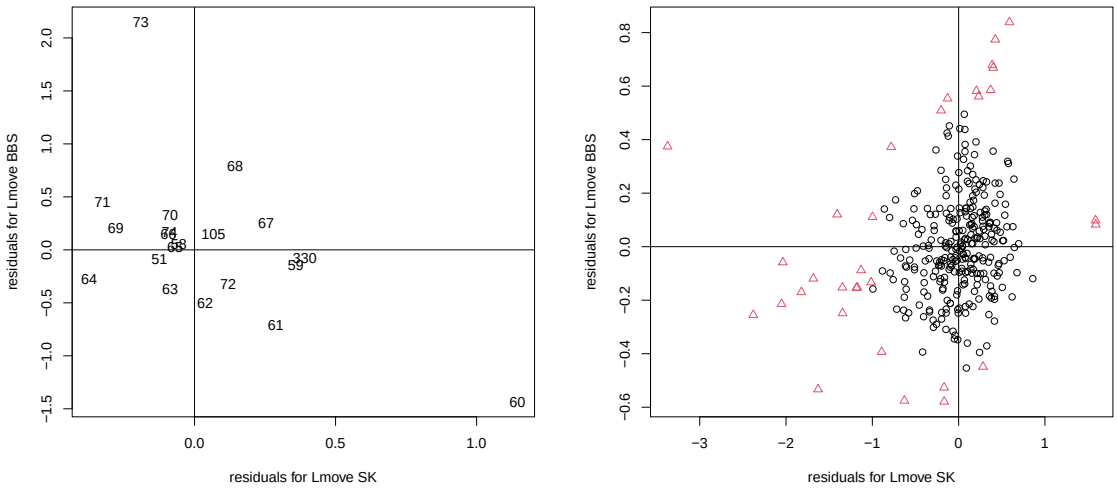

Figure 3.1: Scatterplots of the estimated residuals for the weeks assigned to the first (left) and second (right) clusters detected by the overall best model. Points of the first scatterplot are labelled with the number of the corresponding weeks. Black circle and red triangle in the second scatterplot correspond to typical and outlying weeks, respectively.

different vector of covariates. Models from this class allow for simultaneous robust clustering and detection of mild outliers in multivariate regression analysis. They encompass several other types of Gaussian mixture-based linear regression models previously proposed in the literature, such as the ones illustrated in Galimberti and Soffritti (2020), Mazza and Punzo (2020) and Jones and McLachlan (1992), providing a robust and flexible tool for modelling data in practical applications where different regressors are considered to be relevant for the prediction of different dependent variables. Previous research (see Mazza and Punzo, 2020 and Perrone and Soffritti, 2023) demonstrated that BIC and ICL could be effectively employed to select a proper value for K in the presence of mildly contaminated data. Thanks to an imposition of an eigen-decomposed structure on the K variance-covariance matrices of $\mathbf{Y}|\mathbf{X} = \mathbf{x}$, the presented models are characterised by a reduced number of variance-covariance parameters to be included in the analysis, thus improving flexibility, usefulness and effectiveness of an approach to multivariate linear regression analysis based on finite Gaussian mixture models in real data applications.

Bibliography

Baird IG, Quastel N (2011) Dolphin-safe tuna from California to Thailand: localisms in environmental certification of global commodity networks. Ann Assoc Am Geogr 101(2):337–355

Biernacki C, Celeux G, Govaert G (2000) Assessing a mixture model for clustering with the integrated completed likelihood. IEEE Trans Pattern Anal Mach Intell 22(7):719–725

Cadavez VAP, Hennningsen A (2012) The use of seemingly unrelated regression (SUR) to predict the carcass composition of lambs. Meat Sci 92(4):548–553

Celeux G, Govaert G (1995) Gaussian parsimonious clustering models. Pattern Recognit 28(5):781–793

Chevalier JA, Kashyap AK, Rossi PE (2003) Why don't prices rise during periods of peak demand? Evidence from scanner data. Am Econ Rev 93(1):15–37

Disegna M, Osti L (2016) Tourists' expenditure behaviour: the influence of satisfaction and the dependence of spending categories. Tour Econ 22(1):5–30

Galimberti G, Soffritti G (2020) Seemingly unrelated clusterwise linear regression. Adv Data Anal Classif 14(2):235–260

Jones PN, McLachlan GJ (1992) Fitting finite mixture models in a regression context. Aust J Stat 34(2):233–240

Mazza A, Punzo A (2020) Mixtures of multivariate contaminated normal regression models Stat Pap 61(2):787–822

Meng XL, Rubin DB (1993) Maximum likelihood estimation via the ECM algorithm: A general framework. Biometrika 80(2):267–278

Perrone G, Soffritti S (2023) Seemingly unrelated clusterwise linear regression for contaminated data. Stat Pap 64: 883–921. https://doi.org/10.1007/s00362-022-01344-6

R Core Team (2021) R: a language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org

Ritter G (2015) Robust cluster analysis and variable selection. Chapman and Hall, Boca Raton

Schwarz G (1978) Estimating the dimension of a model. Ann Stat 6(2):461–464

Srivastava VK, Giles DEA (1987) Seemingly unrelated regression equations models. Marcel Dekker, New York

White EN, Hewings GJD (1982) Space-time employment modelling: some results using seemingly unrelated regression estimators. J Reg Sci 22(3):283–302

Chapter 4

Parsimonious seemingly unrelated contaminated normal cluster-weighted models[1]

[1]This chapter coincides with the paper under review: Perrone G., Soffritti S. (2022). "Parsimonious seemingly unrelated contaminated normal cluster-weighted models".

Abstract

Normal cluster-weighted models constitute a modern approach to linear regression which allows to simultaneously perform model-based cluster analysis and multivariate linear regression analysis with random quantitative regressors. Robustified models have been recently developed, based on the use of the contaminated normal distribution, which can manage the presence of mildly atypical observations. A more flexible class of contaminated normal linear cluster-weighted models is specified here, in which the researcher is free to use a different vector of regressors for each response. The novel class also includes parsimonious models, where parsimony is attained by imposing suitable constraints on the component-covariance matrices of either the responses or the regressors. Identifiability conditions are illustrated and discussed. An expectation-conditional maximisation algorithm is provided for the maximum likelihood estimation of the model parameters. The effectiveness and usefulness of the proposed models are shown through the analysis of simulated and real datasets.

Keywords: Contaminated normal distribution, ECM algorithm, Mixture model, Model-based cluster analysis, Parsimonious model, Seemingly unrelated regression

4.1 Introduction

In an era of rapid technological change, vast amounts of complex data are being generated in many fields. Eliciting information from these kinds of data sets represents a crucial challenge faced by scientists and researchers. In order to achieve this aim, advanced and flexible tools and methods are required. From a statistical point of view, problems in learning from data have been classified as either unsupervised or supervised (Hastie et al., 2009). This latter class typically involves the task of modelling the dependence of M responses $\mathbf{Y} = (Y_1, \ldots, Y_M)'$ on P given predictors $\mathbf{X} = (X_1, \ldots, X_P)'$ through multivariate regression techniques. In this setting, several issues can make the data analysis more complex. The novel methods introduced in this chapter have been devised so as to be specifically employed when all the variables in $\mathbf{Y}$ as well as in $\mathbf{X}$ are continuous and the following situations arise.

(I) Data contain measurements obtained without actively controlling or manipulating any of the variables to be analysed. This is typically true in several disciplines (i.e., sociology, economics, business, ecology and geology). For the analysis of such data, regression models should treat both $\mathbf{X}$ and $\mathbf{Y}$ as random vectors. Thus, the joint distribution of $(\mathbf{X}', \mathbf{Y}')'$ in a given population of an investigation, say G, is generally modelled using a probability density function (p.d.f.) $f(\mathbf{x}, \mathbf{y})$ specified so as to take account of the different role played by the responses and predictors in the analysis; that is: $f(\mathbf{x}, \mathbf{y}) = f(\mathbf{x})f(\mathbf{y}|\mathbf{x})$.

(II) The population G is heterogeneous, as it is composed of K disjoint and homogeneous sub-populations, say $G_1, \ldots, G_k, \ldots, G_K$, and the sample data available for the estimation of the regression model are $\mathcal{S} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_I, \mathbf{y}_I)\}$. This means that the information about the specific sub-population each sample observation belongs to is missing. Furthermore, this source of unobserved heterogeneity in the data affects the distribution of $(\mathbf{X}', \mathbf{Y}')'$.

(III) The data $\mathcal{S}$ are contaminated by the presence of mildly atypical observations (Ritter, 2015); that is, observations that in some way deviate from the general pattern of the data (Maronna et al., 2006). In a regression framework, an observation $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{S}$ can be an outlier either in the $\mathbf{y}$-direction (vertical or regression outlier) or in the $\mathbf{x}$-direction (leverage point), depending on whether it occurs in the responses or the predictors, respectively (see, e.g., Rousseeuw and Leroy, 2005). When $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{S}$ is both a regression outlier and a

leverage point it will have a large influence on the estimation of the regression coefficients; thus, it is considered a bad leverage point (Rousseeuw and Leroy, 2005).

(IV) The multivariate regression model specified by the researcher is composed of a system of M regression equations (one equation for each response) with equation-dependent vectors of predictors (i.e., vectors which do not necessarily contain the same predictors for all the responses). This means that certain regressors contained in $\mathbf{X}$ are absent from certain regression equations. This situation is not unusual in economics or social sciences, where different predictors may be expected to be relevant in the prediction of the M responses according to some general theory or prior information about the phenomenon. Furthermore, the M responses contained in $\mathbf{Y}$ are correlated. This latter feature is typically observed with multivariate longitudinal data, time-series data or repeated measures.

An approach able to properly model the distribution of $(\mathbf{X}', \mathbf{Y}')'$ in the presence of the unobserved source of heterogeneity illustrated in situation (II) relies on the cluster-weighted (CW) models (Gershenfeld, 1997). In this approach, the missing information about the memberships to the K sub-populations is modelled using a mixture of K different p.d.f.'s, and each one of these functions is specified by taking account of the different role played by $\mathbf{X}$ and $\mathbf{Y}$. This leads to the following mixture model for the joint distribution of $\mathbf{X}$ and $\mathbf{Y}$:

$$f(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{K} \pi_k f(\mathbf{x}|G_k) f(\mathbf{y}|\mathbf{x}, G_k), \ (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{P+M}, \tag{4.1}$$

where $\pi_1, \ldots, \pi_K$ are positive mixing weights summing to one and representing the prior probabilities of the K sub-populations (i.e., $\mathbb{P}(G_k) = \pi_k$), $f(\mathbf{x}|G_k)$ is the conditional p.d.f. of $\mathbf{X}$ given G_k, and $f(\mathbf{y}|\mathbf{x}, G_k)$ is the conditional p.d.f. of $\mathbf{Y}$ given $\mathbf{x}$ and G_k. An eminent member of the class of CW models for real-valued responses and predictors is the normal CW (NCW hereafter) model (Ingrassia et al., 2012; Dang et al., 2017). In this model, normal distributions are employed for the p.d.f. of both $\mathbf{X}|G_k$ and $\mathbf{Y}|(\mathbf{X} = \mathbf{x}, G_k)$, for $k = 1, \ldots, K$. Thus, equation (4.1) becomes

$$f(\mathbf{x}, \mathbf{y}; \boldsymbol{\vartheta}) = \sum_{k=1}^{K} \pi_k \phi(\mathbf{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \phi(\mathbf{y}|\mathbf{x}; \boldsymbol{\beta}_k' \mathbf{x}^*, \boldsymbol{\Xi}_k), \ (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{P+M}, \tag{4.2}$$

where $\phi(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ represents the p.d.f. of a normal random vector with expected value $\boldsymbol{\mu}$ and positive definite covariance matrix $\boldsymbol{\Sigma}$, $\boldsymbol{\beta}_k' \mathbf{x}^* = \mathbb{E}(\mathbf{Y}|\mathbf{X} = \mathbf{x}, G_k)$, $k = 1, \ldots, K$, $\mathbf{x}^* = (1, \mathbf{x}')'$,

$\boldsymbol{\beta}_k \in \mathbb{R}^{(1+P)\times M}$ is a matrix of intercepts and regression coefficients, and $\boldsymbol{\vartheta} = (\boldsymbol{\vartheta}_1, \ldots, \boldsymbol{\vartheta}_K)$ is the vector of the model parameters, with $\boldsymbol{\vartheta}_k = (\pi_k, \boldsymbol{\vartheta}_{k\mathbf{x}}, \boldsymbol{\vartheta}_{k\mathbf{y}})$, $\boldsymbol{\vartheta}_{k\mathbf{x}} = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, $\boldsymbol{\vartheta}_{k\mathbf{y}} = (\boldsymbol{\beta}_k, \boldsymbol{\Xi}_k)$, $k = 1, \ldots, K$. CW models which allow $\mathbf{X}|G_k$ and $\mathbf{Y}|(\mathbf{X} = \mathbf{x}, G_k)$ to be modelled using skewed distributions have been recently introduced (Gallaugher et al., 2022). A by-product of a regression analysis based on a CW model is a set of estimated posterior probabilities that each sample observation comes from the K sub-populations. Thus, a clustering of the I sample observations that compose $\mathcal{S}$ can also be obtained, based on a rule that assigns an observation to the sub-population from which it has the highest posterior probability of coming. As a result, CW models allow to simultaneously perform multivariate regression and cluster analysis.

Mildly atypical observations in the data mentioned in situation (III) cause departures from the normal distribution. A way to manage these departures is to resort to heavy-tailed models, such as the t distribution or the contaminated normal distribution (see, e.g., Tukey, 1960; Aitkin and Wilson, 1980). This latter distribution is defined as a mixture of two normal distributions having the same expected mean values but different variances-covariances; the normal distribution having the smallest mixing weight also has inflated variances-covariances and is employed to represent the mildly atypical observations. Multivariate regression models robust against the presence of such observations and also suitable for the situations (I) and (II) have been obtained from equation (4.1) by specifying either a t distribution (see, e.g., Ingrassia et al., 2012, 2014) or a contaminated normal distribution (Punzo and McNicholas, 2017) for both $\mathbf{X}|G_k$ and $\mathbf{Y}|(\mathbf{X} = \mathbf{x}, G_k)$, $k = 1, \ldots, K$. The CW models proposed by Punzo and McNicholas (2017) are also called contaminated normal cluster-weighted (CNCW) models. By relying on such models, it is possible able to produce a simultaneous clustering of the sample observations and the detection of both mild outliers and leverage points in a multivariate regression context with random regressors. A limitation of the CNCW models is that the same vector of predictors has to be employed for all the M responses.

Multivariate correlated responses and the systems of regression equations with equation-dependent vectors of predictors illustrated in situation (IV) can be managed by resorting to the so-called seemingly unrelated regression approach (see, e.g., Srivastava and Giles, 1987; Park, 1993). This approach has been recently embedded into the specification of a class of NCW models by Diani et al. (2022), thus leading to seemingly unrelated normal cluster-weighted (SuNCW) models. Thus, the methods based on these latter models are suitable for jointly managing the situations (I), (II) and (IV). However, they are not insensitive to the possible presence of mild

outliers and leverage points in the K sub-populations.

Based on all these considerations, a novel class of multivariate seemingly unrelated contaminated normal cluster-weighted (SuCNCW) models for the analysis of data containing mildly atypical observations either in the distribution of $\mathbf{X}|G_k$ or in the distribution of $\mathbf{Y}|(\mathbf{X}=\mathbf{x}, G_k)$, $k = 1, \ldots, K$, are introduced here. With these novel models, the four situations mentioned above are jointly managed when predicting the responses in a multivariate linear regression framework with random predictors. In particular, SuCNCW models can be considered a more flexible version of the CNCW models described in Punzo and McNicholas (2017), as the linear terms in the M regression equations of a SuCNCW model are defined so that a different vector of regressors can be employed for each dependent variable. In order to keep the total number of parameters as low as possible, the novel class also includes parsimonious SuCNCW models; parsimony is attained by parameterising the covariance matrices of both $\mathbf{X}|G_k$ and $\mathbf{Y}|(\mathbf{X}=\mathbf{x}, G_k)$, for $k = 1, \ldots, K$, with their eigen-decomposition, and by imposing constraints on parts of the elements of this decomposition (see, e.g., Celeux and Govaert, 1995). This leads to a flexible approach for the analysis of linear dependencies in multivariate data.

In summary, this chapter provides the following key contributions:

- new parsimonious SuCNCW models able to jointly manage the situations (I)-(IV) are introduced;
- the relationships between the proposed models and other mixture regression models are described;
- conditions for the identifiability of the SuCNCW models are illustrated;
- maximum likelihood (ML) estimation via an expectation-conditional maximisation (ECM) algorithm (Meng and Rubin, 1993) is detailed;
- strategies for the initialisation and convergence of the ECM algorithm as well as for model selection are presented;
- the effectiveness of the new models in comparison with NCW, CNCW and SuNCW models is investigated through simulated datasets;
- a study of the effects of prices and promotional activities on sales for two U.S. brands of canned tuna is carried out.

The remainder of this chapter is structured as follows. Section 4.2.1 illustrates the specification of the SuCNCW models. A comparison between these models and other mixture regression models is provided in Section 4.2.2. Identifiability conditions are reported in Section 4.2.3. The ECM algorithm for the ML estimation of the model parameters is detailed in Section 4.2.4. Computational details about the ECM algorithm (i.e., initialisation and convergence) are given in Section 4.2.5. Some criteria which can be employed to establish the value of K are summarised in Section 4.2.6. Parsimonious models are introduced in Section 4.2.7. The experimental results obtained from the analysis of simulated data are summarised in Section 4.3. The application to the study of the effects of prices and promotional activities on sales for two U.S. brands of canned tuna is presented in Section 4.4. Finally, concluding remarks and ideas for future research are illustrated in Section 4.5.

4.2 Seemingly unrelated contaminated normal cluster-weighted analysis

4.2.1 Seemingly unrelated contaminated normal cluster-weighted models

The new class of SuCNCW models is introduced starting from the CNCW models illustrated by Punzo and McNicholas (2017). These latter models can be obtained by replacing the normal distributions for $\mathbf{X}|G_k$ and $\mathbf{Y}|(\mathbf{X}=\mathbf{x}, G_k)$ in equation (4.1) with the following contaminated normal distributions, respectively:

$$\begin{aligned} h(\mathbf{x};\boldsymbol{\theta}_{k\mathbf{x}}) &= \alpha_k\phi\left(\mathbf{x};\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\right)+(1-\alpha_k)\phi\left(\mathbf{x};\boldsymbol{\mu}_k,\eta_k\boldsymbol{\Sigma}_k\right),\ \mathbf{x}\in\mathbb{R}^P,\\ h(\mathbf{y}|\mathbf{x};\tilde{\boldsymbol{\vartheta}}_{k\mathbf{y}}) &= \tau_k\phi\left(\mathbf{y}|\mathbf{x};\boldsymbol{\beta}_k,\boldsymbol{\Xi}_k\right)+(1-\tau_k)\phi\left(\mathbf{y}|\mathbf{x};\boldsymbol{\beta}_k,\lambda_k\boldsymbol{\Xi}_k\right),\ \mathbf{y}\in\mathbb{R}^M, \end{aligned}$$

where $\boldsymbol{\theta}_{k\mathbf{x}}=(\boldsymbol{\vartheta}_{k\mathbf{x}},\alpha_k,\eta_k)$, $\tilde{\boldsymbol{\vartheta}}_{k\mathbf{y}}=(\boldsymbol{\vartheta}_{k\mathbf{y}},\tau_k,\lambda_k)$. Parameters $\alpha_k\in(0,1)$ and $\tau_k\in(0,1)$ represent the weights of the typical observations in the $\mathbf{x}$-direction and the $\mathbf{y}$-direction, respectively, within the sub-population G_k. Since in robust statistics it is generally assumed that at least half of the observations are typical (see, e.g., Punzo and McNicholas, 2016, 2017), it is possible to require that $\alpha_k\in[0.5,1)$ and $\tau_k\in[0.5,1)$. Parameters $\eta_k>1$ and $\lambda_k>1$ determine the degree of the contamination in the normal distributions for $\mathbf{X}|G_k$ and $\mathbf{Y}|(\mathbf{X}=\mathbf{x},G_k)$; namely, η_k and λ_k control the increase in variability due to the presence of the leverage points and the mild outliers, respectively, within G_k. Thus, the random vector $(\mathbf{X},\mathbf{Y})$ follows a CNCW model of

order K if its p.d.f. has the form

$$f(\mathbf{x},\mathbf{y};\boldsymbol{\theta}) = \sum_{k=1}^{K} \pi_k h(\mathbf{x};\boldsymbol{\theta}_{k\mathbf{x}}) h(\mathbf{y}|\mathbf{x};\tilde{\boldsymbol{\vartheta}}_{k\mathbf{y}}),\ (\mathbf{x},\mathbf{y}) \in \mathbb{R}^{P+M}, \tag{4.3}$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_K)$, with $\boldsymbol{\theta}_k = (\pi_k, \boldsymbol{\theta}_{k\mathbf{x}}, \tilde{\boldsymbol{\vartheta}}_{k\mathbf{y}})$.

If only P_m of the P covariates ($P_m \leq P$) are known or assumed to be relevant for the prediction of Y_m ($m = 1,\ldots,M$), the linear predictor $\boldsymbol{\beta}_k'\mathbf{x}^*$ employed for modelling the conditional expected value $\mathbb{E}(\mathbf{Y}|\mathbf{X}=\mathbf{x},G_k)$ in equation (4.3) should be modified accordingly. To this end, let $\mathbf{X}_m = (\tilde{X}_1,\ldots,\tilde{X}_{P_m})'$ be the vector composed of such P_m covariates, and let $\boldsymbol{\beta}_{km} = (\beta_{km1},\ldots,\beta_{kmP_m})'$ be the vector of the P_m regression coefficients capturing the linear effect of $\mathbf{X}_m$ on Y_m in the kth sub-population. Furthermore, let $\mathbf{X}_m^* = (1,\mathbf{X}_m')'$ and $\boldsymbol{\beta}_{km}^* = (\beta_{km0},\boldsymbol{\beta}_{km}')'$. Then, $\boldsymbol{\beta}_k^* = (\boldsymbol{\beta}_{k1}^{*\prime},\ldots,\boldsymbol{\beta}_{km}^{*\prime},\ldots,\boldsymbol{\beta}_{kM}^{*\prime})'$ represents the (P^*+M)-dimensional vector containing all the linear effects of the relevant predictors on the M responses in the kth sub-population, where $P^* = \sum_{m=1}^{M} P_m$. Finally, the $(P^*+M)\times M$ design matrix is defined as follows:

$$\tilde{\mathbf{X}}^* = \begin{bmatrix} \mathbf{X}_1^* & \mathbf{0}_{P_1+1} & \ldots & \mathbf{0}_{P_1+1} \\ \mathbf{0}_{P_2+1} & \mathbf{X}_2^* & \ldots & \mathbf{0}_{P_2+1} \\ \vdots & \vdots & & \vdots \\ \mathbf{0}_{P_M+1} & \mathbf{0}_{P_M+1} & \ldots & \mathbf{X}_M^* \end{bmatrix},$$

where $\mathbf{0}_{P_m+1}$ represents the (P_m+1)-dimensional null vector. Using this additional notation, it is possible to obtain the following definition for the conditional expected value of $\mathbf{Y}|\mathbf{X}=\mathbf{x}$ in the kth sub-population:

$$\mathbb{E}(\mathbf{Y}|\mathbf{X}=\mathbf{x},G_k) = \tilde{\mathbf{x}}^{*\prime}\boldsymbol{\beta}_k^* = \begin{bmatrix} \mathbf{x}_1^{*\prime}\ \boldsymbol{\beta}_{k1}^* \\ \vdots \\ \mathbf{x}_m^{*\prime}\ \boldsymbol{\beta}_{km}^* \\ \vdots \\ \mathbf{x}_M^{*\prime}\ \boldsymbol{\beta}_{kM}^* \end{bmatrix}, \tag{4.4}$$

where $\tilde{\mathbf{x}}^*$ is the realisation of the design matrix $\tilde{\mathbf{X}}^*$ obtained when $\mathbf{X}=\mathbf{x}$. The vector defined in equation (4.4) has length M; its mth element is given by a linear combination of the P_m regressors selected by the researcher for the prediction of Y_m whose coefficients are given by

the elements of the vector $\boldsymbol{\beta}_{km}^*$. Thus, inserting the expression given in equation (4.4) into the CNCW model (4.3) leads to the new SuCNCW model. More formally, the random vector $(\mathbf{X}, \mathbf{Y})$ follows a SuCNCW model of order K if its p.d.f. has the form

$$f(\mathbf{x}, \mathbf{y}; \boldsymbol{\psi}) = \sum_{k=1}^{K} \pi_k h\left(\mathbf{x}; \boldsymbol{\theta}_{k\mathbf{x}}\right) h\left(\mathbf{y}|\mathbf{x}; \boldsymbol{\theta}_{k\mathbf{y}}\right), \ (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{P+M}. \tag{4.5}$$

The vector of the model parameters is $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_K)$, with $\boldsymbol{\psi}_k = (\pi_k, \boldsymbol{\theta}_{k\mathbf{x}}, \boldsymbol{\theta}_{k\mathbf{y}})$, $\boldsymbol{\theta}_{k\mathbf{y}} = (\boldsymbol{\beta}_k^*, \boldsymbol{\Xi}_k, \tau_k, \lambda_k)$. From the comparison between $\boldsymbol{\psi}$ and the vector $\boldsymbol{\theta}$ with the parameters of the model (4.3) it is clear that a CNCW model of order K and a SuCNCW model of order K have the same parameters except for the K matrices containing the intercepts and regression coefficients. In model (4.5) it is assumed that $\pi_k > 0$ for $k = 1, \ldots, K$ and $\sum_{k=1}^{K} \pi_k = 1$. As far as the parameters α_k, η_k, τ_k and λ_k are concerned, the requirements coincide with those previously illustrated for the model (4.3). The number of free parameters in model (4.5) is $n_{\boldsymbol{\psi}} = 5K - 1 + K(P + P^* + M) + K[\frac{P(P+1)}{2} + \frac{M(M+1)}{2}]$.

The typical properties of the CNCW model (4.3) (i.e., the ability to determine the membership of an observation $(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{S}$ to a specific sub-population and to establish whether $(\mathbf{x}_i, \mathbf{y}_i)$ is an outlier in the $\mathbf{y}$-direction and/or in the $\mathbf{x}$-direction in that sub-population) are inherited by the SuCNCW model (4.5). In addition, this latter model offers a more parsimonious specification of the linear term to be employed in the prediction of $\mathbf{Y}$ whenever it is known or assumed that certain covariates are not relevant for this task. Model (4.5) can also be considered as a CNCW model in which some regression coefficients are constrained to be a priori equal to zero. To the best of the authors' knowledge, including such constraints in the specification of a multivariate CNCW model has not been addressed yet.

4.2.2 Comparisons with other mixture regression models

When specific conditions are met, some normal CW models can be obtained from model (4.5).

- If $M > 1$, $P_m = P$ and $\mathbf{X}_m = \mathbf{X}$ $\forall m$ (the same vector of covariates is employed in the prediction of the M responses), the realisation of the design matrix $\tilde{\mathbf{X}}^*$ is equal to $\tilde{\mathbf{x}}^* = \mathbf{I}_M \otimes \mathbf{x}^*$, with $\mathbf{I}_M$ being the identity matrix of order M and $\otimes$ denoting the Kronecker product operator (see, e.g., Magnus and Neudecker, 1988). Thus, equation (4.4) becomes

$$\mathbb{E}(\mathbf{Y}|\mathbf{X} = \mathbf{x}, G_k) = \left(\mathbf{I}_M \otimes \mathbf{x}^*\right)' \boldsymbol{\beta}_k^* = \boldsymbol{\beta}_k' \mathbf{x}^*, \ k = 1, \ldots, K, \tag{4.6}$$

where $\boldsymbol{\beta}_k = [\boldsymbol{\beta}^*_{k1} \cdots \boldsymbol{\beta}^*_{km} \cdots \boldsymbol{\beta}^*_{kM}]$. Thus, equation (4.5) reduces to the CNCW model (Punzo and McNicholas, 2017).

- If $M > 1$, $\alpha_k \to 1$, $\eta_k \to 1$, $\tau_k \to 1$ and $\lambda_k \to 1$ $\forall k$ (there is no contamination in the data), the model resulting from equation (4.5) coincides with the SuNCW model described in Diani et al. (2022).

- If $M > 1$, $P_m = P$ and $\mathbf{X}_m = \mathbf{X}$ $\forall m$, $\alpha_k \to 1$, $\eta_k \to 1$, $\tau_k \to 1$ and $\lambda_k \to 1$ $\forall k$ (there is no contamination in the data and the same vector of covariates is employed in the prediction of the M responses), equation (4.5) leads to the multivariate NCW model (4.2) (Dang et al., 2017).

As illustrated in Section 4.2, SuCNCW models assume that $\mathbf{X}|G_k$ follows a contaminated normal distribution with parameters $\boldsymbol{\theta}_{k\mathbf{x}} = (\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \alpha_k, \eta_k)$, for $k = 1, \ldots, K$. However, for some datasets it may happen that the probability a point $(\mathbf{x}, \mathbf{y})$ belongs to one of the K distributions of the mixture (4.5) is the same for all covariate values $\mathbf{x}$. In that case, the assignment of the data points to the sub-populations is independent of the covariates. This condition is known as assignment independence (see, e.g., Hennig, 2000). This implies that the p.d.f of $\mathbf{X}|G_k$ does not depend on G_k, and $h(\mathbf{x}; \boldsymbol{\theta}_{k\mathbf{x}}) = h(\mathbf{x}; \boldsymbol{\theta})$ for every $k = 1, \ldots, K$, where $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\Sigma}, \alpha, \eta)$. Thus, under the assignment independence condition, equation (4.5) becomes

$$f(\mathbf{x}, \mathbf{y}; \boldsymbol{\psi}) = h(\mathbf{x}; \boldsymbol{\theta}) \sum_{k=1}^{K} \pi_k h\left(\mathbf{y}|\mathbf{x}; \boldsymbol{\theta}_{k\mathbf{y}}\right), \ (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{P+M},$$

where

$$f(\mathbf{y}|\mathbf{x}; \tilde{\boldsymbol{\psi}}) = \sum_{k=1}^{K} \pi_k h\left(\mathbf{y}|\mathbf{x}; \boldsymbol{\theta}_{k\mathbf{y}}\right), \ \mathbf{y} \in \mathbb{R}^{M}, \tag{4.7}$$

with $\tilde{\boldsymbol{\psi}} = (\tilde{\boldsymbol{\psi}}_1, \ldots, \tilde{\boldsymbol{\psi}}_K)$, $\tilde{\boldsymbol{\psi}}_k = (\pi_k, \boldsymbol{\theta}_{k\mathbf{y}})$, is the seemingly unrelated contaminated normal clusterwise regression model described in Perrone and Soffritti (2023). As a consequence, when in model (4.5) the following conditions hold true: $\boldsymbol{\mu}_k = \boldsymbol{\mu}$, $\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$, $\alpha_k = \alpha$ and $\eta_k = \eta$ for $k = 1, \ldots, K$, then the task of extracting the information about both the K disjoint sub-populations that compose the population G and the distinction between typical observations and mild outliers in the $\mathbf{y}$-direction within each sub-population can be equivalently carried out using either the conditional p.d.f. $f(\mathbf{y}|\mathbf{x}; \tilde{\boldsymbol{\psi}})$ through seemingly unrelated contaminated normal

clusterwise models or the joint p.d.f. $f(\mathbf{x}, \mathbf{y}; \boldsymbol{\psi})$ through SuCNCW models.

4.2.3 Identifiability

Since identifiability represents a regularity condition for the asymptotic theory to hold for the ML estimator, a discussion about identifiability of model (4.5) is provided here. In particular, this discussion focuses on the class of models $\mathfrak{F} = \{\mathfrak{F}_K, K = 1, \ldots, K_{max}\}$, with $\mathfrak{F}_K = \{f(\mathbf{x}, \mathbf{y}; \boldsymbol{\psi}), \boldsymbol{\psi} \in \boldsymbol{\Psi}\}$, where $f(\mathbf{x}, \mathbf{y}; \boldsymbol{\psi})$ is the p.d.f. of $(\mathbf{X}', \mathbf{Y}')'$ under the SuCNCW model of order K defined in (4.5) and K_{max} denotes the maximum order specified by the researcher for that model. This class is identifiable if, for any two members M, $\tilde{M} \in \mathfrak{F}$ with parameters $\boldsymbol{\psi} = (\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_k, \ldots, \boldsymbol{\psi}_K)$ and $\tilde{\boldsymbol{\psi}} = (\tilde{\boldsymbol{\psi}}_1, \ldots, \tilde{\boldsymbol{\psi}}_s, \ldots, \tilde{\boldsymbol{\psi}}_{\tilde{K}})$, respectively, the equality

$$\sum_{k=1}^{K} \pi_k h\left(\mathbf{x}; \boldsymbol{\theta}_{k\mathbf{x}}\right) h\left(\mathbf{y}|\mathbf{x}; \boldsymbol{\theta}_{k\mathbf{y}}\right) = \sum_{s=1}^{\tilde{K}} \tilde{\pi}_s h\left(\mathbf{x}; \tilde{\boldsymbol{\theta}}_{s\mathbf{x}}\right) h\left(\mathbf{y}|\mathbf{x}; \tilde{\boldsymbol{\theta}}_{s\mathbf{y}}\right) \;\; \forall \; (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{P+M}$$

implies that $K = \tilde{K}$ and for each $k \in \{1, \ldots, K\}$ there exists $s \in \{1, \ldots, K\}$ such that $\pi_k = \tilde{\pi}_s$, $\boldsymbol{\theta}_{k\mathbf{x}} = \tilde{\boldsymbol{\theta}}_{s\mathbf{x}}$ and $\boldsymbol{\theta}_{k\mathbf{y}} = \tilde{\boldsymbol{\theta}}_{s\mathbf{y}}$.

The model class $\mathfrak{F}$ is affected by several sources of non-identifiability. As any finite mixture model, also model (4.5) is invariant under relabelling the K distributions of the mixture (label switching). Another source is represented by potential overfitting associated with empty components or equal components of the mixture (see, e.g., Frühwirth-Schnatter, 2006, for further details). In order to prevent such sources of non-identifiability for $\mathfrak{F}$, some constraints have been imposed on the parameter space $\boldsymbol{\Psi}$. They have been obtained by suitably modifying the constraints described in Punzo and McNicholas (2017) for ensuring the identifiability of CNCW models. Namely, for the model (4.5), it is required that $\pi_k > 0 \; \forall \; k$ and $(\boldsymbol{\beta}_k^*, \boldsymbol{\Xi}_k) \neq (\boldsymbol{\beta}_h^*, \boldsymbol{\Xi}_h) \; \forall \; k \neq h$. Thanks to these constraints, the two sources of non-identifiability due to empty components and equal components can be avoided. Thus, in order to ensure identifiability, the following restricted class of SuCNCW models is introduced:

$$\bar{\mathfrak{F}} = \big\{ f(\mathbf{x}, \mathbf{y}; \bar{\boldsymbol{\psi}}) : f(\mathbf{x}, \mathbf{y}; \bar{\boldsymbol{\psi}}) = \sum_{k=1}^{K} \pi_k h\left(\mathbf{x}; \boldsymbol{\theta}_{k\mathbf{x}}\right) h\left(\mathbf{y}|\mathbf{x}; \boldsymbol{\theta}_{k\mathbf{y}}\right), \; (\mathbf{x}, \mathbf{y}) \in \mathbb{R}^{P+M}, \bar{\boldsymbol{\psi}} \in \bar{\boldsymbol{\Psi}}, K \in \mathbb{N} \big\},$$

where $\bar{\boldsymbol{\Psi}}$ is the following constrained parameter space:

$$\bar{\boldsymbol{\Psi}} = \left\{ \bar{\boldsymbol{\psi}} \in \boldsymbol{\Psi} : \pi_k > 0 \; \forall \; k, \sum_{k=1}^{K} \pi_k = 1, (\boldsymbol{\beta}_k^*, \boldsymbol{\Xi}_k) \neq (\boldsymbol{\beta}_h^*, \boldsymbol{\Xi}_h) \; \forall \; k \neq h \right\}.$$

For the identifiability of the class $\bar{\mathfrak{F}}$ it is also required that there exists a set $\mathcal{X} \subseteq \mathbb{R}^P$ having probability equal to one according to the P-dimensional contaminated normal distribution such that the following mixture of contaminated normal regression models

$$\sum_{k=1}^{K} \pi_k(\mathbf{x}) h\left(\mathbf{y}|\mathbf{x}; \tilde{\mathbf{x}}^{*\prime} \boldsymbol{\beta}_k^*, \boldsymbol{\Xi}_k\right), \; \mathbf{y} \in \mathbb{R}^M,$$

is identifiable for each fixed $\mathbf{x} \in \mathcal{X}$, where $\pi_1(\mathbf{x}), ..., \pi_K(\mathbf{x})$ are positive weights summing to one for each $\mathbf{x} \in \mathcal{W}$. Then, it is possible to prove that the class $\bar{\mathfrak{F}}$ is identifiable in $\mathcal{X} \times \mathbb{R}^M$. Such a proof can be easily obtained by exploiting the same arguments described in Punzo and McNicholas (2017, Appendix B) for the identifiability of CNCW models with the following modifications: *(i)* the linear term to be considered in the conditional expected value of $\mathbf{Y}|(\mathbf{X} = \mathbf{x}, G_k)$ is $\tilde{\mathbf{x}}^{*\prime} \boldsymbol{\beta}_k^*$; *(ii)* the set of all covariate points to be employed to distinct different regression coefficients $\boldsymbol{\beta}_k^*$ by different values of $\tilde{\mathbf{x}}^{*\prime} \boldsymbol{\beta}_k^*$ is:

$$\begin{aligned} \mathcal{X} \; = \; & \Big\{ \mathbf{x} \in \mathbb{R}^P : \forall \; \mathbf{x}_m \in \{\mathbf{x}_1, \ldots, \mathbf{x}_M\}, \forall \; k, h \in \{1, \ldots, K\} \text{ and } s, t \in \{1, \ldots, \tilde{K}\} \; , \\ & \tilde{\mathbf{x}}_m^{*\prime} \boldsymbol{\beta}_{km}^* = \tilde{\mathbf{x}}_m^{*\prime} \boldsymbol{\beta}_{hm}^* \Rightarrow \boldsymbol{\beta}_{km}^* = \boldsymbol{\beta}_{hm}^*, \; \tilde{\mathbf{x}}_m^{*\prime} \boldsymbol{\beta}_{km}^* = \tilde{\mathbf{x}}_m^{*\prime} \tilde{\boldsymbol{\beta}}_{sm}^* \Rightarrow \boldsymbol{\beta}_{km}^* = \tilde{\boldsymbol{\beta}}_{sm}^* \; , \\ & \tilde{\mathbf{x}}_m^{*\prime} \tilde{\boldsymbol{\beta}}_{sm}^* = \tilde{\mathbf{x}}_m^{*\prime} \tilde{\boldsymbol{\beta}}_{tm}^* \Rightarrow \tilde{\boldsymbol{\beta}}_{sm}^* = \tilde{\boldsymbol{\beta}}_{tm}^* \Big\} . \end{aligned}$$

4.2.4 An ECM algorithm for ML estimation

Let $\mathcal{S} = \{(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_I, \mathbf{y}_I)\}$ be a sample of I independent observations drawn from model (4.5). Under these conditions, the log-likelihood function can be written as

$$l(\boldsymbol{\psi}) = \sum_{i=1}^{I} \ln \left(\sum_{k=1}^{K} \pi_k h\left(\mathbf{x}_i; \boldsymbol{\theta}_{k\mathbf{x}}\right) h\left(\mathbf{y}_i|\mathbf{x}_i; \boldsymbol{\theta}_{k\mathbf{y}}\right) \right).$$

Similarly to other finite mixture models and following Punzo and McNicholas (2017), ML estimation of $\boldsymbol{\psi}$ has been carried out for a fixed value of K under a general framework dealing with incomplete-data problems (Dempster et al., 1977; Meng and Rubin, 1993). In the considered situation, there are three different types of incompleteness in the data $\mathcal{S}$: *(i)* the missing information about the specific sub-populations from which the I sample observations come from;

(ii) the missing information about whether such observations are leverage points with reference to any given G_k or not; *(iii)* the missing information about whether each observation is an outlier with reference to any given G_k or not. The first type is typical of any finite mixture model; the second and third types are specific for model (4.5). Such information can be described using three different types of K-dimensional vectors. For the ith sample observation, they are given by $\mathbf{z}_i$, $\mathbf{v}_i$, $\mathbf{u}_i$. Namely, $\mathbf{z}_i = (z_{i1}, \ldots, z_{iK})'$, with $z_{ik} = 1$ if the ith observation comes from the sub-population G_k and $z_{ik} = 0$ otherwise, for $k = 1, \ldots, K$; $\mathbf{v}_i = (v_{i1}, \ldots, v_{iK})'$, with $v_{ik} = 1$ if the ith observation is not a leverage point within the sub-population G_k and $v_{ik} = 0$ if it is a leverage point; $\mathbf{u}_i = (u_{i1}, \ldots, u_{iK})'$, with $u_{ik} = 1$ if the ith observation is typical within the sub-population G_k and $u_{ik} = 0$ if it is an outlier. Thus, the complete data would be $\mathcal{S}_c = \{(\mathbf{x}_1, \mathbf{y}_1, \mathbf{z}_1, \mathbf{v}_1, \mathbf{u}_1), \ldots, (\mathbf{x}_I, \mathbf{y}_I, \mathbf{z}_I, \mathbf{v}_I, \mathbf{u}_I)\}$. Then, following Punzo and McNicholas (2017), to find the ML estimates $\hat{\boldsymbol{\psi}}$, an ECM algorithm (Meng and Rubin, 1993) has been developed. To this end, the complete-data likelihood function has been derived:

$$L_c(\boldsymbol{\psi}) = \prod_{i=1}^{I}\prod_{k=1}^{K}\Big\{\pi_k\Big[\alpha_k\phi_P\Big(\mathbf{x}_i;\boldsymbol{\mu}_k,\boldsymbol{\Sigma}_k\Big)\Big]^{v_{ik}}\Big[(1-\alpha_k)\phi_P\Big(\mathbf{x}_i;\boldsymbol{\mu}_k,\eta_k\boldsymbol{\Sigma}_k\Big)\Big]^{1-v_{ik}}$$
$$\Big[\tau_k\phi_M\Big(\mathbf{y}_i;\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*,\boldsymbol{\Xi}_k\Big)\Big]^{u_{ik}}\Big[(1-\tau_k)\phi_M\Big(\mathbf{y}_i;\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*,\lambda_k\boldsymbol{\Xi}_k\Big)\Big]^{1-u_{ik}}\Big\}^{z_{ik}};$$

thus, up to an additive constant, the complete-data log-likelihood function employed in the ECM algorithm for the computation of $\hat{\boldsymbol{\psi}}$ is equal to:

$$\begin{aligned}\ell_c(\boldsymbol{\psi}) = \sum_{i=1}^{I}\sum_{k=1}^{K} z_{ik}\Big[&\ln\pi_k + v_{ik}\ln\alpha_k + (1-v_{ik})\ln(1-\alpha_k) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_k| + \\ &-\Big(\frac{P}{2}\ln\eta_k\Big)(1-v_{ik}) - \frac{1}{2}\Big(v_{ik} + \frac{1-v_{ik}}{\eta_k}\Big)\delta^2_{\boldsymbol{\Sigma}_k}\Big(\mathbf{x}_i,\boldsymbol{\mu}_k\Big) + \\ &+ u_{ik}\ln\tau_k + (1-u_{ik})\ln(1-\tau_k) - \frac{1}{2}\ln|\boldsymbol{\Xi}_k| + \\ &-\Big(\frac{M}{2}\ln\lambda_k\Big)(1-u_{ik}) - \frac{1}{2}\Big(u_{ik} + \frac{1-u_{ik}}{\lambda_k}\Big)\delta^2_{\boldsymbol{\Xi}_k}\Big(\mathbf{y}_i,\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*\Big)\Big],\end{aligned}$$

where

$$\delta^2_{\boldsymbol{\Sigma}_k}(\mathbf{x}_i,\boldsymbol{\mu}_k) = (\mathbf{x}_i-\boldsymbol{\mu}_k)'\boldsymbol{\Sigma}_k^{-1}(\mathbf{x}_i-\boldsymbol{\mu}_k), \tag{4.8}$$

$$\delta^2_{\boldsymbol{\Xi}_k}\big(\mathbf{y}_i,\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*\big) = (\mathbf{y}_i-\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*)'\boldsymbol{\Xi}_k^{-1}(\mathbf{y}_i-\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*) \tag{4.9}$$

are squared Mahalanobis distances: the first is computed between $\mathbf{x}_i$ and $\boldsymbol{\mu}_k$ with respect to $\boldsymbol{\Sigma}_k$; the second is computed between $\mathbf{y}_i$ and $\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*$ with respect to $\boldsymbol{\Xi}_k$.

The ECM algorithm consists in an iterative sequence. At each iteration, an E-step is followed by two CM-steps. The first CM-step focuses on the parameter sub-vector $\boldsymbol{\psi}_a = (\boldsymbol{\pi}, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\alpha}, \boldsymbol{\beta}^*, \boldsymbol{\Xi}, \boldsymbol{\tau})$, where $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)$, $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_K)$, $\boldsymbol{\Sigma} = (\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_K)$, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_K)$, $\boldsymbol{\beta}^* = (\boldsymbol{\beta}_1^*, \ldots, \boldsymbol{\beta}_K^*)$, $\boldsymbol{\Xi} = (\boldsymbol{\Xi}_1, \ldots, \boldsymbol{\Xi}_K)$, $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_K)$. The second CM-step involves the parameter sub-vector $\boldsymbol{\psi}_b = (\boldsymbol{\eta}, \boldsymbol{\lambda})$, where $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_K)$, $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_K)$. Iterations are repeated until convergence.

- On the hth iteration of the E-step, given the current estimate $\boldsymbol{\psi}^{(h)}$ of the model parameters $\boldsymbol{\psi}$, the conditional expectation of $l_c(\boldsymbol{\psi})$ has to be computed; up to an additive constant, it is equal to:

$$
\begin{aligned}
Q\left(\boldsymbol{\psi}|\boldsymbol{\psi}^{(h)}\right) &= \mathbb{E}_{\boldsymbol{\psi}^{(h)}}[\ell_c(\boldsymbol{\psi})] \\
&= \sum_{i=1}^{I}\sum_{k=1}^{K} \hat{z}_{ik}^{(h)} \Big\{ \ln \pi_k^{(h)} + \hat{v}_{ik}^{(h)} \ln \alpha_k^{(h)} + (1-\hat{v}_{ik}^{(h)}) \ln(1-\alpha_k^{(h)}) + \\
&+ Q_{i1}\Big(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k|\boldsymbol{\psi}^{(h)}\Big) + \hat{u}_{ik}^{(h)} \ln \tau_k^{(h)} + (1-\hat{u}_{ik}^{(h)}) \ln(1-\tau_k^{(h)}) + \\
&+ Q_{i2}\Big(\boldsymbol{\beta}_k^*, \boldsymbol{\Xi}_k|\boldsymbol{\psi}^{(h)}\Big)\Big\},
\end{aligned}
$$

where

$$
\begin{aligned}
Q_{i1}\Big(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k|\boldsymbol{\psi}^{(h)}\Big) &= -\frac{1}{2}\Big[\ln|\boldsymbol{\Sigma}_k^{(h)}| + P(1-\hat{v}_{ik}^{(h)}) \ln \eta_k^{(h)} + \\
&+ \Big(\hat{v}_{ik}^{(h)} + \frac{1-\hat{v}_{ik}^{(h)}}{\eta_k^{(h)}}\Big) \delta^2_{\boldsymbol{\Sigma}_k^{(h)}}\Big(\mathbf{x}_i, \boldsymbol{\mu}_k^{(h)}\Big)\Big], \\
Q_{i2}\Big(\boldsymbol{\beta}_k^*, \boldsymbol{\Xi}_k|\boldsymbol{\psi}^{(h)}\Big) &= -\frac{1}{2}\Big[\ln|\boldsymbol{\Xi}_k^{(h)}| + M(1-\hat{u}_{ik}^{(h)}) \ln \lambda_k^{(h)} + \\
&+ \Big(\hat{u}_{ik}^{(h)} + \frac{1-\hat{u}_{ik}^{(h)}}{\lambda_k^{(h)}}\Big) \delta^2_{\boldsymbol{\Xi}_k^{(h)}}\Big(\mathbf{y}_i, \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^{*(h)}\Big)\Big],
\end{aligned}
$$

$$\hat{z}_{ik}^{(h)} = \mathbb{E}_{\boldsymbol{\psi}^{(h)}}[Z_{ik}|(\mathbf{x}_i, \mathbf{y}_i)] = \frac{\pi_k^{(h)} h\left(\mathbf{x}_i; \boldsymbol{\theta}_{k\mathbf{x}}^{(h)}\right) h\left(\mathbf{y}_i; \boldsymbol{\theta}_{k\mathbf{y}}^{(h)}\right)}{f\left(\mathbf{x}_i, \mathbf{y}_i; \boldsymbol{\psi}^{(h)}\right)}, \tag{4.10}$$

$$\hat{v}_{ik}^{(h)} = \mathbb{E}_{\boldsymbol{\psi}^{(h)}}[V_{ik}|(\mathbf{x}_i, \mathbf{z}_i)] = \frac{\alpha_k^{(h)} \phi\left(\mathbf{x}_i; \boldsymbol{\mu}_k^{(h)}, \boldsymbol{\Sigma}_k^{(h)}\right)}{h\left(\mathbf{x}_i; \boldsymbol{\theta}_{k\mathbf{x}}^{(h)}\right)}, \tag{4.11}$$

$$\hat{u}_{ik}^{(h)} = \mathbb{E}_{\boldsymbol{\psi}^{(h)}}[U_{ik}|(\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)] = \frac{\tau_k^{(h)} \phi\left(\mathbf{y}_i; \tilde{\mathbf{x}}_i^{*\prime} \boldsymbol{\beta}_k^{*(h)}, \boldsymbol{\Xi}_k^{(h)}\right)}{h\left(\mathbf{y}_i; \boldsymbol{\theta}_{k\mathbf{y}}^{(h)}\right)}, \tag{4.12}$$

with $\mathbf{Z}_i = (Z_{i1}, \ldots, Z_{iK})'$ denoting a K-dimensional multinomial random vector with probabilities $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_K)'$, $V_{ik}|Z_{ik} = 1$ and $U_{ik}|Z_{ik} = 1$ being two Bernoulli random variables with success probability of α_k and τ_k, respectively, for $i = 1, \ldots, I$ and $k = 1, \ldots, K$. Thus, $\hat{z}_{ik}^{(h)}$, $\hat{v}_{ik}^{(h)}$ and $\hat{u}_{ik}^{(h)}$ represent posterior probabilities (evaluated using $\boldsymbol{\psi}^{(h)}$) of the following three events: *(i)* the sample observation $(\mathbf{x}_i, \mathbf{y}_i)$ comes from the kth distribution of the mixture (4.5); *(ii)* $(\mathbf{x}_i, \mathbf{y}_i)$ is not a leverage point within such a distribution; *(iii)* $(\mathbf{x}_i, \mathbf{y}_i)$ is not an outlier within such a distribution.

- At the first CM-step on the $(h+1)$th iteration of the ECM algorithm, the sub-vector $\boldsymbol{\psi}_a^{(h)}$ is updated through the maximisation of $Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(h)})$ with respect to $\boldsymbol{\psi}_a$ with $\boldsymbol{\psi}_b$ fixed at $\boldsymbol{\psi}_b^{(h)}$. The resulting updates of $\pi_k^{(h)}$, $\alpha_k^{(h)}$, $\tau_k^{(h)}$, $\boldsymbol{\mu}_k^{(h)}$ and $\boldsymbol{\Sigma}_k^{(h)}$ are:

$$\begin{aligned}
\pi_k^{(h+1)} &= \frac{1}{I}\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}, \\
\alpha_k^{(h+1)} &= \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{v}_{ik}^{(h)}}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}}, && (4.13) \\
\tau_k^{(h+1)} &= \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{u}_{ik}^{(h)}}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}}, && (4.14) \\
\boldsymbol{\mu}_k^{(h+1)} &= \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{1_{ik}}^{(h)} \mathbf{x}_i}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{1_{ik}}^{(h)}}, && (4.15) \\
\boldsymbol{\Sigma}_k^{(h+1)} &= \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{1_{ik}}^{(h)} \left(\mathbf{x}_i - \boldsymbol{\mu}_k^{(h+1)}\right)\left(\mathbf{x}_i - \boldsymbol{\mu}_k^{(h+1)}\right)'}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}}, && (4.16)
\end{aligned}$$

where

$$\hat{w}_{1_{ik}}^{(h)} = \hat{v}_{ik}^{(h)} + \frac{1 - \hat{v}_{ik}^{(h)}}{\eta_k^{(h)}}. \tag{4.17}$$

Such updates coincide with the solutions obtained for the CNCW model (for further details

see Punzo and McNicholas, 2017, Appendices C.1-C.4). As far as the remaining elements of the sub-vector $\boldsymbol{\psi}_a^{(h)}$ are concerned, their updates are:

$$\boldsymbol{\beta}_k^{*(h+1)} = \Big(\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{2_{ik}}^{(h)} \tilde{\mathbf{x}}_i^* \boldsymbol{\Xi}_k^{(h)^{-1}} \tilde{\mathbf{x}}_i^{*\prime} \Big)^{-1} \Big(\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{2_{ik}}^{(h)} \tilde{\mathbf{x}}_i^* \boldsymbol{\Xi}_k^{(h)^{-1}} \boldsymbol{y}_i \Big), \tag{4.18}$$

$$\boldsymbol{\Xi}_k^{(h+1)} = \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{2_{ik}}^{(h)} \Big(\mathbf{y}_i - \tilde{\mathbf{x}}_i^{*\prime} \boldsymbol{\beta}_k^{*(h+1)} \Big) \Big(\mathbf{y}_i - \tilde{\mathbf{x}}_i^{*\prime} \boldsymbol{\beta}_k^{*(h+1)} \Big)'}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}}, \tag{4.19}$$

where

$$\hat{w}_{2_{ik}}^{(h)} = \hat{u}_{ik}^{(h)} + \frac{1 - \hat{u}_{ik}^{(h)}}{\lambda_k^{(h)}}. \tag{4.20}$$

The updates illustrated in equations (4.18)-(4.19) coincide with the ones obtained for the seemingly unrelated contaminated normal clusterwise regression models (4.7) (further details can be found in Perrone and Soffritti, 2023, Appendix A).

- At the second CM-step on the $(h+1)$th iteration of the ECM algorithm, the update of $\boldsymbol{\psi}_b^{(h)}$ is obtained by maximising $Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(h)})$ with respect to $\boldsymbol{\psi}_b$ with $\boldsymbol{\psi}_a$ fixed at $\boldsymbol{\psi}_a^{(h+1)}$. The resulting updates of $\eta_k^{(h)}$ and $\lambda_k^{(h)}$ are (further details can be found in Punzo et al., 2018):

$$\eta_k^{(h+1)} = \max\Big\{1, \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} (1 - \hat{v}_{ik}^{(h)}) \delta^2_{\boldsymbol{\Sigma}_k^{(h+1)}} \Big(\mathbf{x}_i, \boldsymbol{\mu}_k^{(h+1)} \Big)}{P \sum_{i=1}^{I} \hat{z}_{ik}^{(h)} (1 - \hat{v}_{ik}^{(h)})} \Big\}, \tag{4.21}$$

$$\lambda_k^{(h+1)} = \max\Big\{1, \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} (1 - \hat{u}_{ik}^{(h)}) \delta^2_{\boldsymbol{\Xi}_k^{(h+1)}} \Big(\mathbf{y}_i, \tilde{\mathbf{x}}_i^{*\prime} \boldsymbol{\beta}_k^{*(h+1)} \Big)}{M \sum_{i=1}^{I} \hat{z}_{ik}^{(h)} (1 - \hat{u}_{ik}^{(h)})} \Big\}. \tag{4.22}$$

It is worth noting that the update $\boldsymbol{\beta}_k^{*(h+1)}$ can be computed only if the matrix $\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{w}_{2_{ik}}^{(h)} \tilde{\mathbf{x}}_i^* \boldsymbol{\Xi}_k^{(h)^{-1}} \tilde{\mathbf{x}}_i^{*\prime}$ in equation (4.18) is nonsingular. This equation also shows that the update $\boldsymbol{\beta}_k^{*(h+1)}$ can be seen as a generalised least squares estimate with weights depending on $\hat{z}_{ik}^{(h)}$ and $\hat{w}_{2_{ik}}^{(h)}$; such weights also affect the update $\boldsymbol{\Xi}_k^{(h+1)}$ in (4.19), which represents a weighted sum of squared residuals. As a consequence, sample observations with the highest posterior estimated probabilities of being generated from the kth distribution of the mixture (4.5) and of representing typical points in $\mathbf{y}$-direction within that distribution will have the largest impact on the updates of both the regression coefficients and covariances of $\mathbf{Y}|(\mathbf{X} = \mathbf{x}, G_k)$. For this reason, this approach provides robust estimates of $\boldsymbol{\beta}_k^{*(h+1)}$ and $\boldsymbol{\Xi}_k^{(h+1)}$ for $k = 1, \ldots K$. In a similar way,

the term $\hat{w}_{1_{ik}}^{(h)}$ in equations (4.15) and (4.16) allows to reduce the impact of the leverage points on the estimation of $\boldsymbol{\mu}_k^{(h+1)}$ and $\boldsymbol{\Sigma}_k^{(h+1)}$, thereby proving to represent a robust solution also for the estimation of these latter parameters. Furthermore, equations (4.21) and (4.22) show that the updates $\eta_k^{(h+1)}$ and $\lambda_k^{(h+1)}$ will be larger when the kth distribution of the mixture in the model (4.5) is highly contaminated by the presence of outliers and leverage points, respectively (i.e., when many observations show small values of $\hat{v}_{ik}^{(h)}$ and $\hat{u}_{ik}^{(h)}$ or, equivalently, large squared Mahalanobis distances from $\boldsymbol{\mu}_k^{(h+1)}$ and $\tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^{*(h+1)}$).

The main result of the ECM algorithm is represented by the ML estimate $\hat{\boldsymbol{\psi}}$, that is the value of $\boldsymbol{\psi}^{(h)}$ at convergence. As a by-product, by exploiting equations (4.10)-(4.12) this algorithm also provides estimates of the following posterior probabilities: $\mathbb{P}_{\hat{\boldsymbol{\psi}}}[Z_{ik} = 1|(\mathbf{x}_i, \mathbf{y}_i)] = \hat{z}_{ik}$, $\mathbb{P}_{\hat{\boldsymbol{\psi}}}[V_{ik} = 1|(\mathbf{x}_i, \hat{\mathbf{z}}_i)] = \hat{v}_{ik}$ and $\mathbb{P}_{\hat{\boldsymbol{\psi}}}[U_{ik} = 1|(\mathbf{x}_i, \mathbf{y}_i, \hat{\mathbf{z}}_i)] = \hat{u}_{ik}$, for $i = 1, \ldots, I$ and $k = 1, \ldots, K$. Then, the I sample observations can be partitioned into K clusters according to the rule of the maximum a posteriori probability; for the ith observation:

$$\text{MAP}(\hat{z}_{ik}) = \begin{cases} 1 & \text{if } \max_h\{\hat{z}_{ih}\} \text{ occurs when } h = k; \\ 0 & \text{otherwise.} \end{cases}$$

Furthermore, the estimates $\hat{v}_{ik}$ and $\hat{u}_{ik}$ can be employed to define two intra-cluster distinctions. Namely, if $\hat{v}_{ih} < 0.5$, where h is the label of the cluster for which $\text{MAP}(\hat{z}_{ik}) = 1$, the ith observation will be classified as a leverage point for that cluster; in a similar way, if $\hat{u}_{ih} < 0.5$, the ith observation will be classified as a mild outlier for the same cluster. The ML estimates can also be exploited in conjunction with equations (4.8) and (4.9) to compute the estimated squared Mahalanobis distances $\hat{d}_{ik\mathbf{x}}^2 = \delta_{\hat{\boldsymbol{\Sigma}}_k}^2(\mathbf{x}_i, \hat{\boldsymbol{\mu}}_k)$ and $\hat{d}_{ik\mathbf{y}}^2 = \delta_{\hat{\boldsymbol{\Xi}}_k}^2\left(\mathbf{y}_i, \tilde{\mathbf{x}}_i^{*\prime}\hat{\boldsymbol{\beta}}_k^*\right)$, for $i = 1, \ldots, I$ and $k = 1, \ldots, K$, which can be interpreted as intra-cluster quantifications of the amount of deviations from the pattern of the observations assigned to any given cluster. Thus, a more detailed analysis of the leverage points and mild outliers could be carried out by considering the values of $\hat{d}_{ik\mathbf{x}}^2$ and $\hat{d}_{ik\mathbf{y}}^2$ $\forall(i,k) \in \{i \in \{1, \ldots, I\}, k : \text{MAP}(\hat{z}_{ik}) = 1\}$ and by focusing on the largest values obtained in this way (see McLachlan and Peel, 2000, p. 232).

4.2.5 Technical details about the ECM algorithm

Generally speaking, in finite mixture modelling, the parameter estimates resulting from an EM-based algorithm are dependent on the values employed to initialise the iterative process. Thus,

the quality of the solution can be largely affected by the choice of the starting value for the model parameters. As this is true also for model (4.5), appropriately choosing $\boldsymbol{\psi}^{(0)}$ is paramount for obtaining a proper ML estimation of $\boldsymbol{\psi}$. To this end, strategies usually employed in finite mixture models (e.g., multiple executions of the algorithm using multiple random initialisations, approaches based on non-random choices of either $\boldsymbol{\psi}^{(0)}$ or the missing information) could be adopted (see, e.g., Biernacki et al., 2003; Karlis and Xekalaki, 2003, for more details). More specific initialisation strategies could be devised by resorting to the normal mixture model of order K for $(\mathbf{X}, \mathbf{Y})$. This latter model has been proved to represent a reparameterisation of the NCW model (4.2) (see Ingrassia et al., 2012, for more details) which, in turn, is nested in the CNCW model (4.3) when $\alpha_k \to 1^-$, $\tau_k \to 1^-$, $\eta_k \to 1^+$ and $\lambda_k \to 1^+$, $k = 1, \ldots, K$. Thus, a first strategy for choosing the initial values $\hat{z}_{ik}^{(0)}$, $i = 1, \ldots, I$, $k = 1, \ldots, K$, could set such quantities equal to the estimated posterior probabilities of the normal mixture model of order K for $(\mathbf{X}, \mathbf{Y})$. Furthermore, $\hat{v}_{ik}^{(0)}$ and $\hat{u}_{ik}^{(0)}$ could be set equal to 0.999 for $i = 1, \ldots, I$ and $k = 1, \ldots, K$. In the analyses reported in Sections 4.3 and 4.4, the ECM algorithm has been initialised using a strategy composed of the following three steps. Firstly, the normal mixture model of order K for $(\mathbf{X}, \mathbf{Y})$ is estimated using the data $\mathcal{S}$. The resulting estimates of the mixing weights, the expected values and the variances-covariances of $\mathbf{X}$ are employed to obtain the starting values $\pi_k^{(0)}$, $\boldsymbol{\mu}_k^{(0)}$ and $\boldsymbol{\Sigma}_k^{(0)}$. Secondly, a seemingly unrelated linear regression model for $\mathbb{E}(\mathbf{Y}|\mathbf{X}_m = \mathbf{x}_m)$ is fitted to subsample of $\mathcal{S}$ composed of the observations assigned to the kth cluster detected by the normal mixture model considered in the previous step ($k = 1, \ldots, K$). The starting values $\boldsymbol{\beta}_k^{*(0)}$ and $\boldsymbol{\Xi}_k^{(0)}$ are given by the vector containing the estimated intercept and regression coefficients and the matrix with the variances and covariances of the sample residuals, respectively. Thirdly, $\alpha_k^{(0)}$ and $\tau_k^{(0)}$, for $k = 1 \ldots, K$, are set equal to 0.999; $\eta_k^{(0)}$ and $\lambda_k^{(0)}$ are set equal to 1.001. The packages `mclust` (Scrucca et al., 2017) and `systemfit` (Henningsen and Hamann, 2007) in the `R` environment (R Core Team, 2022) have been employed to estimate the models involved in the first two steps.

As far as the estimation of α_k and τ_k is concerned, equations (4.13) and (4.14) of the ECM algorithm have been modified so as to guarantee that the estimated proportions of typical observations both in the $\mathbf{x}$-direction and in the $\mathbf{y}$-direction within each cluster is at least 0.5. The two modified equations are: $\alpha_k^{(h+1)} = \max\left\{0.5, \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{v}_{ik}^{(h)}}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}}\right\}$ and $\tau_k^{(h+1)} = \max\left\{0.5, \frac{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)} \hat{u}_{ik}^{(h)}}{\sum_{i=1}^{I} \hat{z}_{ik}^{(h)}}\right\}$, for $k = 1, \ldots, K$.

The iterative process is stopped using either a convergence criterion which exploits the Aitken

acceleration (Aitken, 1926) or a stopping rule in which the ECM algorithm is stopped after a given maximum number of iterations. The convergence criterion is based on the computation of the quantity $|\ell_A^{(h+1)} - \ell(\boldsymbol{\psi}^{(h)})|$, where $\ell_A^{(h+1)}$ is $(h+1)$th Aitken accelerated estimate of the log-likelihood limit and $\ell(\boldsymbol{\psi}^{(h)})$ is the incomplete log-likelihood evaluated at $\boldsymbol{\psi}^{(h)}$ (see, e.g., McNicholas, 2010). Iterations are stopped when this quantity is lower than a positive and finite tolerance threshold ϵ. The analyses reported in Sections 4.3 and 4.4 have been carried out with $\epsilon = 10^{-4}$ and 500 as the maximum number of iterations. Finally, some constraints on the eigenvalues of $\boldsymbol{\Sigma}_k^{(h)}$ and $\boldsymbol{\Xi}_k^{(h)}$ $(k = 1, \ldots, K)$ have been embedded in the ECM algorithm so as to avoid the issue of a unbounded likelihood caused by a degenerate model. Namely, following Dang et al. (2017), all eigenvalues have been required to be greater than the conservative bound 10^{-20}; furthermore, the ratio between the smallest and the largest eigenvalues of such matrices is required to be not lower than 10^{-10}.

4.2.6 Determining the value of K

Since the ECM algorithm allows to obtain an estimate of $\boldsymbol{\psi}$ for a given value of K, in any practical application in which this number is not known, it has to be determined from the data $\mathcal{S}$. This task is typically carried out by resorting to model selection criteria, such as the Bayesian information criterion (Schwarz, 1978) or the integrated completed likelihood (Biernacki et al., 2000). They can be computed as follows:

$$\begin{aligned} BIC &= 2\ell(\hat{\boldsymbol{\psi}}) - n_{\boldsymbol{\psi}} \ln I, \\ ICL_1 &= 2\ell(\hat{\boldsymbol{\psi}}) - n_{\boldsymbol{\psi}} \ln I + 2 \sum_{i=1}^{I} \sum_{k=1}^{K} \text{MAP}(\hat{z}_{ik}) \ln \hat{z}_{ik}, \\ ICL_2 &= 2\ell(\hat{\boldsymbol{\psi}}) - n_{\boldsymbol{\psi}} \ln I + 2 \sum_{i=1}^{I} \sum_{k=1}^{K} \hat{z}_{ik} \ln \hat{z}_{ik}. \end{aligned}$$

Higher values of these criteria indicate better-fit models. The BIC evaluates the adequacy of a model by taking account of the trade-off between the fit and the model complexity. In the computation of the ICL, an additional penalty accounting for the uncertainty of the estimated partition is considered (see, e.g., Andrews and McNicholas, 2011; Baek and McLachlan, 2011). In the equations for ICL_1 and ICL_2, such a penalty is based on either a soft (i.e., $\hat{z}_{ik}$) or hard (i.e., $\text{MAP}(\hat{z}_{ik})$) clustering of the sample observations. As a consequence, ICL_1 and ICL_2 penalize complex models more severely than BIC; furthermore, they should less likely split one cluster into two different components. This latter feature is consistent with the fact that the

ICL has been proposed as a criterion able to select the model which shows the greatest evidence of clustering (Biernacki et al., 2000). In contrast, selecting the number of components which leads to a good approximation to the density is the aspect which the BIC mainly focuses on (Baudry et al., 2010). In Section 4.4, BIC, ICL_1 and ICL_2 have been employed also to identify the vectors of predictors $\mathbf{X}_1, \ldots, \mathbf{X}_M$ required for the definition of the design matrix $\tilde{\mathbf{X}}^*$ in the specification of model (4.5).

4.2.7 Parsimonious models

In practical applications in which the analysis involves either many responses or many predictors, using model (4.5) to perform the analysis can become unfeasible. This is a consequence of the fact that the number of free parameters $n_{\boldsymbol{\psi}}$ of a SuCNCW model increases quadratically both with M and with P. A way to manage this issue is to resort to the approach illustrated in Celeux and Govaert (1995). With this approach, a reparameterisation of model (4.5) is obtained, in which the covariance matrices $\boldsymbol{\Sigma}_k$ and $\boldsymbol{\Xi}_k$, for $k = 1, \ldots, K$, are expressed in terms of their eigenvalues and eigenvectors; furthermore, the introduction of suitable constraints on such quantities allows to obtain parsimonious SuCNCW models. More specifically, let $\mathbf{A}_k$ be the diagonal matrix containing the eigenvalues of $\boldsymbol{\Sigma}_k$, normalised in such a way that $|\mathbf{A}_k| = 1$; let $\mathbf{D}_k$ be the matrix with the corresponding eigenvectors, and $\xi_k = |\boldsymbol{\Sigma}_k|^{1/D}$. By exploiting the eigen-decomposition $\boldsymbol{\Sigma}_k = \xi_k \mathbf{D}_k \mathbf{A}_k \mathbf{D}_k'$, variances and covariances in $\boldsymbol{\Sigma}_k$ can be obtained from ξ_k, $\mathbf{A}_k$ and $\mathbf{D}_k$, which control the volume, shape and orientation of the kth cluster of observations with respect to the predictors. Constraining ξ_k, $\mathbf{A}_k$ and $\mathbf{D}_k$ on this decomposition in model (4.5) with $K > 1$ will lead to 14 different covariance structures for the predictors. Additional information about these parameterisations are reported in Table 4.1. When ξ_k, $\mathbf{A}_k$ and $\mathbf{D}_k$ are all variable across the K clusters (`VVV` acronym in Table 4.1), the resulting covariance structures of the predictors will be fully unconstrained. From the simultaneous application of the same decomposition to the covariance matrices $\boldsymbol{\Xi}_k$, for $k = 1, \ldots, K$, 196 differentially parameterised SuCNCW models of order K can be obtained, for any given $K > 1$. The updates of $\boldsymbol{\Sigma}_k^{(h)}$ and $\boldsymbol{\Xi}_k^{(h)}$ reported in equations (4.16) and (4.19) apply to the SuCNCW models with the `VVV` parameterisation for either $\boldsymbol{\Sigma}_k$ or $\boldsymbol{\Xi}_k$ (i.e.: fully unconstrained covariance structures of the predictors or responses). For the ML estimation of $\boldsymbol{\Sigma}_k$ or $\boldsymbol{\Xi}_k$ under any other SuCNCW model, the M step updates in the ECM algorithm depend on the specific parameterisation to be employed (see Celeux and Govaert, 1995, for more details). For the estimation of models obtained using the `EVE` and `VVE`

Table 4.1: Parameterisations of the component-covariance matrices.

Acronym	Model	Distribution	Volume	Shape	Orientation
EEE	$\xi\mathbf{DAD}'$	Ellipsoidal	Equal	Equal	Equal
VVV	$\xi_k\mathbf{D}_k\mathbf{A}_k\mathbf{D}_k'$	Ellipsoidal	Variable	Variable	Variable
EII	$\xi\mathbf{I}$	Spherical	Equal	Equal	–
VII	$\xi_k\mathbf{I}$	Spherical	Variable	Equal	–
EEI	$\xi\mathbf{A}$	Diagonal	Equal	Equal	–
VEI	$\xi_k\mathbf{A}$	Diagonal	Variable	Equal	–
EVI	$\xi\mathbf{A}_k$	Diagonal	Equal	Variable	–
VVI	$\xi_k\mathbf{A}_k$	Diagonal	Variable	Variable	–
EEV	$\xi\mathbf{D}_k\mathbf{A}\mathbf{D}_k'$	Ellipsoidal	Equal	Equal	Variable
VEV	$\xi_k\mathbf{D}_k\mathbf{A}\mathbf{D}_k'$	Ellipsoidal	Variable	Equal	Variable
EVE	$\xi\mathbf{D}\mathbf{A}_k\mathbf{D}'$	Ellipsoidal	Equal	Variable	Equal
VVE	$\xi_k\mathbf{D}\mathbf{A}_k\mathbf{D}'$	Ellipsoidal	Variable	Variable	Equal
VEE	$\xi_k\mathbf{D}\mathbf{A}\mathbf{D}'$	Ellipsoidal	Variable	Equal	Equal
EVV	$\xi\mathbf{D}_k\mathbf{A}_k\mathbf{D}_k'$	Ellipsoidal	Equal	Variable	Variable

parameterisations, it is possible to resort to some algorithms which are computationally feasible also in high-dimensional situations (Browne and McNicholas, 2014a,b). As far as SuCNCW models of order $K = 1$ are concerned, the possible covariance structures for both responses and covariates are: diagonal with different entries (VI), diagonal with the same entries (EI) and fully unconstrained (VV). Thus, when $K = 1$, only nine differentially parameterised models can be specified.

4.3 Simulation studies

4.3.1 Settings

The task of investigating the effectiveness of SuCNCW models in comparison with NCW, CNCW and SuNCW models has been carried out in a multivariate setting with $M = 2$ responses, $P = 3$ covariates and simulated datasets comprising observations randomly sampled from $K = 3$ different distributions.

All the models employed to generate the datasets have been specified within the seemingly unrelated approach. More specifically, the response Y_1 has been assumed to linearly depend on X_1 and X_2, while the assumption for Y_2 is that it linearly depends on X_1 and X_3. Thus, $\mathbf{X}_1 = (X_1, X_2)'$, $\mathbf{X}_2 = (X_1, X_3)'$, and equation (4.4) reduces to:

$$\begin{aligned}\mathbb{E}(Y_1|\mathbf{X} = \mathbf{x}, G_k) &= \mathbf{x}_1^{*\prime}\boldsymbol{\beta}_{k1}^* = \beta_{k10} + \beta_{k11}x_1 + \beta_{k12}x_2,\\ \mathbb{E}(Y_2|\mathbf{X} = \mathbf{x}, G_k) &= \mathbf{x}_2^{*\prime}\boldsymbol{\beta}_{k2}^* = \beta_{k20} + \beta_{k21}x_1 + \beta_{k22}x_3.\end{aligned}$$

As far as the data generation processes are concerned, models belonging to the following classes have been employed:

(a) SuNCW;

(b) SuCNCW with $\alpha_k = 0.95,\ \eta_k = 5, \tau_k = 0.9,\ \lambda_k = 10\ \forall k$;

(c) Student-t CW models with $\nu_1 = \nu_2 = \nu_3 = 4$ degrees of freedom.

All of these processes share the following common parameters for the data generation: $\pi_1 = 0.4$, $\pi_2 = 0.35$, $\pi_3 = 0.25$, $\boldsymbol{\mu}_1 = (0, 0, 0)'$, $\boldsymbol{\mu}_2 = (2, 4, -2)'$, $\boldsymbol{\mu}_3 = \boldsymbol{\mu}_2 + 2\epsilon \cdot \mathbf{1}_P$, where $\mathbf{1}_P$ is the $P \times 1$ vector having each element equal to 1, $\boldsymbol{\beta}_1^* = (-2, 0.75, 1, 1, 0.5, -2)'$, $\boldsymbol{\beta}_2^* = (0.5, 1.75, 0.25, 1, 1, 1)'$, $\boldsymbol{\beta}_3^* = \boldsymbol{\beta}_2^* + \epsilon \cdot \mathbf{1}_6$, $\boldsymbol{\Sigma}_1 = \begin{pmatrix} 1.72 & -0.18 & 0.27 \\ -0.18 & 1.89 & 0.27 \\ 0.27 & 0.27 & 2.89 \end{pmatrix}$, $\boldsymbol{\Sigma}_2 = \begin{pmatrix} 2.33 & -0.52 & -0.06 \\ -0.52 & 0.88 & -0.34 \\ -0.06 & -0.34 & 1.04 \end{pmatrix}$, $\boldsymbol{\Sigma}_3 = \boldsymbol{\Sigma}_2$, $\boldsymbol{\Xi}_1 = \begin{pmatrix} 1.34 & 0.47 \\ 0.47 & 1.66 \end{pmatrix}$, $\boldsymbol{\Xi}_2 = \begin{pmatrix} 0.50 & 0.04 \\ 0.04 & 1.50 \end{pmatrix}$, $\boldsymbol{\Xi}_3 = \boldsymbol{\Xi}_2$. Thus, the covariance structures of both the predictors and the responses within the three groups have been obtained using the `VVV` parameterisation. Since the difference between the parameters $(\boldsymbol{\theta}_{k\mathbf{x}}, \boldsymbol{\theta}_{k\mathbf{y}})$ for $k = 2, 3$ only depends on ϵ, different values of ϵ can be chosen so as to determine different degrees of separation between the second and third groups of sample observations.

Under each process mentioned above, 100 different datasets have been generated considering the sample size ($I = 500, 1000$) and the degree of separation ($\epsilon = 0.35, 0.55$) as experimental factors. Thus, 1200 different datasets have been generated. The whole analysis has been run by employing an IBM x3750 M4 server with 4 Intel Xeon E5-4620 processors with 8 cores and 128GB RAM.

4.3.2 Results

The comparative study of the effectiveness of the four model classes has been structured into two parts. In the first part, SuNCW, NCW, CNCW and SuCNCW models of order $K = 3$ with the `VVV` parameterisation for both $\boldsymbol{\Sigma}_k$ and $\boldsymbol{\Xi}_k$ $(k = 1, 2, 3)$ have been fitted to each dataset. As far as NCW and CNCW models are concerned, each response has been assumed to linearly

depend on all the covariates; namely:

$$\begin{aligned}\mathbb{E}(Y_1|\mathbf{X}=\mathbf{x},G_k) &= \beta_{k10}+\beta_{k11}x_1+\beta_{k12}x_2+\beta_{k13}x_3,\\ \mathbb{E}(Y_2|\mathbf{X}=\mathbf{x},G_k) &= \beta_{k20}+\beta_{k21}x_1+\beta_{k22}x_3+\beta_{k23}x_2.\end{aligned}$$

With this specification, the fitted NCW and CNCW models are not parsimonious: for each k, six regression coefficients have been estimated although, in fact, only four of them are different from zero. As far as the time elapsed between the start and completion of the parameter estimation is concerned, fitting a SuCNCW model has required - on average over the 100 datasets with $I = 500$ - between 1.069 and 95.243 seconds, depending on the data generation process and the value of ϵ. The minimum and maximum average execution times have resulted to be equal to 0.991 and 20.113 seconds with SuNCW models, 0.995 and 18.782 seconds with NCW models, 1.034 and 116.922 seconds with CNCW models. However, it is important to note that the ECM algorithm has not been implemented with the goal of being efficient from a computational point of view. Thus, more efficient implementations could greatly reduce these illustrative CPU times. In the first part of this study, the comparison among the competing models has been carried out by examining their performances with reference to the following three aspects: *(i)* the estimation of the proportions of typical observations and the degrees of contamination both in the $\mathbf{x}$-direction (proper estimation of α_k and η_k) and in the $\mathbf{y}$-direction (proper estimation of τ_k and λ_k); *(ii)* the ability to recover the true values of the unknown parameters (parameter recovery); *(iii)* the capability to recover the true partition of the sample observations (classification recovery). The aspect *(i)* has been studied only for the fitted CNCW and SuCNCW models. The evaluation of the aspect *(ii)* has been focused on the regression coefficients. In order to prevent the effects of label switching issues on the evaluation of these aspects, the components of the mixtures involved in each fitted model have been labelled by minimising the Euclidean distance to the true parameter values (see, e.g., Bai et al., 2012; Yao, 2014; Punzo and McNicholas, 2017; Perrone and Soffritti, 2023).

In the second part, the study aims at evaluating the performances of the four model classes without exploiting the knowledge of the true value of K. Thus, also SuNCW, NCW, CNCW and SuCNCW models of order $K = 1, 2, 4$ with the `VVV` parameterisation for $\boldsymbol{\Sigma}_k$ and $\boldsymbol{\Xi}_k$ have been fitted to each dataset. The results obtained for all the examined values of K have been employed to study the following aspects: *(iv)* the capability to reach the best trade-off between

the fit and model complexity; *(v)* the ability of BIC, ICL_1 and ICL_2 to detect the true value of K (comparison among information criteria); *(vi)* a further evaluation of the classification recovery.

Estimation of α_k, τ_k, η_k, λ_k

When the datasets only contain typical observations in either directions (first process), the averages of the estimated proportions of good points ($\hat{\alpha}_k$ and $\hat{\tau}_k$) and the estimated inflation parameters ($\hat{\eta}_k$ and $\hat{\lambda}_k$) are close to 1. In the presence of datasets with contaminated observations generated according to the second process, the estimates of such parameters are, on average, close to their true values. These results hold true under both CNCW and SuCNCW models, regardless of the level of separation and the sample size (see the upper and central parts of Tables 4.2–4.5). Thus, the proportions of good points and the inflation parameters appear to be properly estimated using either types of models. However, in the second process, slightly higher standard deviations have been registered for the estimated inflation parameters, especially for λ_k. These latter results seem to highlight that the estimation of the inflation parameters is characterised by a certain instability under both CNCW and SuCNCW models. This phenomenon seems to reduce as the sample size increases. Finally, with the contaminated datasets generated according to the third process, the mean values of $\hat{\alpha}_k$, $\hat{\tau}_k$, $\hat{\eta}_k$ and $\hat{\lambda}_k$ for $k = 1, 2, 3$ are all quite far from 1, regardless of the values of ϵ and I (see the lower part of Tables 4.2–4.5). Thus, CNCW and SuCNCW models have been able to detect the departure from a normal distribution for both $\mathbf{X}|G_k$ and $\mathbf{Y}|(\mathbf{X} = \mathbf{x}, G_k)$, for $k = 1, 2, 3$, due to the use of the Student-t distribution in the third data generation process.

Parameter recovery

To evaluate the aspect *(ii)* with respect to the regression coefficients β_{kmp}, the following quantities have been computed:

$$\text{Bias}\left(\hat{\beta}_{kmp}\right) = \left|\frac{\sum_{r=1}^{100}\hat{\beta}_{kmp}^{(r)}}{100} - \beta_{kmp}\right|, \; k = 1, 2, 3, \; m = 1, 2, \; p = 1, 2,$$

$$\text{RMSE}\left(\hat{\beta}_{kmp}\right) = \sqrt{\frac{\sum_{r=1}^{100}\left(\beta_{kmp} - \hat{\beta}_{kmp}^{(r)}\right)^2}{100}}, \; k = 1, 2, 3, \; m = 1, 2, \; p = 1, 2,$$

Table 4.2: Estimation of α_k and η_k: averages and standard deviations of the estimates over 100 samples for the fitted CNCW and SuCNCW models of order $K = 3$ ($I = 500$).

	CNCW						SuCNCW					
	$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\alpha}_3$	$\hat{\eta}_1$	$\hat{\eta}_2$	$\hat{\eta}_3$	$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\alpha}_3$	$\hat{\eta}_1$	$\hat{\eta}_2$	$\hat{\eta}_3$
I process, $\epsilon = 0.55$												
average	0.999	0.999	0.999	1.001	1.001	1.001	0.999	0.999	0.999	1.001	1.001	1.001
s.d.	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
I process, $\epsilon = 0.35$												
average	0.995	0.990	0.992	1.037	1.079	1.052	0.990	0.991	0.993	1.051	1.060	1.032
s.d.	0.037	0.042	0.051	0.235	0.313	0.217	0.061	0.046	0.050	0.285	0.265	0.176
II process, $\epsilon = 0.55$												
average	0.914	0.918	0.935	4.796	4.600	5.258	0.910	0.916	0.934	4.894	4.752	5.313
s.d.	0.114	0.083	0.060	2.950	2.560	2.835	0.119	0.083	0.059	2.913	2.651	2.775
II process, $\epsilon = 0.35$												
average	0.905	0.925	0.927	4.030	4.036	4.946	0.918	0.921	0.929	3.938	4.209	5.121
s.d.	0.123	0.068	0.079	2.977	2.034	2.710	0.111	0.075	0.076	2.697	2.197	2.781
III process, $\epsilon = 0.55$												
average	0.767	0.791	0.801	5.651	5.801	4.857	0.761	0.782	0.788	5.386	6.250	5.010
s.d.	0.173	0.151	0.123	6.862	6.066	2.083	0.168	0.150	0.121	3.790	6.145	1.929
III process, $\epsilon = 0.35$												
average	0.804	0.834	0.816	13.887	6.280	4.745	0.793	0.826	0.784	6.690	7.785	4.958
s.d.	0.170	0.126	0.134	69.355	5.339	2.883	0.164	0.123	0.126	9.576	9.342	2.640

Table 4.3: Estimation of τ_k and λ_k: averages and standard deviations of the estimates over 100 samples for the fitted CNCW and SuCNCW models of order $K = 3$ ($I = 500$).

	CNCW						SuCNCW					
	$\hat{\tau}_1$	$\hat{\tau}_2$	$\hat{\tau}_3$	$\hat{\lambda}_1$	$\hat{\lambda}_2$	$\hat{\lambda}_3$	$\hat{\tau}_1$	$\hat{\tau}_2$	$\hat{\tau}_3$	$\hat{\lambda}_1$	$\hat{\lambda}_2$	$\hat{\lambda}_3$
I process, $\epsilon = 0.55$												
average	0.999	0.999	0.999	1.001	1.001	1.001	0.999	0.999	0.999	1.001	1.001	1.001
s.d.	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
I process, $\epsilon = 0.35$												
average	0.991	0.978	0.993	1.138	1.193	1.078	0.994	0.981	0.993	1.058	1.138	1.077
s.d.	0.057	0.079	0.050	0.743	0.697	0.362	0.050	0.072	0.050	0.412	0.509	0.347
II process, $\epsilon = 0.55$												
average	0.881	0.894	0.901	9.589	9.361	10.107	0.877	0.895	0.897	10.045	9.295	10.171
s.d.	0.077	0.061	0.048	4.844	4.401	3.350	0.069	0.058	0.045	4.428	3.957	3.222
II process, $\epsilon = 0.35$												
average	0.886	0.892	0.901	12.431	8.671	9.155	0.889	0.893	0.896	10.334	8.981	9.887
s.d.	0.096	0.077	0.055	22.491	4.854	3.940	0.081	0.070	0.049	6.784	4.698	3.583
III process, $\epsilon = 0.55$												
average	0.794	0.787	0.802	9.019	7.516	10.560	0.786	0.774	0.796	9.125	8.822	10.598
s.d.	0.179	0.161	0.153	14.686	6.687	25.360	0.173	0.159	0.150	12.024	9.031	24.847
III process, $\epsilon = 0.35$												
average	0.866	0.799	0.834	12.529	9.095	6.161	0.833	0.780	0.800	9.607	9.505	6.278
s.d.	0.154	0.155	0.137	47.673	19.489	9.883	0.154	0.156	0.135	17.960	20.006	7.455

Table 4.4: Estimation of α_k and η_k: averages and standard deviations of the estimates over 100 samples for the fitted CNCW and SuCNCW models of order $K = 3$ ($I = 1000$).

	CNCW						SuCNCW					
	$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\alpha}_3$	$\hat{\eta}_1$	$\hat{\eta}_2$	$\hat{\eta}_3$	$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\alpha}_3$	$\hat{\eta}_1$	$\hat{\eta}_2$	$\hat{\eta}_3$
I process, $\epsilon = 0.55$												
average	0.999	0.999	0.999	1.001	1.001	1.001	0.999	0.999	0.999	1.001	1.001	1.001
s.d.	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
I process, $\epsilon = 0.35$												
average	0.999	0.999	0.996	1.002	1.002	1.016	0.999	0.999	0.999	1.002	1.001	1.002
s.d.	0.000	0.000	0.002	0.006	0.004	0.099	0.000	0.000	0.000	0.011	0.001	0.001
II process, $\epsilon = 0.55$												
average	0.919	0.943	0.938	4.642	5.020	4.777	0.917	0.942	0.937	4.660	5.100	4.811
s.d.	0.076	0.030	0.036	2.044	1.768	1.477	0.079	0.030	0.035	2.006	1.764	1.434
II process, $\epsilon = 0.35$												
average	0.918	0.936	0.950	4.782	5.007	4.998	0.928	0.933	0.947	4.783	5.005	5.362
s.d.	0.104	0.062	0.034	2.773	2.042	1.969	0.083	0.063	0.034	2.163	1.838	1.851
III process, $\epsilon = 0.55$												
average	0.801	0.819	0.829	6.689	5.684	5.657	0.793	0.832	0.829	5.689	6.229	5.676
s.d.	0.134	0.099	0.084	5.687	2.650	2.999	0.133	0.097	0.083	2.851	4.547	2.975
III process, $\epsilon = 0.35$												
average	0.794	0.793	0.828	7.890	5.918	6.546	0.771	0.801	0.821	6.197	6.636	6.745
s.d.	0.144	0.127	0.098	13.509	5.243	9.151	0.138	0.118	0.104	8.745	5.342	9.151

Table 4.5: Estimation of τ_k and λ_k: averages and standard deviations of the estimates over 100 samples for the fitted CNCW and SuCNCW models of order $K = 3$ ($I = 1000$).

	CNCW						SuCNCW					
	$\hat{\tau}_1$	$\hat{\tau}_2$	$\hat{\tau}_3$	$\hat{\lambda}_1$	$\hat{\lambda}_2$	$\hat{\lambda}_3$	$\hat{\tau}_1$	$\hat{\tau}_2$	$\hat{\tau}_3$	$\hat{\lambda}_1$	$\hat{\lambda}_2$	$\hat{\lambda}_3$
I process, $\epsilon = 0.55$												
average	0.999	0.999	0.999	1.001	1.001	1.001	0.999	0.999	0.999	1.001	1.001	1.001
s.d.	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
I process, $\epsilon = 0.35$												
average	0.999	0.997	0.998	1.015	1.023	1.021	0.999	0.994	0.999	1.002	1.035	1.001
s.d.	0.001	0.021	0.006	0.100	0.212	0.201	0.000	0.000	0.000	0.009	0.240	0.002
II process, $\epsilon = 0.55$												
average	0.888	0.903	0.900	10.379	10.362	10.249	0.888	0.901	0.898	10.416	10.447	10.290
s.d.	0.052	0.035	0.029	3.943	2.849	2.371	0.044	0.033	0.027	3.813	2.688	2.171
II process, $\epsilon = 0.35$												
average	0.898	0.909	0.911	9.163	8.950	8.706	0.897	0.903	0.905	9.442	9.152	9.722
s.d.	0.078	0.048	0.044	5.473	4.411	3.872	0.059	0.047	0.035	4.857	4.289	3.096
III process, $\epsilon = 0.55$												
average	0.792	0.768	0.819	8.368	6.934	7.159	0.792	0.773	0.824	8.054	7.149	7.251
s.d.	0.144	0.140	0.109	7.353	5.327	4.632	0.144	0.140	0.103	6.703	6.907	4.871
III process, $\epsilon = 0.35$												
average	0.822	0.783	0.810	7.178	7.504	6.651	0.801	0.781	0.805	7.670	7.371	6.647
s.d.	0.147	0.160	0.136	8.954	7.514	4.071	0.151	0.148	0.137	8.435	5.392	3.768

where $\hat{\beta}_{kmp}^{(r)}$ is the ML estimate of β_{kmp} obtained from the rth dataset ($r = 1, \ldots, 100$). Since NCW and CNCW models also contain some regression coefficients associated with irrelevant regressors, the bias and RMSE have been computed also for these additional coefficients, using 0 as their true value.

The ability to recover the regression coefficients using SuNCW and SuCNCW models under the first process results to be the same with both sample sizes and both degrees of separation (see their biases and RMSEs in Tables 4.6 and 4.7). As all the parameters able to capture the possible presence of mildly atypical observations in SuCNCW have been properly estimated (see the previous aspect), the inclusion of these parameters when the analysed datasets do not contain atypical observations does not show any relevant impact on the recovery of the true β_{kmp}. On the contrary, if irrelevant predictors are included in both regression equations (i.e., using NCW and CNCW models), a slight increase in the RMSEs of some regression coefficients is observed when $I = 500$, and this is especially true for $\epsilon = 0.35$. However, such an effect almost disappears with the sample size 1000. With the contaminated datasets generated using the second process, as expected, SuCNCW models show the best performance with both sample sizes and both degrees of separation (see Tables 4.8 and 4.9). With this process, the accuracy of CNCW models seem to be slightly higher than that of NCW and SuNCW models for the majority of the regression coefficients. Under the third process, the lowest RMSEs are still obtained using the SuCNCW model with all the examined experimental situations (see Tables 4.10 and 4.11). Furthermore, thanks to their effectiveness in detecting the non-normality of the distributions of $\mathbf{X}|G_k$ and $\mathbf{Y}|(\mathbf{X} = \mathbf{x}, G_k)$ for $k = 1, 2, 3$, CNCW models generally perform slightly better than NCW and SuNCW models. However, it is worth noting that, with the lowest values of I and ϵ, the RMSEs obtained using the SuNCW model are slightly lower than those registered with CNCW models for the majority of the regression coefficients. As far as the irrelevant regressors are concerned, NCW and CNCW models appear to be equally capable of recognising their presence in the analysis of uncontaminated datasets (I process), as the corresponding estimated regression coefficients are on average quite close to 0. However, when the data are contaminated (II and III processes), large values of the RMSE have been registered for the estimates of the regression coefficients associated with the irrelevant regressors in the second and third cluster. This latter result is particularly evident when the separation between these two clusters is low. Furthermore, the precision of CNCW models in the estimation of the effect of most irrelevant regressors using contaminated datasets results to be higher than that of NCW models.

Table 4.6: Bias and RMSE for the regression coefficients β_{kmp} under the four types of models in the first process ($I = 500$).

	Bias				RMSE			
	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW
$\epsilon = 0.55$								
β_{111}	0.004	0.007	0.007	0.004	0.047	0.049	0.049	0.047
β_{112}	0.004	0.010	0.010	0.004	0.075	0.079	0.079	0.075
β_{121}	0.004	0.003	0.003	0.004	0.068	0.074	0.074	0.068
β_{122}	0.002	0.001	0.001	0.002	0.103	0.110	0.110	0.103
β_{211}	0.000	0.001	0.001	0.000	0.037	0.038	0.038	0.037
β_{212}	0.003	0.002	0.002	0.003	0.060	0.061	0.061	0.060
β_{221}	0.002	0.001	0.001	0.002	0.058	0.064	0.064	0.058
β_{222}	0.006	0.008	0.008	0.006	0.089	0.098	0.098	0.089
β_{311}	0.002	0.003	0.003	0.002	0.064	0.064	0.064	0.064
β_{312}	0.002	0.004	0.004	0.002	0.058	0.060	0.060	0.058
β_{321}	0.006	0.005	0.005	0.006	0.073	0.074	0.074	0.073
β_{322}	0.006	0.004	0.004	0.006	0.052	0.054	0.054	0.052
Irrelevant regressors								
β_{113}	-	0.003	0.003	-	-	0.046	0.046	-
β_{123}	-	0.006	0.006	-	-	0.060	0.060	-
β_{213}	-	0.012	0.012	-	-	0.069	0.069	-
β_{223}	-	0.008	0.008	-	-	0.133	0.133	-
β_{313}	-	0.004	0.004	-	-	0.056	0.056	-
β_{323}	-	0.006	0.006	-	-	0.114	0.114	-
$\epsilon = 0.35$								
β_{111}	0.009	0.009	0.009	0.009	0.036	0.039	0.039	0.036
β_{112}	0.008	0.013	0.014	0.010	0.056	0.085	0.086	0.058
β_{121}	0.002	0.001	0.001	0.001	0.049	0.055	0.055	0.049
β_{122}	0.034	0.038	0.037	0.034	0.083	0.097	0.096	0.082
β_{211}	0.008	0.010	0.010	0.008	0.032	0.046	0.046	0.032
β_{212}	0.008	0.009	0.010	0.008	0.053	0.081	0.081	0.053
β_{221}	0.015	0.017	0.017	0.016	0.049	0.055	0.055	0.049
β_{222}	0.014	0.017	0.017	0.014	0.070	0.079	0.079	0.069
β_{311}	0.004	0.004	0.004	0.003	0.042	0.042	0.042	0.042
β_{312}	0.001	0.000	0.000	0.001	0.039	0.042	0.042	0.039
β_{321}	0.000	0.001	0.001	0.001	0.047	0.049	0.049	0.048
β_{322}	0.005	0.007	0.007	0.005	0.041	0.043	0.043	0.041
Irrelevant regressors								
β_{113}	-	0.003	0.003	-	-	0.030	0.030	-
β_{123}	-	0.006	0.006	-	-	0.049	0.048	-
β_{213}	-	0.003	0.003	-	-	0.074	0.074	-
β_{223}	-	0.002	0.002	-	-	0.104	0.104	-
β_{313}	-	0.004	0.004	-	-	0.060	0.060	-
β_{423}	-	0.006	0.005	-	-	0.094	0.094	-

Table 4.7: Bias and RMSE for the regression coefficients β_{kmp} under the four types of models in the first process ($I = 1000$).

	Bias				RMSE			
	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW
$\epsilon = 0.55$								
β_{111}	0.003	0.004	0.004	0.003	0.033	0.034	0.034	0.033
β_{112}	0.002	0.003	0.003	0.002	0.053	0.059	0.059	0.053
β_{121}	0.001	0.002	0.002	0.001	0.051	0.056	0.056	0.051
β_{122}	0.012	0.008	0.008	0.012	0.081	0.086	0.086	0.081
β_{211}	0.003	0.004	0.004	0.003	0.030	0.030	0.030	0.030
β_{212}	0.002	0.003	0.003	0.002	0.047	0.047	0.047	0.047
β_{221}	0.002	0.003	0.003	0.002	0.046	0.052	0.052	0.046
β_{222}	0.009	0.010	0.010	0.009	0.067	0.070	0.070	0.067
β_{311}	0.004	0.004	0.004	0.004	0.047	0.048	0.048	0.047
β_{312}	0.002	0.002	0.002	0.002	0.040	0.044	0.044	0.040
β_{321}	0.004	0.004	0.004	0.004	0.050	0.051	0.051	0.050
β_{322}	0.003	0.003	0.003	0.003	0.040	0.041	0.041	0.040
Irrelevant regressors								
β_{113}	-	0.004	0.004	-	-	0.038	0.038	-
β_{123}	-	0.006	0.006	-	-	0.049	0.049	-
β_{213}	-	0.010	0.010	-	-	0.054	0.054	-
β_{223}	-	0.009	0.009	-	-	0.094	0.094	-
β_{313}	-	0.002	0.002	-	-	0.043	0.043	-
β_{323}	-	0.001	0.001	-	-	0.088	0.088	-
$\epsilon = 0.35$								
β_{111}	0.004	0.004	0.004	0.004	0.025	0.026	0.026	0.025
β_{112}	0.001	0.001	0.001	0.001	0.043	0.048	0.048	0.043
β_{121}	0.004	0.008	0.008	0.004	0.040	0.042	0.042	0.040
β_{122}	0.001	0.004	0.004	0.002	0.058	0.059	0.059	0.058
β_{211}	0.002	0.002	0.002	0.002	0.019	0.020	0.020	0.019
β_{212}	0.011	0.013	0.013	0.011	0.032	0.035	0.035	0.032
β_{221}	0.001	0.000	0.000	0.001	0.031	0.034	0.034	0.031
β_{222}	0.008	0.011	0.011	0.008	0.058	0.057	0.057	0.058
β_{311}	0.005	0.006	0.006	0.005	0.030	0.031	0.031	0.030
β_{312}	0.000	0.001	0.001	0.000	0.030	0.033	0.033	0.030
β_{321}	0.014	0.015	0.015	0.015	0.038	0.039	0.039	0.038
β_{322}	0.006	0.007	0.007	0.006	0.028	0.030	0.030	0.028
Irrelevant regressors								
β_{113}	-	0.000	0.000	-	-	0.025	0.025	-
β_{123}	-	0.005	0.005	-	-	0.037	0.037	-
β_{213}	-	0.001	0.001	-	-	0.032	0.032	-
β_{223}	-	0.017	0.016	-	-	0.068	0.068	-
β_{313}	-	0.003	0.003	-	-	0.033	0.033	-
β_{323}	-	0.005	0.005	-	-	0.063	0.063	-

Table 4.8: Bias and RMSE for the regression coefficients β_{kmp} under the four types of models in the second process ($I = 500$).

	Bias				RMSE			
	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW
$\epsilon = 0.55$								
β_{111}	0.010	0.014	0.007	0.000	0.096	0.100	0.070	0.063
β_{112}	0.041	0.072	0.038	0.006	0.199	0.269	0.203	0.107
β_{121}	0.005	0.006	0.007	0.001	0.096	0.114	0.087	0.076
β_{122}	0.025	0.006	0.008	0.005	0.153	0.160	0.126	0.111
β_{211}	0.008	0.012	0.007	0.002	0.105	0.145	0.178	0.039
β_{212}	0.074	0.095	0.083	0.003	0.291	0.333	0.358	0.062
β_{221}	0.011	0.006	0.017	0.002	0.112	0.179	0.157	0.062
β_{222}	0.045	0.049	0.012	0.002	0.375	0.316	0.288	0.095
β_{311}	0.012	0.014	0.003	0.003	0.088	0.089	0.066	0.065
β_{312}	0.001	0.004	0.002	0.001	0.075	0.085	0.068	0.062
β_{321}	0.006	0.007	0.001	0.001	0.093	0.098	0.075	0.074
β_{322}	0.009	0.006	0.004	0.002	0.064	0.071	0.059	0.053
Irrelevant regressors								
β_{113}	-	0.005	0.003	-	-	0.060	0.051	-
β_{123}	-	0.003	0.006	-	-	0.105	0.073	-
β_{213}	-	0.063	0.034	-	-	0.246	0.191	-
β_{223}	-	0.029	0.005	-	-	0.228	0.168	-
β_{313}	-	0.030	0.003	-	-	0.190	0.322	-
β_{323}	-	0.020	0.002	-	-	0.315	0.328	-
$\epsilon = 0.35$								
β_{111}	0.071	0.094	0.043	0.022	0.141	0.172	0.129	0.075
β_{112}	0.230	0.321	0.115	0.063	0.464	0.516	0.334	0.324
β_{121}	0.050	0.023	0.024	0.016	0.252	0.263	0.212	0.210
β_{122}	0.003	0.038	0.052	0.032	0.248	0.276	0.193	0.150
β_{211}	0.026	0.035	0.017	0.006	0.213	0.234	0.182	0.177
β_{212}	0.093	0.120	0.115	0.073	0.335	0.408	0.439	0.265
β_{221}	0.008	0.039	0.003	0.005	0.213	0.270	0.157	0.138
β_{222}	0.052	0.084	0.099	0.038	0.439	0.554	0.555	0.269
β_{311}	0.011	0.011	0.001	0.001	0.085	0.086	0.059	0.059
β_{312}	0.003	0.003	0.002	0.004	0.081	0.086	0.061	0.058
β_{321}	0.005	0.004	0.002	0.004	0.098	0.098	0.079	0.075
β_{322}	0.007	0.007	0.005	0.003	0.075	0.083	0.062	0.059
Irrelevant regressors								
β_{113}	-	0.002	0.004	-	-	0.057	0.044	-
β_{123}	-	0.005	0.001	-	-	0.097	0.074	-
β_{213}	-	0.234	0.099	-	-	0.358	0.259	-
β_{223}	-	0.148	0.027	-	-	0.363	0.264	-
β_{313}	-	0.044	0.040	-	-	0.306	0.241	-
β_{323}	-	0.037	0.026	-	-	0.459	0.463	-

Table 4.9: Bias and RMSE for the regression coefficients β_{kmp} under the four types of models in the second process ($I = 1000$).

	Bias				RMSE			
	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW
$\epsilon = 0.55$								
β_{111}	0.007	0.015	0.001	0.001	0.067	0.080	0.041	0.033
β_{112}	0.024	0.053	0.012	0.007	0.154	0.212	0.107	0.058
β_{121}	0.000	0.014	0.000	0.000	0.074	0.090	0.057	0.053
β_{122}	0.006	0.018	0.006	0.009	0.104	0.120	0.081	0.080
β_{211}	0.008	0.011	0.006	0.004	0.041	0.042	0.035	0.031
β_{212}	0.028	0.032	0.011	0.001	0.168	0.176	0.110	0.044
β_{221}	0.000	0.005	0.006	0.005	0.057	0.073	0.055	0.046
β_{222}	0.020	0.022	0.006	0.010	0.239	0.265	0.189	0.065
β_{311}	0.007	0.005	0.001	0.002	0.058	0.059	0.048	0.047
β_{312}	0.008	0.006	0.002	0.003	0.060	0.064	0.047	0.044
β_{321}	0.002	0.002	0.004	0.004	0.062	0.065	0.053	0.053
β_{322}	0.001	0.002	0.006	0.006	0.053	0.056	0.043	0.042
Irrelevant regressors								
β_{113}	-	0.005	0.001	-	-	0.049	0.042	-
β_{123}	-	0.013	0.003	-	-	0.066	0.046	-
β_{213}	-	0.042	0.003	-	-	0.181	0.099	-
β_{223}	-	0.065	0.010	-	-	0.177	0.100	-
β_{313}	-	0.012	0.003	-	-	0.062	0.041	-
β_{323}	-	0.003	0.001	-	-	0.126	0.095	-
$\epsilon = 0.35$								
β_{111}	0.054	0.082	0.020	0.009	0.092	0.126	0.080	0.061
β_{112}	0.130	0.267	0.097	0.013	0.221	0.377	0.261	0.104
β_{121}	0.018	0.018	0.014	0.000	0.142	0.131	0.136	0.115
β_{122}	0.060	0.006	0.014	0.006	0.237	0.178	0.150	0.099
β_{211}	0.004	0.019	0.029	0.027	0.167	0.208	0.178	0.109
β_{212}	0.103	0.108	0.061	0.027	0.299	0.337	0.341	0.167
β_{221}	0.037	0.013	0.029	0.014	0.229	0.137	0.184	0.069
β_{222}	0.182	0.142	0.005	0.013	0.566	0.502	0.492	0.151
β_{311}	0.001	0.000	0.004	0.003	0.059	0.061	0.047	0.046
β_{312}	0.003	0.003	0.001	0.001	0.058	0.057	0.045	0.041
β_{321}	0.003	0.004	0.008	0.007	0.065	0.068	0.058	0.055
β_{322}	0.004	0.001	0.003	0.003	0.044	0.053	0.044	0.041
Irrelevant regressors								
β_{113}	-	0.004	0.000	-	-	0.046	0.035	-
β_{123}	-	0.003	0.003	-	-	0.068	0.052	-
β_{213}	-	0.246	0.121	-	-	0.401	0.341	-
β_{223}	-	0.152	0.075	-	-	0.470	0.457	-
β_{313}	-	0.041	0.059	-	-	0.180	0.228	-
β_{323}	-	0.002	0.003	-	-	0.250	0.200	-

Table 4.10: Bias and RMSE for the regression coefficients β_{kmp} under the four types of models in the third process ($I = 500$).

	Bias				RMSE			
	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW
$\epsilon = 0.55$								
β_{111}	0.026	0.024	0.005	0.004	0.103	0.090	0.050	0.045
β_{1121}	0.124	0.182	0.028	0.017	0.290	0.394	0.181	0.136
β_{121}	0.051	0.049	0.025	0.020	0.129	0.130	0.099	0.084
β_{122}	0.049	0.006	0.010	0.009	0.164	0.124	0.112	0.106
β_{211}	0.002	0.015	0.000	0.004	0.168	0.159	0.092	0.063
β_{212}	0.125	0.134	0.043	0.010	0.341	0.337	0.274	0.070
β_{221}	0.015	0.007	0.012	0.010	0.178	0.285	0.229	0.123
β_{222}	0.183	0.191	0.068	0.022	0.690	0.730	0.427	0.104
β_{311}	0.004	0.006	0.002	0.003	0.069	0.071	0.055	0.054
β_{312}	0.004	0.000	0.003	0.003	0.068	0.074	0.057	0.051
β_{321}	0.009	0.011	0.007	0.006	0.090	0.096	0.065	0.065
β_{322}	0.036	0.034	0.006	0.007	0.162	0.177	0.046	0.042
Irrelevant regressors								
β_{113}	-	0.004	0.001	-	-	0.055	0.048	-
β_{123}	-	0.002	0.000	-	-	0.118	0.069	-
β_{213}	-	0.174	0.031	-	-	0.375	0.189	-
β_{223}	-	0.073	0.014	-	-	0.204	0.141	-
β_{313}	-	0.029	0.022	-	-	0.147	0.212	-
β_{323}	-	0.107	0.078	-	-	0.473	0.335	-
$\epsilon = 0.35$								
β_{111}	0.047	0.079	0.009	0.002	0.145	0.152	0.190	0.106
β_{112}	0.219	0.373	0.148	0.060	0.311	0.477	0.435	0.164
β_{121}	0.069	0.036	0.012	0.021	0.250	0.278	0.396	0.133
β_{122}	0.122	0.042	0.078	0.060	0.208	0.246	0.751	0.147
β_{211}	0.012	0.034	0.045	0.015	0.204	0.254	0.245	0.200
β_{212}	0.185	0.248	0.205	0.055	0.344	0.443	0.574	0.271
β_{221}	0.018	0.010	0.008	0.002	0.233	0.325	0.303	0.196
β_{222}	0.138	0.195	0.073	0.037	0.581	0.803	0.501	0.163
β_{311}	0.001	0.018	0.002	0.000	0.068	0.158	0.063	0.061
β_{312}	0.002	0.013	0.003	0.000	0.072	0.101	0.060	0.061
β_{321}	0.014	0.031	0.008	0.008	0.073	0.134	0.069	0.066
β_{322}	0.003	0.028	0.005	0.005	0.081	0.230	0.045	0.046
Irrelevant regressors								
β_{113}	-	0.010	0.002	-	-	0.075	0.040	-
β_{123}	-	0.056	0.013	-	-	0.199	0.073	-
β_{213}	-	0.333	0.223	-	-	0.401	0.500	-
β_{223}	-	0.149	0.020	-	-	0.310	0.484	-
β_{313}	-	0.128	0.138	-	-	0.342	0.506	-
β_{323}	-	0.108	0.029	-	-	0.496	0.441	-

Table 4.11: Bias and RMSE for the regression coefficients β_{kmp} under the four types of models in the third process ($I = 1000$).

	Bias				RMSE			
	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW
$\epsilon = 0.55$								
β_{111}	0.034	0.029	0.003	0.000	0.085	0.071	0.034	0.030
β_{112}	0.119	0.168	0.017	0.002	0.256	0.345	0.114	0.046
β_{121}	0.043	0.030	0.003	0.002	0.113	0.114	0.055	0.048
β_{122}	0.060	0.015	0.005	0.002	0.153	0.107	0.077	0.068
β_{211}	0.015	0.002	0.007	0.001	0.125	0.159	0.057	0.026
β_{212}	0.121	0.144	0.012	0.001	0.317	0.349	0.096	0.035
β_{221}	0.017	0.012	0.005	0.000	0.140	0.218	0.056	0.043
β_{222}	0.182	0.137	0.027	0.014	0.589	0.603	0.093	0.062
β_{311}	0.001	0.001	0.002	0.002	0.044	0.046	0.037	0.037
β_{312}	0.009	0.010	0.008	0.006	0.042	0.048	0.033	0.032
β_{321}	0.013	0.012	0.008	0.008	0.074	0.080	0.045	0.045
β_{322}	0.006	0.007	0.001	0.002	0.076	0.080	0.035	0.033
Irrelevant regressors								
β_{113}	-	0.002	0.001	-	-	0.037	0.033	-
β_{123}	-	0.005	0.003	-	-	0.085	0.043	-
β_{213}	-	0.172	0.018	-	-	0.343	0.126	-
β_{223}	-	0.097	0.015	-	-	0.205	0.094	-
β_{313}	-	0.047	0.004	-	-	0.126	0.057	-
β_{323}	-	0.015	0.012	-	-	0.273	0.089	-
$\epsilon = 0.35$								
β_{111}	0.025	0.043	0.022	0.001	0.076	0.089	0.087	0.038
β_{112}	0.173	0.307	0.099	0.014	0.245	0.395	0.270	0.099
β_{121}	0.109	0.081	0.026	0.024	0.156	0.150	0.114	0.084
β_{122}	0.117	0.043	0.002	0.015	0.159	0.137	0.128	0.093
β_{211}	0.016	0.041	0.031	0.002	0.216	0.222	0.192	0.134
β_{212}	0.247	0.286	0.096	0.015	0.368	0.417	0.253	0.095
β_{221}	0.021	0.027	0.032	0.016	0.187	0.168	0.124	0.080
β_{222}	0.212	0.181	0.020	0.006	0.659	0.669	0.413	0.080
β_{311}	0.006	0.006	0.005	0.004	0.046	0.046	0.038	0.037
β_{312}	0.010	0.011	0.009	0.005	0.044	0.044	0.038	0.036
β_{321}	0.009	0.007	0.001	0.000	0.077	0.078	0.044	0.043
β_{322}	0.002	0.001	0.005	0.001	0.078	0.080	0.038	0.033
Irrelevant regressors								
β_{113}	-	0.003	0.007	-	-	0.032	0.062	-
β_{123}	-	0.008	0.013	-	-	0.079	0.050	-
β_{213}	-	0.314	0.106	-	-	0.371	0.248	-
β_{223}	-	0.138	0.066	-	-	0.268	0.190	-
β_{313}	-	0.110	0.065	-	-	0.239	0.175	-
β_{323}	-	0.036	0.009	-	-	0.266	0.113	-

Table 4.12: Classification recovery of the fitted SuNCW, NCW, CNCW and SuCNCW models with $K = 3$: average values (standard deviations) of the ARI index over 100 samples.

I	Process	ϵ	SuNCW	NCW	CNCW	SuCNCW
500	I	0.55	0.988 (0.009)	0.988 (0.009)	0.988 (0.009)	0.988 (0.009)
	I	0.35	0.949 (0.017)	0.945 (0.037)	0.945 (0.037)	0.949 (0.018)
	II	0.55	0.921 (0.089)	0.915 (0.093)	0.939 (0.078)	0.954 (0.039)
	II	0.35	0.799 (0.119)	0.770 (0.123)	0.848 (0.112)	0.882 (0.077)
	III	0.55	0.848 (0.141)	0.838 (0.146)	0.911 (0.085)	0.923 (0.060)
	III	0.35	0.663 (0.108)	0.639 (0.095)	0.744 (0.131)	0.804 (0.108)
1000	I	0.55	0.988 (0.005)	0.988 (0.006)	0.988 (0.006)	0.988 (0.005)
	I	0.35	0.954 (0.010)	0.953 (0.010)	0.953 (0.010)	0.954 (0.010)
	II	0.55	0.938 (0.060)	0.935 (0.060)	0.962 (0.038)	0.966 (0.010)
	II	0.35	0.805 (0.127)	0.781 (0.123)	0.858 (0.119)	0.892 (0.079)
	III	0.55	0.855 (0.145)	0.850 (0.146)	0.930 (0.051)	0.938 (0.015)
	III	0.35	0.678 (0.122)	0.665 (0.110)	0.804 (0.116)	0.845 (0.080)

Classification recovery

The study of the aspect *(iii)* has required an evaluation of the agreement between the partitions of the sample units detected by the four types of models and the true partition. To this end, the adjusted Rand index (ARI) (Hubert and Arabie, 1985) has been employed. Average values and standard deviations of this index (over the 100 datasets) for the four model classes under the three data generation processes by the examined levels of the two experimental factors are reported in Table 4.12. When the analysed datasets do not contain atypical observations, the classification recovery of all model classes is almost perfect with both levels of separation and both sample sizes ($ARI \geq 0.945$). The results obtained under the second and third processes show that the classification recovery associated with the use of all models increases with the level of separation between the second and third components for each value of I; it also increases with the sample size for each value of ϵ. With datasets generated using these processes, SuCNCW models are characterised by the greatest ability to properly estimate the true classification of the sample observations for each examined level of the two experimental factors. The partitions obtained from SuCNCW models also show a good agreement with the true partitions ($0.804 \leq ARI \leq 0.966$). Among the other three model classes, CNCW models outperforms both SuNCW and NCW models. For these two latter models, the classification recovery appears to be markedly lower, especially with the lowest level of separation ($0.639 \leq ARI \leq 0.663$ with $I = 500$, $0.665 \leq ARI \leq 0.678$ with $I = 1000$).

Trade-off between fit and complexity

In order to study the aspect *(iv)*, for each dataset and each model class, the models of order $\hat{K}_{IC}$ have been selected, where IC denotes an information criterion ($IC \in \{BIC, ICL_1, ICL_2\}$) and $\hat{K}_{IC} = \arg\max IC(K)$ for $K \in \{1, 2, 3, 4\}$. Then, for each information criterion and each dataset, the four values of $IC(\hat{K}_{IC})$ associated with the four examined model classes have been compared, and the model with the highest IC value has been selected as the most adequate fitted model. Table 4.13 provides the frequency distribution of the models selected in this way by each IC for each data generation process and each value of ϵ and I. As expected, SuNCW models have almost always been selected as the most adequate for the analysis of uncontaminated datasets. With datasets containing atypical observations generated through the second and the third processes, best trade-off between fit and complexity is generally obtained by the fitted SuCNCW models. Such results hold true regardless of the level of separation and the information criterion employed to perform model selection.

Comparison among information criteria

Information on the aspect *(v)* has been obtained by evaluating the number of times each value of K has been selected by each examined criterion. The obtained results are reported in Tables 4.14 and 4.15. When the analysed datasets do not contain atypical observations and the level of separation between the second and third cluster is high (first process, $\epsilon = 0.55$), the presence of three clusters is (almost) always recognised by all the examined information criteria regardless of the fitted model and the sample size (see the upper part of Tables 4.14 and 4.15). If the level of separation is reduced ($\epsilon = 0.35$), the ability of the BIC to correctly detect the presence of three clusters remains good regardless of the fitted model only with the largest sample size. When $I = 500$, the true order of the generated datasets is slightly underestimated by the BIC when CNCW models are employed. With datasets generated using the first process, ICL_1 and ICL_2 show a clear preference for $K = 3$ components regardless of the model type with both values of ϵ but only when the sample size is $I = 1000$. Otherwise, the true value of K is almost always properly estimated by these two criteria as long as models embedding the information on the relevant regressors are fitted (e.g., SuNCW and SuCNCW). With the other two examined models, the true number of clusters appears to be underestimated, and this is especially true of CNCW models.

Under the second and third processes, when SuNCW and NCW models are fitted to the

Table 4.13: Trade-off between fit and compexity: number of selections over 100 samples of SuNCW, NCW, CNCW and SuCNCW models, based on the highest BIC, ICL_1 and ICL_2.

I	IC	Process	ϵ	SuNCW	NCW	CNCW	SuCNCW
500	BIC	I	0.55	100	0	0	0
		I	0.35	100	0	0	0
		II	0.55	0	1	1	98
		II	0.35	1	2	2	95
		III	0.55	2	1	1	96
		III	0.35	0	0	8	92
	ICL_1	I	0.55	100	0	0	0
		I	0.35	99	1	0	0
		II	0.55	0	1	1	98
		II	0.35	1	2	2	95
		III	0.55	2	2	1	95
		III	0.35	2	0	6	92
	ICL_2	I	0.55	99	1	0	0
		I	0.35	99	1	0	1
		II	0.55	0	0	2	98
		II	0.35	0	2	3	95
		III	0.55	2	2	1	95
		III	0.35	2	0	7	91
1000	BIC	I	0.55	100	0	0	0
		I	0.35	100	0	0	0
		II	0.55	0	0	0	100
		II	0.35	3	1	4	92
		III	0.55	0	0	1	99
		III	0.35	0	0	8	92
	ICL_1	I	0.55	100	0	0	0
		I	0.35	99	1	0	0
		II	0.55	0	0	0	100
		II	0.35	4	1	6	89
		III	0.55	0	0	1	99
		III	0.35	0	0	6	94
	ICL_2	I	0.55	100	0	0	0
		I	0.35	98	2	0	0
		II	0.55	0	0	0	100
		II	0.35	3	1	7	89
		III	0.55	0	0	1	99
		III	0.35	1	1	6	92

data, the BIC shows a tendency to select $K = 4$ (outliers are typically accommodated using an additional cluster), regardless of the level of separation (see also Mazza and Punzo, 2020). This tendency appears to be more evident when the sample size is large. ICL_1 shows the same behaviour in association with SuNCW models (for both levels of separation) and NCW models (for $\epsilon = 0.55$). On the contrary, with SuCNCW models, BIC and ICL_1 correctly identify three clusters for the majority of the simulated datasets, regardless of the sample size and the degree of separation. When these two criteria are employed in the selection of CNCW models, the true value of K is properly estimated provided that the degree of separation is high or the sample size is large; otherwise, the order of CNCW models is generally underestimated. As far as ICL_2 is concerned, CNCW and SuCNCW models of order $K = 3$ are almost always selected with $\epsilon = 0.55$ regardless of the sample size; however, when the degree of separation is low, the order of the CNCW models is generally underestimated. With SuNCW and NCW models, ICL_2 shows a clear tendency to overestimate the true K when the analysed datasets have a large sample size. With $I = 500$ and $\epsilon = 0.35$, SuNCW and NCW models of order $K = 2$ are generally selected by using the ICL_2. These latter results may depend on the penalty employed by ICL_2 (a function of the uncertainty of the estimated posterior probabilities $\hat{z}_{ik}$), which is the most severe and is also expected to be particularly large whenever the generated datasets contain poorly separated clusters.

Classification recovery (without exploiting the knowledge of K)

In order to study the aspect *(vi)*, for each generated dataset the ARI index has been computed between the partitions of the sample units detected by the fitted models showing the highest BIC value under each competing model class and the true partition. In general, the resulting average values of the ARI index (see Table 4.16) are quite similar to the ones obtained by exploiting the knowledge of the true K (see Table 4.12). Obviously, whenever the value of K determined according to the BIC is equal to the true K, the ability to recover the true classification coincides with the one evaluated in Section 4.3.2. Thus, in general, using the BIC to estimate the value of K seems to have a negligible impact on the classification recovery of SuCNCW models. The impact on the performance of the other three model types is more evident, especially for SuNCW and NCW models. More specifically, in the presence of datasets with contaminated observations generated according to the second and third processes, SuNCW and NCW models of order $\hat{K}_{BIC}$ show a slight increase in the ability to estimate the true

Table 4.14: Comparison among information criteria: number of selections over 100 samples for SuNCW, NCW, CNCW and SuCNCW models of order $K \in \{1, 2, 3, 4\}$ ($I = 500$).

K	$BIC(K)$				$ICL_1(K)$				$ICL_2(K)$			
	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW
I process, $\epsilon = 0.55$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	99	99	99	99	99	99	99	99	99	99	99	99
4	1	1	1	1	1	1	1	1	1	1	1	1
I process, $\epsilon = 0.35$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	1	5	19	3	4	15	38	7	6	31	63	12
3	98	95	81	97	96	85	62	93	94	69	37	88
4	1	0	0	0	0	0	0	0	0	0	0	0
II process, $\epsilon = 0.55$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	0	0	1	1	0	1	2	2	0
3	15	23	94	98	27	36	94	98	59	66	95	98
4	85	75	5	2	73	63	5	2	40	32	3	2
II process, $\epsilon = 0.35$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	7	22	40	6	22	47	69	14	47	79	91	31
3	24	24	60	93	23	17	31	85	19	6	9	69
4	69	54	0	1	55	36	0	1	34	15	0	0
III process, $\epsilon = 0.55$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	2	2	1	2	4	2	1	2	4	2	1
3	30	34	91	95	37	40	92	96	50	49	93	96
4	70	64	7	4	61	56	6	3	48	47	5	3
III process, $\epsilon = 0.35$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	5	25	57	21	28	51	75	40	53	73	88	71
3	31	33	42	77	24	28	24	59	21	15	11	28
4	64	42	1	2	48	21	1	1	26	12	1	1

Table 4.15: Comparison among information criteria: number of selections over 100 samples for SuNCW, NCW, CNCW and SuCNCW models of order $K \in \{1, 2, 3, 4\}$ ($I = 1000$).

K	$BIC(K)$				$ICL_1(K)$				$ICL_2(K)$			
	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW	SuNCW	NCW	CNCW	SuCNCW
I process, $\epsilon = 0.55$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0
3	100	100	100	100	100	100	100	100	100	100	100	100
4	0	0	0	0	0	0	0	0	0	0	0	0
I process, $\epsilon = 0.35$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	2	3	4	3	4	5	6	3
3	97	97	97	97	97	96	95	97	96	94	94	97
4	3	3	3	3	1	1	1	0	0	1	0	0
II process, $\epsilon = 0.55$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	1	1	0	0
3	3	4	98	100	3	5	98	100	7	12	98	100
4	97	96	2	0	97	95	2	0	92	87	2	0
II process, $\epsilon = 0.35$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	16	3	0	0	25	8	6	27	54	15
3	12	16	79	89	15	17	75	89	12	10	46	85
4	88	84	5	8	85	83	0	3	82	63	0	0
III process, $\epsilon = 0.55$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	1	0	0	0	1	0	0	0	1	0
3	0	1	98	99	5	5	98	99	12	14	98	100
4	100	99	1	1	95	95	1	1	88	86	1	0
III process, $\epsilon = 0.35$												
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	12	1	4	11	31	5	21	43	78	35
3	8	12	76	90	15	24	68	90	16	20	21	63
4	92	88	12	9	81	65	1	5	63	37	1	2

Table 4.16: Classification recovery of the fitted SuNCW, NCW, CNCW and SuCNCW models with the highest BIC: average values (standard deviations) of the ARI index over 100 samples.

I	Process	ϵ	SuNCW	NCW	CNCW	SuCNCW
500	I	0.55	0.987 (0.012)	0.987 (0.011)	0.987 (0.011)	0.987 (0.012)
	I	0.35	0.945 (0.035)	0.932 (0.068)	0.891 (0.119)	0.940 (0.055)
	II	0.55	0.891 (0.048)	0.890 (0.059)	0.952 (0.038)	0.957 (0.019)
	II	0.35	0.805 (0.091)	0.767 (0.111)	0.791 (0.134)	0.886 (0.068)
	III	0.55	0.870 (0.071)	0.868 (0.078)	0.921 (0.050)	0.923 (0.043)
	III	0.35	0.749 (0.102)	0.682 (0.108)	0.711 (0.123)	0.803 (0.106)
1000	I	0.55	0.988 (0.005)	0.988 (0.006)	0.988 (0.006)	0.988 (0.005)
	I	0.35	0.952 (0.015)	0.951 (0.015)	0.951 (0.146)	0.952 (0.015)
	II	0.55	0.885 (0.034)	0.885 (0.034)	0.965 (0.011)	0.966 (0.010)
	II	0.35	0.827 (0.077)	0.819 (0.085)	0.861 (0.109)	0.898 (0.061)
	III	0.55	0.880 (0.021)	0.879 (0.023)	0.934 (0.037)	0.938 (0.016)
	III	0.35	0.769 (0.102)	0.737 (0.122)	0.834 (0.083)	0.865 (0.032)

classification of the sample observations in comparison with the same models of order 3. A possible explanation of this behaviour could be related to the fact that such models are not able to properly account for contaminated observations; thus, according to the BIC, SuNCW and NCW models of order 4 should be preferred.

4.4 Analysis of canned tuna sales

The `tuna` dataset (Chevalier et al., 2003), which is available in the `R` package `bayesm` (Rossi, 2012), contains information about the volume of weekly sales (`Move`) for seven of the top 10 U.S. brands in the canned tuna product category for $I = 338$ weeks between September 1989 and May 1997. The same dataset also provides information about measures of the display activity (`Nsale`) and log price (`Lprice`) of each brand in each week. The dependence of log sales (`Lmove`) on log prices and promotional activites for some brands selected from this dataset has been already studied through either clusterwise linear regression models or cluster-weighted models (see, e.g., Galimberti et al., 2016; Galimberti and Soffritti, 2020; Diani et al., 2022). Such studies showed that the analysed dependencies are characterised by heterogeneity over time. They were also able to highlight some weeks (from week no. 58 to weeks no. 73/74) in which the volume of weekly sales for one brand (Bumble Bee) were affected by a worldwide boycott campaign because that brand was found to be buying yellow-fin tuna caught by dolphin-unsafe techniques (Baird and Quastel, 2011). In a recent research conducted on the brands Star Kist 6 oz. (SK) and Bumble Bee Solid 6.12 oz. (BBS) through mixtures of contaminated linear regression models with fixed covariates (Perrone and Soffritti, 2023), some atypical observations in the $\mathbf{y}$-direction

Table 4.17: Pearson's correlation matrix (lower diagonal part) and p-values of the Student's t test (upper diagonal part) for the hypotheses of pairwise linear independence between six variables from the `tuna` dataset.

	`Lmove SK`	`Lmove BBS`	`Nsale SK`	`Lprice SK`	`Nsale BBS`	`Lprice BBS`
`Lmove SK`	1.0000	0.0181	$< 10^{-20}$	$< 10^{-52}$	0.2267	0.4000
`Lmove BBS`	−0.1285	1.0000	0.1575	0.4734	$< 10^{-10}$	$< 10^{-9}$
`Nsale SK`	0.4757	0.0771	1.0000	$< 10^{-36}$	0.0174	0.3831
`Lprice SK`	−0.7067	0.0391	−0.6139	1.0000	0.0043	0.3808
`Nsale BBS`	0.0659	0.3256	0.1293	−0.1550	1.0000	$< 10^{-52}$
`Lprice BBS`	−0.0459	−0.3172	−0.0476	0.0478	−0.7050	1.0000

were also detected.

In line with this latter study, the analysis illustrated here has been focused on the SK and BBS products. More specifically, the following vectors of variables have been considered: $\mathbf{Y} = (Y_1 =$ `Lmove SK`, $Y_2 =$ `Lmove BBS`$)$, $\mathbf{X} = (X_1 =$ `Nsale SK`, $X_2 =$ `Lprice SK`, $X_3 =$ `Nsale BBS`, $X_4 =$ `Lprice BBS`$)$. Thus, $M = 2$ and $P = 4$. A preliminary evaluation of the pairwise linear dependencies for such variables has been carried out (see Table 4.17). For each brand, log sales result to be negatively correlated with the log prices (−0.7067 for SK, −0.3172 for BBS) and positively correlated with the display activities (0.4757 for SK, 0.3256 for BBS). A negative correlation also emerges between `Nsale SK` and `Lprice SK` (−0.6139) and between `Nsale BBS` and `Lprice BBS` (−0.7050). Lower but significant (for $\alpha = 0.05$) pairwise linear dependencies characterise `Lmove SK` and `Lmove BBS` (−0.1285), `Nsale SK` and `Nsale BBS` (0.1293), and `Lprice SK` and `Nsale BBS` (−0.1550).

The dataset containing the information about these six variables for the 338 weeks has been analysed with NCW, CNCW, SuNCW and SuCNCW models of order K, with $K \in \{1, 2, \ldots, 9\}$ and with each of the parameterisations illustrated in Section 4.2.7. NCW and CNCW models have been fitted by assuming that prices and promotional activities for one product may also have an impact on the sales of the other product; thus, $\mathbf{X}$ is the vector of the covariates employed for the prediction of both responses. As far as the SuNCW and SuCNCW models are concerned, the selection of the regressors to be employed in the linear predictors of `Lmove SK` and `Lmove BBS` has been carried out by exploiting the results of an exhaustive search of the relevant regressors for such responses reported in Perrone and Soffritti (2023). That search demonstrated that the log unit sales of SK canned tuna should be regressed on the log prices and the promotional activities of the same brand; as far as the BBS log sales are concerned, they should be regressed on the log prices of both brands and the promotional activities of BBS. Thus, for the SuNCW

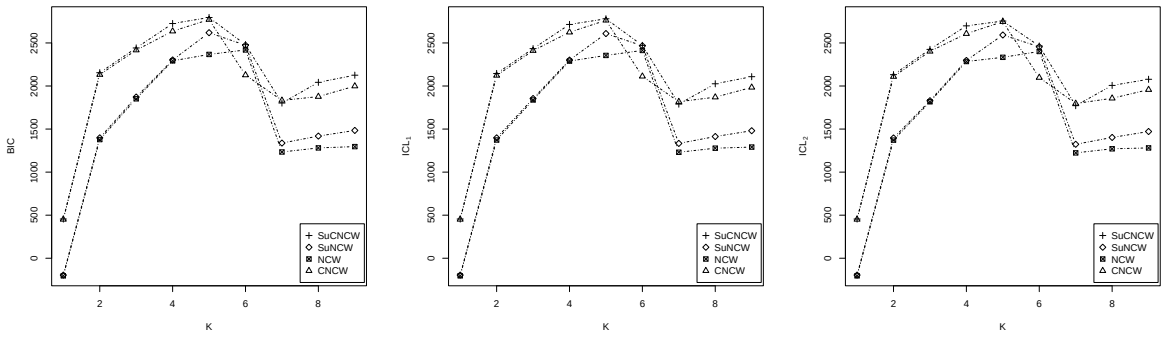

Figure 4.1: Values of BIC, ICL_1 and ICL_2 for the best NCW, CNCW, SuNCW and SuCNCW models by number of components in the analysis of tuna sales.

Table 4.18: Maximised log-likelihood $\ell(\hat{\boldsymbol{\psi}})$ and values of BIC, ICL_1 and ICL_2 for the best models selected from the classes SuCNCW, SuNCW, CNCW and NCW in the analysis of tuna sales.

Class	K	acr.X	acr.Y	$\ell(\hat{\boldsymbol{\psi}})$	$n_{\boldsymbol{\psi}}$	BIC	ICL_1	ICL_2
SuCNCW	5	`VEV`	`EEE`	1747.1	120	2795.5	2780.5	2754.2
SuNCW	5	`VEV`	`VEV`	1624.0	108	2619.1	2608.9	2592.4
CNCW	5	`VEV`	`EVE`	1790.4	139	2771.3	2762.8	2746.8
NCW	6	`VEV`	`EVE`	1622.9	142	2419.0	2413.1	2401.0

and SuCNCW models, the two sub-vectors of $\mathbf{X}$ employed to define the design matrix are $\mathbf{X}_1 = (X_1 = \texttt{Nsale SK}, X_2 = \texttt{Lprice SK})$ and $\mathbf{X}_2 = (X_2 = \texttt{Lprice SK}, X_3 = \texttt{Nsale BBS}, X_4 = \texttt{Lprice BBS})$. The overall number of fitted models from each of the four examined model classes is 1577.

Figure 4.1 shows the values of BIC, ICL_1 and ICL_2 for the best fitted models from each class by K. The best trade-off between the fit and the model complexity is reached by SuCNCW, SuNCW, CNCW models with $K = 5$ and a NCW model with $K = 6$, regardless of the model selection criterion. More detailed information about these models can be found in Table 4.18. According to all model selection criteria, the overall best trade-off is reached by the SuCNCW model with $K = 5$. The distributions of the four regressors in the five clusters of weeks detected by this model are ellipsoidal with variable volumes and orientations and equal shape; as far as the joint conditional distributions of the two responses given the corresponding regressors are concerned, clusters are characterized by equal distribution, volume and shape. The convergence of the ECM algorithm for the estimation of this model has been reached after 136 iterations. The obtained estimates of $\boldsymbol{\pi}$, $\boldsymbol{\alpha}$, $\boldsymbol{\eta}$, $\boldsymbol{\tau}$, $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ and $\boldsymbol{\beta}^*$ are reported in Table 4.19. By focusing the attention on the estimated regression coefficients, it emerges that the effects of prices for either

Table 4.19: Estimated $\boldsymbol{\pi}$, $\boldsymbol{\alpha}$, $\boldsymbol{\eta}$, $\boldsymbol{\tau}$, $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ and $\boldsymbol{\beta}^*$ of the overall best model for the `tuna` dataset.

k	1	2	3	4	5
$\hat{\pi}_k$	0.090	0.151	0.151	0.260	0.348
$\hat{\alpha}_k$	0.506	0.999	0.997	0.982	0.922
$\hat{\eta}_k$	6.942	1.001	1.172	11.213	10.853
$\hat{\tau}_k$	0.999	0.999	0.611	0.900	0.865
$\hat{\lambda}_k$	1.001	1.001	9.714	132.726	116.684
$\hat{\mu}_{k1}$	0.001	0.323	0.597	0.647	0.000
$\hat{\mu}_{k2}$	−0.194	−0.229	−0.320	−0.281	− 0.141
$\hat{\mu}_{k3}$	0.227	0.996	0.693	0.003	0.001
$\hat{\mu}_{k4}$	0.547	0.500	0.497	0.572	0.566
$\hat{\beta}_{k10}$	8.801	8.506	8.547	8.620	8.804
$\hat{\beta}_{k11}$	15.194	0.345	0.031	0.225	−13.874
$\hat{\beta}_{k12}$	−1.563	−3.394	−3.850	−3.549	−2.378
$\hat{\beta}_{k20}$	9.550	9.868	11.136	8.581	9.032
$\hat{\beta}_{k21}$	−0.166	0.299	0.503	−0.041	0.541
$\hat{\beta}_{k22}$	−0.242	0.959	0.199	0.491	4.830
$\hat{\beta}_{k23}$	−3.061	−5.234	−6.009	−1.548	−2.105

Table 4.20: Sizes of the five clusters of weeks detected by the overall best model and their within-cluster distributions into four categories, based on $\hat{u}_{ik}$ and $\hat{v}_{ik}$.

Cluster k	typical	outlier	bad leverage	good leverage	Cluster size
1	17	0	0	14	31
2	53	0	0	0	53
3	33	15	0	0	48
4	79	8	1	0	88
5	94	16	0	8	118

brands on the log unit sales of the same brand are negative within all the clusters detected by the model. Shifting attention towards the estimates of $\boldsymbol{\mu}_k$, for $k = 1, \ldots, 5$, the five clusters of weeks show similar estimated mean values for `Lprice BBS`. However, from an overall inspection of these estimates it also seems that both the joint distribution of prices and promotional activities and the conditional distribution of tuna sales for both brands are affected by a source of unobserved heterogeneity over time. Furthermore, the values of $\hat{\alpha}_k$, $\hat{\eta}_k$, $\hat{\tau}_k$ and $\hat{\lambda}_k$, for $k = 1, \ldots, 5$, seem to suggest that the analysed dataset is also contaminated by the presence of leverage points (in clusters 1, 4 and 5) and regression outliers (in clusters 3, 4 and 5).

Table 4.20 reports the sizes of the five clusters of weeks determined by the best fitted model according to the rule of the maximum a posteriori probability; it also shows the within-cluster sizes of the following four categories of weeks: typical observations ($\hat{u}_{ik} \geq 0.5$ and $\hat{v}_{ik} \geq 0.5$); regression outliers ($\hat{u}_{ik} < 0.5$ and $\hat{v}_{ik} \geq 0.5$); good leverage points ($\hat{u}_{ik} \geq 0.5$ and $\hat{v}_{ik} < 0.5$); bad

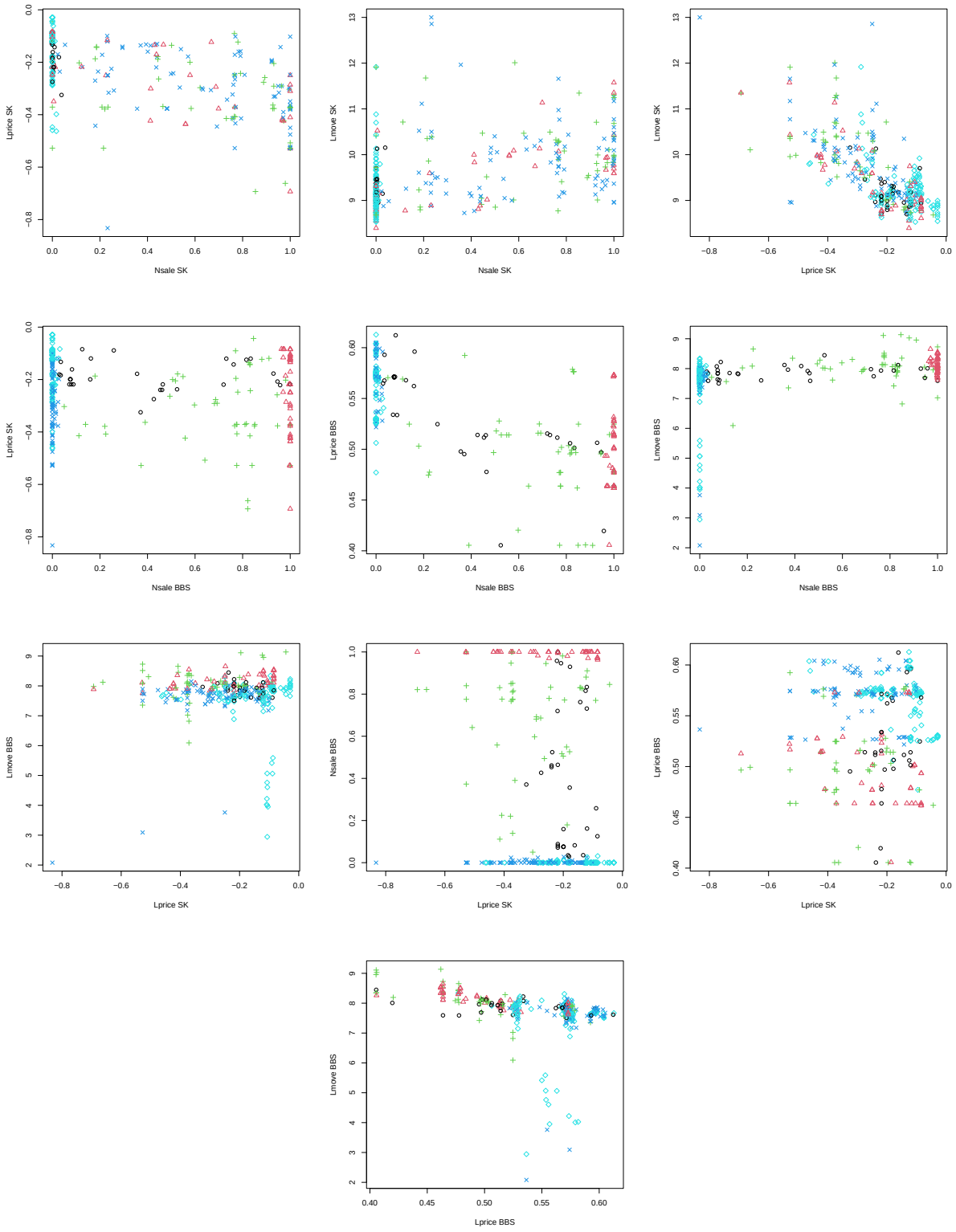

Figure 4.2: Scatterplots for pairs of variables from the analysis of tuna sales. Weeks are pictured with five different colours and symbols according to the classification obtained from the best model.

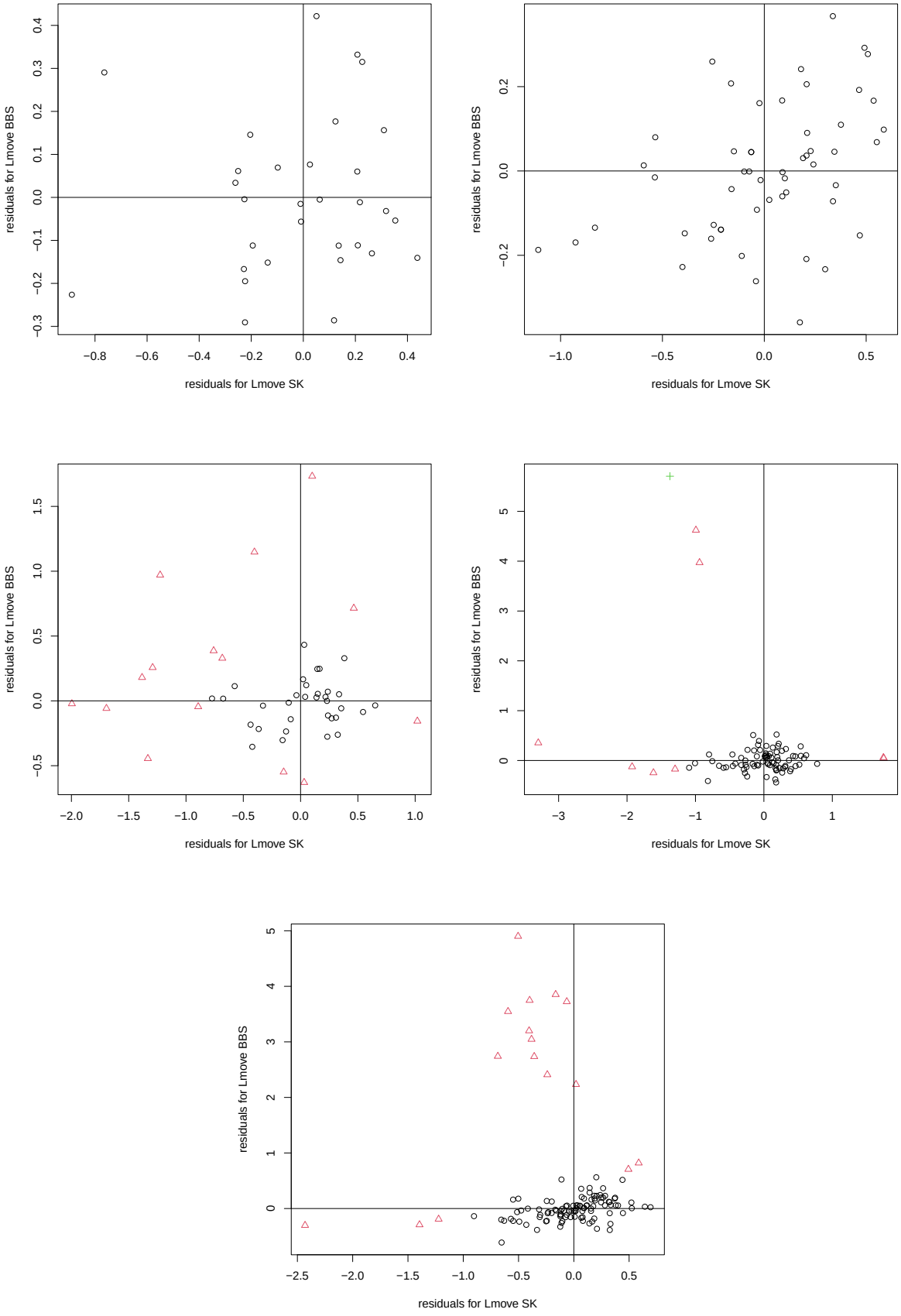

Figure 4.3: Scatterplots of the estimated sample residuals $\mathbf{y}_i - \tilde{\mathbf{x}}_i^* \hat{\boldsymbol{\beta}}_k^*$ $(k = 1, \ldots, 5)$ for the weeks assigned to the five clusters detected by the best model for the analysis of tuna sales. Black circles and red triangles correspond to typical and outlying weeks, respectively. The green plus denotes a bad leverage point.

leverage points ($\hat{u}_{ik} < 0.5$ and $\hat{v}_{ik} < 0.5$). The first cluster detected by the best fitted model contains 31 weeks (see the black circle in the scatterplots of Figure 4.2). Almost half of these weeks have been classified as good leverage points (see the first row of Table 4.20). New Year 1992, Memorial Day 1993, New Year 1994, Halloween 1995 are the special events associated with such weeks. As far as the possible presence of regression outliers in the distribution of $\mathbf{Y}|(\mathbf{X} = \mathbf{x}, G_k)$ is concerned, all the weeks belonging to this cluster can be considered as typical. This latter result is also evident from the estimated sample residuals $\mathbf{y}_i - \tilde{\mathbf{x}}_i^* \hat{\boldsymbol{\beta}}_1^*$ for the weeks of this cluster (see the scatterplot on the left in the upper panel of Figure 4.3). A further proof is given by the low values of $\hat{d}^2_{i1\mathbf{y}}$ (see the first column in Table 4.21). The estimated mean promotional activities in this cluster are quite low, especially those for the SK brand. Furthermore, the estimated effects of promotional activities on sales are positive (and particularly strong) for SK, while they result to be negative and negligible for BBS (see the first column in Table 4.19). Finally, promotional activities of SK tuna appear to be highly homogenous, as suggested by the low estimated variance of `Nsale SK` for the weeks of this cluster (not reported here). The second cluster, which is composed of 53 weeks (labelled using a red triangle point up in the scatterplot of Figure 4.2), only contain typical observations (see the second row in Table 4.20). Similarly to the previous cluster, the estimated sample residuals $\mathbf{y}_i - \tilde{\mathbf{x}}_i^* \hat{\boldsymbol{\beta}}_2^*$ (see the scatterplot on the right in the upper panel of Figure 4.3) and low values of $\hat{d}^2_{i2\mathbf{y}}$ (see the second column in Table 4.21) prove the absence of outlying weeks. Furthermore, this cluster is mainly characterized by the highest estimated mean value of promotional activities for BBS tuna (column 2 in Table 4.19) and the highest variance for `Lprice BBS` (not reported here). Cluster 3 comprises 48 weeks (green plus in Figure 4.2). It is characterized by the absence of leverage points. However, 15 weeks of this cluster have been classified as mild outliers (see the red triangles in the scatterplot on the left in the central panel of Figure 4.3). They are mostly associated with holidays and special events that took place between 1990 and 1992 (weeks close to Easter 1990; Labor Day 1990; weeks close to Halloween 1990; weeks close to Labor Day 1991 and Halloween 1991; Christmas 1991; President Days 1992; Easter 1992) or with the first period of the boycott campaign (weeks from no. 58 to week no. 60). It is worth noting that, for these 15 weeks, the estimated Mahalanobis squared distances $\hat{d}^2_{i3\mathbf{y}}$ are clearly larger than those computed for the other weeks of the same cluster (see the third column in Table 4.21). Overall, the weeks belonging to this cluster show the lowest mean prices of both brands (see the third column in Table 4.19). Furthermore, the estimated effect of promotional activities for SK tuna on the sales of the same brand is negligible.

Table 4.21: Minimum and maximum values of the estimated squared Mahalanobis distances $\hat{d}^2_{ik\mathbf{y}}$ within the five clusters of weeks, by the categories: typical observations/outliers.

k	1	2	3	4	5
Typical observations					
$\min\{\hat{d}^2_{ik\mathbf{y}}\}$	0.01	0.01	0.03	0.01	0.01
$\max\{\hat{d}^2_{ik\mathbf{y}}\}$	0.47	11.26	5.97	10.85	11.73
Outliers					
$\min\{\hat{d}^2_{ik\mathbf{y}}\}$	-	-	7.38	15.32	13.64
$\max\{\hat{d}^2_{ik\mathbf{y}}\}$	-	-	82.80	982.52	691.09

Finally, the weeks belonging to this cluster are characterised by the strongest effect of prices on sales for both brands. As far as cluster 4 is concerned, it contains 88 weeks (dark blue cross in Figure 4.2). 8 of these weeks have been classified as outliers (see the scatterplot on the right in the central panel of Figure 4.3); two of them (weeks no. 71 and no. 72) belong to the period of the boycott campaign. The last week from the period of the boycott campaign (week no. 74) has been detected as a bad leverage (see the green plus in the fourth scatterplot of Figure 4.3). Similarly to the previous cluster, the estimated Mahalanobis squared distances $\hat{d}^2_{i4\mathbf{y}}$ for the 8 outlying weeks of this cluster result to be larger than those computed for the other weeks (see Table 4.21). The main distinctive feature of cluster 4 is the highest estimated mean price of BBS tuna and the highest estimated mean value of promotional activities for SK tuna. Furthermore, promotional activities of BBS tuna are instead negligible (see Table 4.19). Finally, the 88 weeks of this cluster are also characterized by highly homogeneous prices and promotional activities of both brands. Cluster 5 is composed of 118 weeks (sky-blue diamond in Figure 4.2); 8 and 16 of these weeks have been detected as good leverage points and outliers, respectively. The outlying weeks of this cluster can also be easily identified from the scatterplot on the bottom part of Figure 4.3); they also show the largest within-cluster Mahalanobis squared distances $\hat{d}^2_{i5\mathbf{y}}$ (see Table 4.21). 10 of these 16 outlying weeks correspond with the central period of the boycott campaign (weeks from no. 61 to no. 70). The 118 weeks of this cluster are characterised by the lowest estimated mean values of the promotional activities for both brands. Furthermore, they also show a quite high estimated mean price of BBS tuna. Finally, the effects of promotional activities on sales are negative (and particularly strong) for SK and positive for BBS.

4.5 Conclusions

The SuCNCW models introduced here allow to perform robust clustering in multivariate linear regression analysis with correlated responses and random regressors for datasets characterised by unobserved heterogeneity and mildly atypical observations. They can also be employed to identify outliers and leverage points within each detected cluster. The main novelty of these models in reference with the ones introduced by Punzo and McNicholas (2017) is that a different vector of regressors is considered for each response. Thanks to this feature, the data analyst is enabled to convey prior information concerning the absence of certain regressors from the linear term employed in the prediction of a certain response in any application in which different regressors are expected to be relevant in the prediction of different responses. SuCNCW models with a reduced number of variance-covariance parameters have also been specified; they can be more effectively employed when the analysis involves either many responses or many regressors. Furthermore, since SuCNCW models encompass other normal mixture-based linear regression models with random regressors (Dang et al., 2017; Punzo and McNicholas, 2017; Diani et al., 2022), they represent a flexible approach for simultaneous robust clustering and detection of mildly atypical observations in linear regression analysis. Monte Carlo studies have shown that either the inclusion of irrelevant regressors in a cluster-weighted model or the presence of mildly atypical observations in the data can negatively affect the reconstruction of both the true classification and true parameter values, especially when the clusters of observations are not well-separated. Furthermore, they can have a negative impact on the choice of the order K of a CW model. The obtained results have demonstrated that such difficulties can be managed by resorting to SuCNCW models. In practical applications in which the regressors to be considered in the linear predictor of each response have to be determined from the data, an approach based on a joint use of SuCNCW models and techniques for variable selection (e.g., genetic algorithms, stepwise strategies) can also allow to identify the relevant predictors for each regression equation. A disadvantage of using an ECM algorithm to perform ML estimation of SuCNCW models is that it does not provide a direct assessment of the sample variability of the ML estimates. To this end, approaches commonly employed under finite mixture models could be employed (see, e.g., McLachlan and Peel, 2000). This aspect represents an avenue of future research.

4.6 Data availability

- The real-world data supporting the findings of this study reported in Section 4.4 are openly available in the `R` package `bayesm` (Rossi, 2012).

Bibliography

Aitken AC (1926) A series formula for the roots of algebraic and transcendental equations. Proc R Soc Edinb 45(1): 14–22

Aitkin M, Wilson TG (1980) Mixture models, outliers, and the EM algorithm. Technometrics 22(3): 325–331

Andrews JL, McNicholas PD (2011) Extending mixtures of multivariate t-factor analyzers. Stat Comput 21(3): 361–373

Baek J, McLachlan GJ (2011) Mixtures of common t-factor analyzers for clustering high-dimensional microarray data. Bioinformatics 27(9): 1269–1276

Bai X, Yao W, Boyer JE (2012) Robust fitting of mixture regression models. Comput Stat Data Anal 56(7): 2347–2359

Baird IG, Quastel N (2011) Dolphin-safe tuna from California to Thailand: localisms in environmental certification of global commodity networks. Ann Assoc Am Geogr 101(2): 337–355

Baudry JP, Raftery AE, Celeux G, Lo K, Gottardo R (2010) Combining mixture components for clustering. J Comput Graph Stat 19(2): 332–353

Browne RP, McNicholas PD (2014a) Estimating common principal components in high dimensions. Adv Data Anal Classif 8: 217–226

Browne RP, McNicholas PD (2014b) Orthogonal Stiefel manifold optimization for eigen-decomposed covariance parameter estimation in mixture models. Stat Comput 24: 203–210

Biernacki C, Celeux G, Govaert G (2000) Assessing a mixture model for clustering with the integrated completed likelihood. IEEE Trans Pattern Anal Mach Intell 22(7): 719–725

Biernacki C, Celeux G, Govaert G (2003) Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixture models. Comput Stat Data Anal 41(3–4): 561–575

Celeux G, Govaert G (1995) Gaussian parsimonious clustering models. Pattern Recognit 28(5): 781–793

Chevalier JA, Kashyap AK, Rossi PE (2003) Why don't prices rise during periods of peak demand? Evidence from scanner data. Am Econ Rev 93(1): 15–37

Dang UJ, Punzo A, McNicholas PD, Ingrassia S, Browne RP (2017) Multivariate response and parsimony for Gaussian cluster-weighted models. J Classif 34(1): 4–34

Dempster A, Laird N, Rubin D (1977) Maximum likelihood for incomplete data via the EM algorithm. J Roy Stat Soc: Ser B 39(1): 1–38

Diani C, Galimberti G, Soffritti G (2022) Multivariate cluster-weighted models based on seemingly unrelated linear regression. Comput Stat Data Anal 171: 107451

Frühwirth-Schnatter S (2006) Finite mixture and Markov switching models. Springer, New York

Gallaugher MPB, Tomarchio SD, McNicholas PD, Punzo A (2022) Multivariate cluster weighted models using skewed distributions. Adv Data Anal Classif 16: 93–124

Galimberti G, Scardovi E, Soffritti G (2016) Using mixtures in seemingly unrelated linear regression models with non-normal errors. Stat Comput 26(5): 1025–1038

Galimberti G, Soffritti G (2020) Seemingly unrelated clusterwise linear regression. Adv Data Anal Classif 14(2): 235–260

Gershenfeld N (1997) Nonlinear inference and cluster-weighted modeling. Ann. N. Y. Acad. Sci. 808: 18–24

Hastie, Tibshirani, Friedman (2009) The elements of statistical learning. Second edition. Springer, New York

Hennig C (2000) Identifiability of models for clusterwise linear regression. J Classif 17: 273–296

Henningsen A, Hamann JD (2007) `systemfit`: a package for estimating systems of simultaneous equations in `R`. J Stat Softw 23(4): 1–40

Hubert L, Arabie P (1985) Comparing partitions. J Classif 2(1): 193–218

Ingrassia S, Minotti SC, Vittadini G (2012) Local statistical modeling via a cluster-weighted approach with elliptical distributions. J Classif 29(3): 363–401

Ingrassia S, Minotti SC, Punzo A (2014) Model-based clustering via linear cluster-weighted models. Comput Stat Data Anal, 71: 159–182

Karlis D, Xekalaki E (2003) Choosing initial values for the EM algorithm for finite mixtures. Comput Stat Data Anal 41(3–4): 577–590

Magnus JR, Neudecker H (1988) Matrix differential calculus with applications in statistics and econometrics. Wiley, New York

Maronna RA, Martin RD, Yohai VJ (2006) Robust statistics: theory and methods. Wiley, Chichester

Mazza A, Punzo A (2020) Mixtures of multivariate contaminated normal regression models Stat Pap 61(2): 787–822

McLachlan GJ, Peel D (2000) Finite mixture models. Wiley, New York

McNicholas PD (2010) Model-based classification using latent Gaussian mixture models. J Stat Plan Inference 140(5): 1175–1181

Meng XL, Rubin DB (1993) Maximum likelihood estimation via the ECM algorithm: A general framework. Biometrika 80(2): 267–278

Park T (1993) Equivalence of maximum likelihood estimation and iterative two-stage estimation for seemingly unrelated regression models. Commun Stat Theory Methods 22(8): 2285–2296

Perrone G, Soffritti G (2023) Seemingly unrelated clusterwise linear regression for contaminated data. Stat Pap 64: 883–921. https://doi.org/10.1007/s00362-022-01344-6

Punzo A, McNicholas PD (2016) Parsimonious mixtures of multivariate contaminated normal distributions. Biometr J 58(6):1506–1537

Punzo A, McNicholas, PD (2017) Robust clustering in regression analysis via the contaminated Gaussian cluster-weighted model. J Classif 34(2): 249–293

Punzo A, Mazza A, McNicholas, PD (2018) `ContaminatedMixt`: an `R` package for fitting parsimonious mixtures of multivariate contaminated normal distributions. J Stat Softw 85(10): 1–25

R Core Team (2022) `R`: a language and environment for statistical computing. `R` Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org

Ritter G (2015) Robust cluster analysis and variable selection. Chapman and Hall, Boca Raton

Rossi PE (2012) `bayesm`: Bayesian inference for marketing/micro-econometrics. `R` package version 2.2-5. http://CRAN.R-project.org/package=bayesm

Rousseeuw PJ, Leroy AM (2005) Robust regression and outlier detection. Wiley, New York

Schwarz G (1978) Estimating the dimension of a model. Ann Stat 6(2): 461–464

Scrucca L, Fop M, Murphy TB, Raftery AE (2017) `mclust 5`: clustering, classification and density estimation using Gaussian finite mixture models. R J 8(1): 205–223

Srivastava VK, Giles DEA (1987) Seemingly unrelated regression equations models. Marcel Dekker, New York

Tukey JW (1960) A survey of sampling from contaminated distributions. In: Olkin I (ed) Contributions to probability and statistics: essays in honor of Harold Hotelling, Stanford studies in mathematics and statistics. Stanford University Press, California, pp 448–485

Yao W, Wei Y, Yu C (2014) Robust mixture regression using the t-distribution. Comput Stat Data Anal 71: 116–127

Chapter 5

A study on housing tension in the municipalities of the Emilia-Romagna region[1]

[1]The results of this chapter will be summarized in a scientific paper to be submitted for publication.

Abstract

Housing deprivation in Italy is a complex phenomenon that has been widely studied and discussed by several researchers in the last years. The general economic situation has worsened and with it the housing instability of low-income households have grown dramatically. Housing policies in support of marginalised groups of families have been also relatively weak. In such context, strategies and social housing programs appear to be urgent. As far as the case of Emilia-Romagna region is concerned, since 2001 a regional observatory of the housing system has been established in order to monitor housing conditions. Thanks to an implementation agreement between the region and the Department of Statistical Science of the University of Bologna, a study of housing deprivation in the municipalities of the Emilia-Romagna region has been carried out in order to provide a cognitive support for developing strategies and planning tools or for implementing actions that are oriented to address the housing issue. This chapter summarises the main results of this study. In particular, the dependence of housing tension in the municipalities of the Emilia-Romagna region on several indicators coming from a dataset created by the region has been evaluated through Mixtures of Contaminated Seemingly Gaussian regressions (MCSG) models and other clusterwise linear regression models. Furthermore, a new class of MCSG models is introduced here so as to allow the mixing weights to be expressed as a function of some concomitant variables. To select the relevant indicators to be employed for the explanation of the housing tension, a genetic algorithm and a backward elimination technique have been exploited.

Keywords: Emilia-Romagna region, Genetic algorithm, Housing tension, Mild outlier, Mixture of regression models, Model-based cluster analysis, Seemingly unrelated regression

5.1 Introduction

The last decades have been characterized by profound environmental changes, economic crises and an intensification of migration flows. All these factors, together with the recent Covid-19 pandemic, have contributed to worsening the level of social and spatial fragmentation of the cities. Consequently, also the living conditions of the social groups with lower incomes have deteriorated, contributing to increase both the population living below the poverty line and the differences between income groups. As far as the field of housing is concerned, the decline in social housing policies together with the aggravating residential segregation have contributed to make up access to housing increasingly difficult. For these reasons, many researchers have focused their attention on the study of public housing policies in order to understand contemporary housing dynamics and to identify the main determining characteristics of the housing deprivation phenomenon. This is also the case of Italy. On the one hand, the housing deprivation is still today a problem that concerns all the regions, provinces and municipalities. On the other hand, urban policies result to be insufficient compared to the housing needs. To cope with this situation, it results to be necessary to planning and developing strategies able to find solutions for the housing issue. The focus of this chapter is the Emilia-Romagna region and, in particular, the study on housing tension in its municipalities. Since 2001, this region has established the Regional Observatory of the housing system (ORSA) with the regional law n. 24 of 8 August 2001, which aims to evaluate data on housing conditions, allowing for a better assessment of both household conditions and the effectiveness of housing policies. Furthermore, the challenge of ORSA is to analyse housing needs, monitor and evaluate interventions and programs in the housing sector and to support the elaboration of housing policies, in particular, and of welfare, in general. To better understand the phenomenon, the sharing of the data to external public and private entities that are able to contribute to the activity of the Regional Observatory is also appreciated. In thix context, research activities have been carried out within an implementation agreement between the Emilia-Romagna region and the Department of Statistical Sciences of the University of Bologna. Based on data provided by the region, Mixtures of Seemingly Unrelated Contaminated Normal Regression Model (MCSG) (Perrone and Soffritti, 2023) and mixtures of seemingly unrelated Gaussian (MSG) regressions models (Galimberti and Soffritti, 2020) have been employed to study the effects of certain factors pertaining to three different areas (social demography, social life/income conditions, housing supply and market) on housing tension in

328 municipalities of the Emilia-Romagna region, by simultaneously allowing for the detection of latent clusters of municipalities induced by some source of unobserved heterogeneity. In order to ensure additional flexibility, new mixtures of seemingly unrelated contaminated normal linear regressions models with concomitant variables (cMCSG) and mixtures of seemingly unrelated Gaussian regressions models with concomitant variables (cMSG) are introduced here. With these new models, prior probabilities of belonging to latent clusters are assumed to depend on some concomitant covariates. In order to select the relevant predictors of housing tension in MCSG, MSG, cMCSG and cMSG models, a genetic algorithm and a backward elimination technique have been employed.

The chapter is organised as follows. Section 5.2 provides a summary of housing deprivation in Italy. A general introduction of the administrative situation in the Emilia-Romagna region is reported in Section 5.3. Section 5.4 provides details on the indicators and the variables of the analysed dataset. The specific aims of the study are reported in Section 5.5. A summary of the methods employed in the analyses is given in Section 5.6, together with the introduction of the cMCSG and cMSG model classes. The main results are presented in Section 5.7. Finally, in Section 5.8, some conclusions and remarks are discussed.

5.2 Housing deprivation in Italy

In the last two decades, the phenomenon of housing deprivation in Italy has been widely studied and discussed by several researchers. Housing problems, in fact, have increased dramatically in recents years due to several transformations of contemporary society, migrant flows, economic crises and, finally, Covid-19 pandemic. Furthermore, until the 1990s, the local jurisdictions have marginalized housing problems, contributing to increase the mismatch between the income groups. According to some researchers, the incorrect assessment of the severity of such problems has been due to the common thought that housing tension concerns only the most disadvantaged population bracket (renter households, foreigners, elderly people) and not also the middle class (Bonafede, 2021). In general, housing problems have led to the overcoming of the family in the traditional sense and the birth of new forms of family organization (single-parent families, single people, couples without children, etc.). The housing tension phenomenon has been usually focused on three main dimensions (Townsend, 1979): housing inadequacy (structural deficiency or a lack of housing facilities, Kutty (1999)), overcrowding (insufficient space in relation to the

number of users, Gray and Campbell (2001)) and unaffordability (the pressure on households because housing costs (rent or mortgage) take up too large a proportion of the household income, Hancock (1993)). The distribution of housing deprivation on the Italian territory, instead, appears to be more pronounced in municipalities that are densely populated; thus, housing deprivation seems to be more widespread in the North-West and in the South of Italy. For this reason, in order to provide a cognitive support for developing strategies and planning tools for the municipalities with a critical situation in terms of housing problems, some classifications of municipalities have been specified. In particular, since 2003, the Inter-ministerial Committee for Economic Programming (Comitato Interministeriale per la Programmazione Economica, CIPE) has identified some municipalities as having High Housing Tension[2] (Alta Tensione Abitativa, ATA) based on demographic growth thresholds in order to stipulate special agreements. However, this criterion has appeared to be inadequate because it does not take into account the territorial changes of the last years. For this reason, in 2016, the Italian Conference of Regions and Autonomous Provinces proposed a revision of the thresholds for identifying ATA municipalities in order to integrate the previous criterion with an indicator of the housing problem[3]. On the other hand, ISTAT (National Institute of Statistics, Istituto Nazionale di Statistica) has introduced another classification of the Italian municipalities which defines some municipalities as having High Housing Density (Alta Densità Abitativa, ADA) depending on their demographic size. In particular, a municipality is defined as ADA if it has a number of inhabitants greater than 10,000. Furthermore, municipalities can also be classified according to their geographic location. In particular, based on the administrative reform evolved in 1999 with the national law 265/99, it is also possible to identify mountain municipalities. This definition is primarily intended for local public administrations located in mountainous or partially mountainous area. Since 2014, a new strategy (National Strategy for Inner Areas, NSIA) has been established for every region of Italy in order to contribute to their economic and social recovery by creating jobs, fostering social inclusion and cutting the costs of regional depopulation. Specifically, the strategy approved for 2014-2020 defines the "inner areas" as areas at some considerable distance (in terms of time) from a selected municipality to the nearest hubs (Poles - main centres) which provide essential services (education, health and mobility) (Ministry of Economic Development, 2014). The wider this distance, the greater its periphery. In particular, the inner areas plan classified municipalities into Intermediate (I), Peripheral (P) and Ultraperipheral (UP) areas.

[2] https://www.mit.gov.it/normativa/decreto-ministeriale-del-13112003

[3] see Regioni.it 2884 of 18 February 2016

Table 5.1: Joint frequency distribution of the municipalities classified as ADA and ATA in the ERR.

	ATA		
ADA	Yes	No	Tot.
Yes	38	63	101
No	1	228	229
Tot.	39	291	330

Specifically, intermediate areas are those whose distance is between 20 and 40 min., peripheral areas are between 40 and 75 min. and ultra-peripheral areas are greater than 75 min. Moreover, this strategy also introduces an Urban Belts (UB) classification for the municipalities far less than 20 min. from the nearest Pole (Barca et al., 2014; Moretto et al., 2021).

In this context, the Italian Public Residential Building (Edilizia Residenziale Pubblica - ERP) endowment aims to help the economically weakest social groups, by encompassing various types of building interventions. However, the latter policy is completely insufficient, safeguarding only some parts of the population. Furthermore, in addition to ERP endowment, a national fund was established in 1998 to support access to rental housing. Although both policies are capable to limit the housing tension problems, their effectiveness is not fully satisfactory. For these reasons, some Italian regions have invested in strategies and social housing programs in order to improve the housing condition of the low-income households. This is also the case of the Emilia-Romagna region, where for several years the Regional Observatory of the housing system (Osservatorio regionale del sistema abitativo - ORSA[4], art. 16 of Regional Law 24/2001) has been established in order to continuously ascertain housing needs, support the development of housing policies, monitor their effectiveness and, more generally, acquire, collect, process, disseminate and evaluate data on housing conditions and activities in the building sector. In particular, the Observatory has the task of collecting and processing information regarding: local information flows on housing needs, public intervention in the housing sector, the cyclical and structural surveys on housing scenarios, the verification and monitoring of the implementation of the programmes and the methods of using the existing building stock. Housing problems are often nothing more than a lack of disposable income compared to the needs of the family or a lack of accommodation. In all these circumstances, it is important for the Emilia-Romagna regional government to intervene with suitable public housing policies.

[4]https://territorio.regione.emilia-romagna.it/osservatorio-delle-politiche-abitative/fabbisogno-abitativo

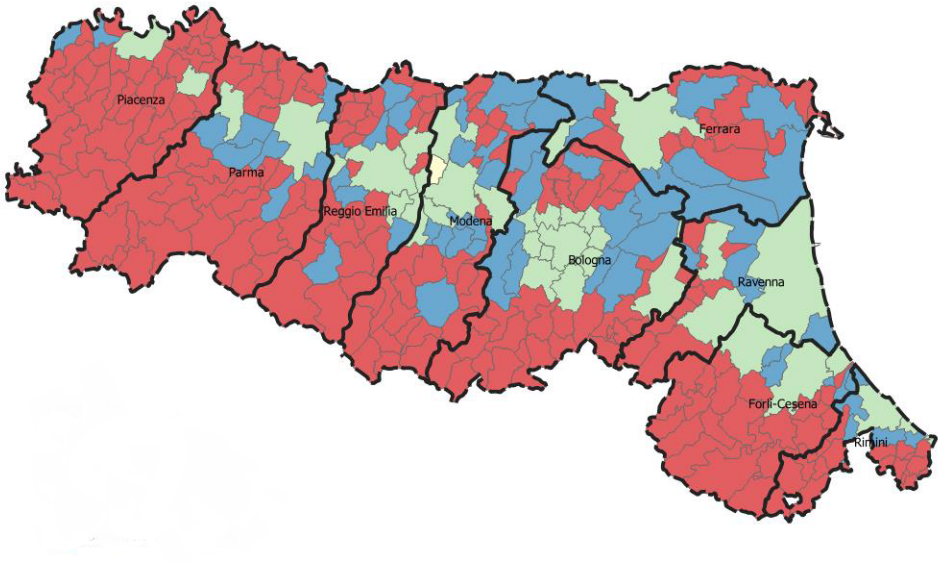

Figure 5.1: Map of the ATA and ADA municipalities in the ERR.

5.3 The Emilia-Romagna Region (ERR)

For administrative purposes, the 20 Italian regions are divided into provinces and municipalities. As far as the Emilia-Romagna Region (in the north-east of Italy) is concerned, it covers an area of 22,510 square kilometers (sq km) for a population of 4.459 million distributed among nine provinces[5]. In particular, these latter provinces of the ERR are (in brackets the province acronym and the number of municipalities for each province): Bologna (BO, 55), Forlì-Cesena (FC, 30), Ferrara (FE, 21), Modena (MO, 47), Piacenza (PC, 46), Parma (PR, 44), Ravenna (RA, 18), Reggio Emilia (RE, 42) and Rimini (RN, 25). Thus, the total current number of the ERR municipalities is 330. As far as the housing tension in the ERR is concerned, it appears to be characterized by heterogeneity among the municipalities of the provinces. Thus, based on the above mentioned classifications, also local public administrations of the ERR have been classified as having High Housing Tension and High Housing Density. Figure 5.1 shows the map of the municipalities based on ADA and ATA classifications. In particular, municipalities that have been classified both as ADA and ATA have been colored in green, the ADA municipalities are in blue, municipalities that are neither ADA nor ATA in red and, finally, in yellow the unique municipality characterized both by a number of inhabitants lower than 10,000 and an High Housing Tension. Table 5.1 shows the joint frequency distribution of the municipalities classified as ADA and ATA. The majority of the ERR municipalities are classified as not ADA nor ATA, while 11.5% have both high housing tension and high housing density. Classification

[5]Regione Emilia-Romagna Statistica. Available online: https://statistica.regione.emilia-romagna.it/)

details of the ATA, ADA and mountain municipalities in the ERR by province have been reported in Table 5.2. From the latter table, it emerges that the municipalities classified as ATA are also classified as ADA municipalities. This is true for each province except Modena, where the municipality of Campogalliano is characterized by High Housing Tension but is a municipality with 8605 inhabitants (2020) (see the municipality in yellow in Figure 5.1). Table 5.3 shows the list of municipalities for each province classified according to the NSIA criterion. Finally, Table 5.4 shows details about the number of municipalities classified as ADA, ATA, Mountains and in one of the five categories of NSIA for the nine Emilia-Romagna provinces.

5.4 Dataset

The dataset has been created by the ERR region and is composed by data from different sources. It refers to the municipalities of the ERR and provides information about several quantitative variables defined in 2020 by the ERR in collaboration with ART-ER[6], within the document "Regional Observatory of housing system and related activities"[7] (Osservatorio Regionale del Sistema Abitativo e attività connesse). From 17 June 2021, two municipalities from the Marche region, Montecopiolo and Sassofeltrio, have been aggregated to the ERR[8]. Thus, since the information assets contained in the dataset cover the period from 2016 to 2020, only the other J=328 municipalities have been considered in this research. As far as the variables are concerned, they have been divided into three macro-areas (named also pillars or fields). Then, a multitude of aspects have been taken into consideration within each pillar. In particular, six Socio - Demographic (SD) indicators, five Social Life and Income Condition (SLIC) indicators and nine Housing Supply and Housing Market conditions indicators (HSHM) have been considered (Table 5.5). They have been chosen because they are closely related with housing tension and, therefore, could play an important role in its explanation. Moreover, some of these indicators may be expected to have a direct (+) or inverse (−) effect on housing tension; in fact, as an indicator increases, the housing tension can be expected also to increase (direct effect) or decrease (inverse effect) (see the first column of Table 5.5 - the absence of the "+" and "−" signs means that there is no any expectation for the effect on housing tension). Furthermore,

[6]Attractiveness Research Territory (ART) is the Emilia-Romagna (ER) Joint Stock Consortium born with the purpose of fostering the region's sustainable growth by developing innovation and knowledge, attractiveness and internationalisation of the territory. https://www.art-er.it/

[7]https://territorio.regione.emilia-romagna.it/osservatorio-delle-politiche-abitative/misure-di-sostegno-alle-famiglie/politiche-erp-regionali-e-locali/orsa_verso_un_sistema_informativo_politiche_abitative_2020.pdf

[8]Law No. 84 of 28 May 2021 published in the Official Gazette n.142 of 16/06/2021

Table 5.2: List of ATA, ADA and mountains municipalities in the ERR.

Provinces	ATA	ADA	Mountains
Bologna	Anzola dell'Emilia - Bologna - Calderara di Reno Casalecchio di Reno - Castel Maggiore - Castenaso Granarolo dell'Emilia - Imola - Pianoro San Lazzaro di Savena - Sasso Marconi - Zola Predosa	Anzola dell'Emilia - Bologna - Budrio Calderara di Reno - Casalecchio di Reno - Castel Maggiore Castel San Pietro Terme - Castenaso - Crevalcore Granarolo dell'Emilia - Imola - Medicina Molinella - Monte San Pietro - Ozzano dell'Emilia Pianoro - San Giovanni in Persiceto - San Lazzaro di Savena San Pietro in Casale - Sasso Marconi - Valsamoggia Zola Pedrosa	Alto Reno Terme - Borgo Tossignano - Camugnano Casalfiumanese - Castel d'Aiano - Castel del Rio Castel di Casio - Castiglione dei Pepoli - Fontanelice Gaggio Montano - Grizzana Morandi - Lizzano in Belvedere Loiano - Marzabotto - Monghidoro Monte San Pietro - Monterenzio - Monzuno Pianoro - San Benedetto Val di Sambro - Sasso Marconi Valsamoggia - Vergato
Forlì-Cesena	Cesena - Cesenatico - Forlì	Bertinoro - Cesena - Cesenatico Forlì - Forlimpopoli - Gambettola San Mauro Pascoli - Savignano sul Rubicone	Bagno di Romagna - Borghi - Civitella di Romagna Dovadola - Galeata - Meldola Mercato Saraceno - Portico e San Benedetto - Predappio Premilcuore - Rocca San Casciano - Roncofreddo Santa Sofia - Sarsina - Sogliano al Rubicone Tredozio - Verghereto
Ferrara	Cento - Ferrara	Argenta - Bondeno - Cento - Codigoro Comacchio - Copparo - Ferrara Portomaggiore - Terre del Reno	
Modena	Campogalliano - Carpi Castelfranco Emilia - Formigine Modena - Sassuolo	Bomporto - Carpi - Castelfranco Emilia Castelnuovo Rangone - Castelvetro di Modena Finale Emilia - Fiorano Modenese - Formigine Maranello - Mirandola - Modena - Nonantola Novi di Modena - Pavullo nel Frignano San Felice sul Panaro - Sassuolo - Soliera Spilamberto - Vignola	Fanano - Fiumalbo - Frassinoro Guiglia - Lama Mocogno - Marano sul Panaro Montecreto - Montefiorino - Montese Palagano - Pavullo nel Frignano - Pievepelago Polinago - Prignano sulla Secchia - Riolunato Serramazzoni - Sestola - Zocca
Piacenza	Fiorenzuola d'Arda – Piacenza	Castel San Giovanni - Fiorenzuola d'Arda Piacenza - Rottofreno	Alta Val Tidone - Bettola - Bobbio Cerignale - Coli - Corte Brugnatella Farini - Ferriere - Gropparello Morfasso - Ottone - Piozzano Travo - Vernasca - Zerba
Parma	Fidenza – Parma	Collecchio - Fidenza - Langhirano Medesano - Montechiarugolo - Noceto Parma - Salsomaggiore Terme Sorbolo Mezzani	Albareto - Bardi - Bedonia Berceto - Bore - Borgo Val di Taro Calestano
Ravenna	Faenza - Lugo – Ravenna	Alfonsine - Bagnacavallo - Cervia Faenza - Lugo - Massa Lombarda Ravenna - Russi	Brisighella - Casola Valsenio - Riolo Terme
Reggio Emilia	Casalgrande - Correggio Montecchio Emilia - Reggio nell'Emilia Rubiera - Scandiano	Bibbiano - Cadelbosco di Sopra - Casalgrande Castellarano - Castelnovo ne' Monti - Correggio Guastalla - Montecchio Emilia - Novellara Quattro Castella - Reggio nell'Emilia - Rubiera Sant'Ilario d'Enza - Scandiano	Baiso - Canossa - Carpineti Casina - Castelnovo ne' Monti - Toano Ventasso - Vetto - Viano - Villa Minozzo
Rimini	Cattolica - Riccione – Rimini	Bellaria Igea Marina - Cattolica - Coriano Misano Adriatico - Riccione - Rimini Santarcangelo di Romagna - Verucchio	Casteldelci - Maiolo - Novafeltria Pennabilli - Poggio Torriana - San Leo Sant'Agata Feltria - Talamello - Verucchio

Table 5.3: List of municipalities in the ERR based on the five NSIA classification (Urban Belts (UB), Intermediate (I), Poles (PO), Peripheral (P) and Ultra-Peripheral (UP)).

Provinces	UB	I	PO	P	UP
Bologna	Anzola dell'Emilia - Argelato - Borgo Tossignano Calderara di Reno - Casalecchio di Reno Casalfiumanese - Castel Guelfo di Bologna Castel Maggiore - Castel San Pietro Terme Castenaso - Dozza - Fontanelice Granarolo dell'Emilia - Malalbergo - Medicina Monte San Pietro - Mordano San Lazzaro di Savena - Sasso Marconi Zola Pedrosa	Baricella - Bentivoglio - Budrio Castel del Rio - Castello d'Argile - Crevalcore Galliera - Marzabotto - Minerbio Molinella - Monzuno - Ozzano dell'Emilia Pianoro - Pieve di Cento - Sala Bolognese San Giorgio di Piano - San Giovanni in Persiceto San Pietro in Casale - Sant'Agata Bolognese Valsamoggia	Bologna - Imola	Alto Reno Terme - Camugnano Castel di Casio - Castiglione dei Pepoli Gaggio Montano - Grizzana Morandi Lizzano in Belvedere - Loiano Monghidoro - Monterenzio San Benedetto Val di Sambro Vergato	Castel d'Aiano
Forlì-Cesena	Bertinoro - Castrocaro Terme e Terra del Sole Cesenatico - Dovadola - Forlimpopoli Gambettola - Gatteo - Longiano Meldola - Mercato Saraceno - Modigliana Montiano - Predappio - Roncofreddo San Mauro Pascoli - Sarsina Savignano sul Rubicone	Bagno di Romagna - Borghi Civitella di Romagna - Galeata Portico e San Benedetto Rocca San Casciano Sogliano al Rubicone Tredozio	Cesena - Forlì	Premilcuore - Santa Sofia Verghereto	
Ferrara	Argenta - Masi Torello Poggio Renatico - Vigarano Mainarda Voghiera	Bondeno - Comacchio - Copparo Fiscaglia - Jolanda di Savoia Ostellato - Portomaggiore Riva del Po - Terre del Reno Tresignana	Ferrara	Cento - Codigoro Goro - Lagosanto Mesola	
Modena	Bastiglia - Bomporto - Campogalliano Castelnuovo Rangone - Cavezzo Concordia sulla Secchia Fiorano Modenese - Formigine Maranello - Nonantola Novi di Modena - San Possidonio San Prospero - Sassuolo - Soliera	Camposanto - Castelfranco Emilia Castelvetro di Modena - Finale Emilia Medolla - Mirandola Ravarino - San Cesario sul Panaro San Felice sul Panaro Savignano sul Panaro Spilamberto - Vignola	Carpi - Modena	Guiglia - Marano sul Panaro Montefiorino Pavullo nel Frignano Polinago Prignano sulla Secchia Serramazzoni - Zocca	Fanano - Fiumalbo Frassinoro - Lama Mocogno Montecreto - Montese - Palagano Pievepelago - Riolunato - Sestola
Piacenza	Alseno - Besenzone - Borgonovo Val Tidone Cadeo - Calendasco - Caorso Carpaneto Piacentino - Cortemaggiore Castel San Giovanni - Castell'Arquato - Castelvetro Piacentino Fiorenzuola d'Arda - Gazzola - Gossolengo Gragnano Trebbiense - Lugagnano Val D'Arda Monticelli d'Ongina - Podenzano - Pontenure Rivergaro - Rottofreno - San Giorgio Piacentino San Pietro in Cerro - Sarmato - Vernasca Vigolzone - Villanova sull'Arda	Agazzano - Bettola Gropparello Pianello Val Tidone - Piozzano Ponte dell'Olio - Travo Ziano Piacentino	Piacenza	Alta Val Tidone - Bobbio Coli - Corte Brugnatella Farini - Morfasso	Cerignale Ferriere Ottone Zerba
Parma	Busseto - Collecchio - Colorno Felino - Fontanellato -Fontevivo Fornovo di Taro - Medesano Montechiarugolo - Noceto Polesine Zibello - Roccabianca Sala Baganza - Salsomaggiore Terme San Secondo Parmense Sissa Trecasali - Soragna Sorbolo Mezzani - Torrile	Calestano - Langhirano Lesignano de' Bagni Pellegrino Parmense - Solignano Traversetolo Varano de' Melegari	Fidenza - Parma	Albareto - Bardi - Bedonia Berceto - Bore Borgo Val di Taro - Compiano Corniglio - Neviano degli Arduini Terenzo - Tizzano Val Parma Tornolo - Valmozzola Varsi	Monchio delle Corti Palanzano
Ravenna	Alfonsine - Bagnacavallo - Bagnara di Romagna Brisighella - Castel Bolognese - Cervia Conselice - Cotignola - Fusignano Massa Lombarda - Riolo Terme - Russi Sant'Agata sul Santerno - Solarolo	Casola Valsenio	Faenza - Lugo Ravenna		
Reggio Emilia	Albinea - Bagnolo in Piano - Bibbiano Brescello - Cadelbosco di Sopra Campagnola Emilia - Campegine Castelnovo di Sotto - Cavriago Correggio - Fabbrico - Gattatico Montecchio Emilia - Novellara Poviglio - Reggiolo - Rio Saliceto Rolo - Rubiera - San Martino in Rio Sant'Ilario d'Enza - Vezzano sul Crostolo	Boretto - Canossa Casalgrande - Castellarano Gualtieri - Guastalla Luzzara - Quattro Castella San Polo d'Enza - Scandiano Viano	Reggio nell'Emilia	Baiso - Carpineti Casina Castelnovo ne' Monti Toano - Vetto Villa Minozzo	Ventasso
Rimini	Bellaria Igea Marina - Cattolica Coriano - Gemmano - Misano Adriatico Montescudo Monte Colombo - Morciano di Romagna Saludecio - San Clemente - San Giovanni in Marignano Santarcangelo di Romagna - Verucchio	Mondaino - Montefiore Conca Montegridolfo - Poggio Torriana Sant'Agata Feltria	Riccione - Rimini	Casteldelci - Maiolo Novafeltria -Pennabilli San Leo - Talamello	

Table 5.4: Number of municipalities classified as ADA, ATA, Mountains and in one of the five NSIA classification for the Emilia-Romagna provinces.

Provinces	ATA		ADA		Mountains		NSIA				
	Yes	No	Yes	No	Yes	No	UB	I	PO	P	UP
Bologna	12	43	22	33	23	32	20	20	2	12	1
Forlì-Cesena	3	27	8	22	17	13	17	8	2	3	0
Ferrara	2	19	9	12	0	21	5	10	1	5	0
Modena	6	41	19	28	18	29	15	12	2	8	10
Piacenza	2	44	4	42	15	31	27	8	1	6	4
Parma	2	42	9	35	7	37	19	7	2	14	2
Ravenna	3	15	8	10	3	15	14	1	3	0	0
Reggio Emilia	6	36	14	28	10	32	22	11	1	7	1
Rimini	3	24	8	19	11	16	12	6	2	7	0
Tot.	39	291	101	229	104	226	151	83	16	62	18

two variables representative of housing tension have been considered: the proportion of the low income households and the proportion of households demanding for public residential housing (Edilizia Residenziale Pubblica, ERP). Thus, a total number of P=22 different indicators have been considered. Dataset also contains two variables that are the Total Resident Population (TRP) and Total Resident Foreigners (TRF) in 2020. Details about the overall indicators are presented in the following subsections.

5.4.1 Socio - Demographic (SD) indicators

The first set of indicators (SD1 - SD6, top side of the second column of Table 5.5) concerns the demographic situation and demographic dynamics of the municipalities. In particular, the SD1 indicator (Population density) represents the number of inhabitants (inh) per square kilometer (sq km) in 2020; thus, for each municipality, the first indicator has been computed as follows:

$$\text{SD1} = \frac{\text{Resident population in 2020 (inh)}}{\text{Municipal area (sq km)}}.$$

Consequently, higher densities of people are expected to be associated with higher housing tension (see the first column of Table 5.5). As far as the SD2, SD3 and SD4 indicators are concerned, they represent changes in resident population, changes in resident households and

Table 5.5: Indicators, meanings, and sources (the "+" or the "-" signs reported in the first column means a direct or inverse expected impact of an indicator on housing tension).

	Indicator	**Description and Reference Unit (Reference years)**	**Source**
	Socio-Demographic (SD)		
+	SD1–Population density	Number of inhabitants per square kilometer (2020)	ISTAT/ERR
+	SD2–Change in population	Change in resident population (2017–2020)	ISTAT/ERR
+	SD3–Change in household	Change in resident households (2017–2020)	ISTAT/ERR
+	SD4–Change in foreigners	Change in resident foreigners (2017–2020)	ISTAT/ERR
+	SD5–Household Size	Annual average of the Average Households Size (2017–2020)	ISTAT/ERR
+	SD6–Migration balance	Annual average of the Migration Balance (2017–2020)	ISTAT/ERR
	Social Life and Income Condition indicators (SLIC)		
+	SLIC1–Education	Proportion of people with medium-high level education (2018–2020)	ISTAT
+	SLIC2–Employment	Proportion of people in employment (2018-2019)	ISTAT
	SLIC3–Taxable Income	Change in Taxable Income per taxpayers (2016-2020)	MEF
+	SLIC4–Low Income Taxpayers	Proportion of low-income taxpayers (2020)	MEF
+	SLIC5–Gini Index	Income inequality index (2020)	MEF
	Housing Supply and Housing market indicators (HSHM)		
−	HSHM1–Housing Stock	Proportion of Housing Units per household (2020)	OMI - Revenue Agency
−	HSHM2–Change in Housing Stock	Change in Housing Stock (2016-2020)	OMI - Revenue Agency
+	HSHM3–Ratio of the rent to income	Ratio of the Maximum Monthly Rent to family income (2020)	OMI - Revenue Agency
+	HSHM4–Change rents	Change in Maximum Monthly Rent (2018-2020)	OMI - Revenue Agency
+	HSHM5–Family Income	Number of annual income years necessary to purchase a house (2020)	OMI - Revenue Agency
+	HSHM6–Dwelling prices	Change in civil Dwelling prices (2017-2020)	OMI - Revenue Agency
+	HSHM7–NNT	Change in averages of the total Number of Normalized Transactions (NNT) (2018-2020/2015-2017)	OMI - Revenue Agency
+	HSHM8–IMI	Housing Stock dynamics index (2016-2020)	OMI - Revenue Agency
+	HSHM9–Property purchase	Change in averages of the property purchase (2018-2020/2017-2019)	OMI - Revenue Agency

changes in resident foreigners between 2017 and 2020 respectively. In particular,

$$\begin{aligned}
\text{SD2} &= \frac{\text{Resident population in 2020 (inh)}}{\text{Resident population in 2017 (inh)}} - 1; \\
\text{SD3} &= \frac{\text{Resident households in 2020 (inh)}}{\text{Resident households in 2017 (inh)}} - 1; \\
\text{SD4} &= \frac{\text{Resident foreigners in 2020 (inh)}}{\text{Resident households in 2017 (inh)}} - 1.
\end{aligned}$$

Thus, positive changes of these indicators are assumed to have a direct impact on housing tension (see the first column of Table 5.5) due to overcrowding issues. Furthermore, as far as the SD4 indicator is concerned, foregneirs face additional problems due to long standing reasons of racial and housing market discrimination (Bogdon and Can, 1997). The last SD indicators have been constructed as follows:

$$\begin{aligned}
\text{SD5} &= \frac{1}{4} \sum_{t=2017}^{2020} \text{AHS}_t, \\
\text{SD6} &= \frac{1}{4} \sum_{t=2017}^{2020} \text{MB}_t,
\end{aligned}$$

where AHS_t and MB_t are the Average Household Size and the Migration Balance (the difference between the number of immigrants and the number of emigrants) at time t, respectively. Thus, as far as the SD5 indicator (Household Size, see the second column of Table 5.5) is considered, an increase in family composition (e.g., presence of children) could have a direct effect on housing tension due to economic and overcrowding issues. The same holds for the SD6 indicator. Table 5.6 reports some descriptive statistics of the indicators employed in the analysis. For the SD indicators, it is worth noting that the Zerba municipality (PC) has the lowest population density (2.9 inh/sq km - SD1), the highest decrease in resident households (-8.6 % - SD3) and the lowest average value for the Household Size indicator (1.3 - SD5). Bologna (BO), instead, is the municipality with the highest value for the first and the last SD indicators (Table 5.6). As far as the SD2 indicator is concerned, the highest decrease and highest increase in resident population correspond respectively to the Farini (PC) and Granarolo dell'Emilia (BO) municipalities; Monchio delle Corti (PR) and Casteldelici (RN), instead, are the ones with the highest decrease (-34.0 %) and highest increase (53.8 %) in resident foreigners (SD4). Luzzara and Reggiolo, two municipalities of the Reggio Emilia province, correspond respectively at the minimum (-88.500) and the maximum (2.633) values of the SD6 and SD2 indicators. Finally, the

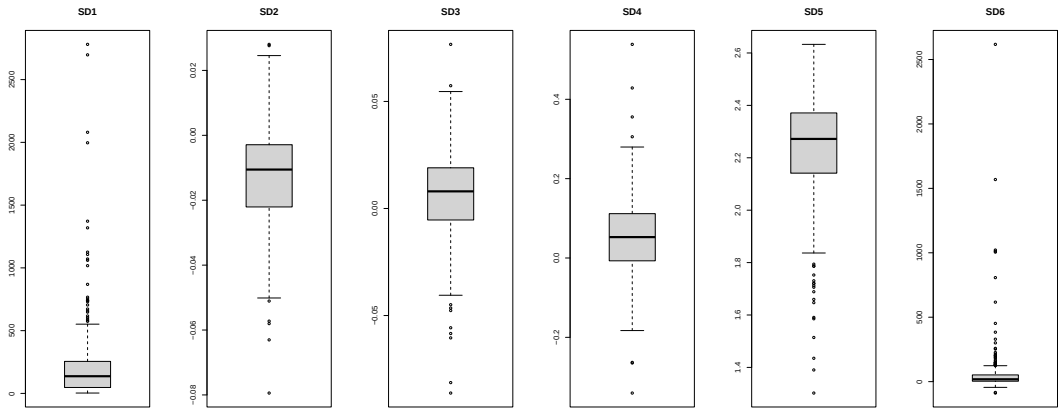

Figure 5.2: Boxplot of the SD indicators

municipality of Borghi (FC) has the maximum positive change in resident household (7.7 % - SD3) (Table 5.6). For the construction of the overall indicators, ISTAT and ERR sources have been taken into consideration. To better describe the overall descriptive statistics, Figure 5.2 shows the box-plots for this first strand of indicators. From the latter figure it emerges that each SD indicator is characterized by the presence of outliers. In particular, some extreme outliers can be easily identified in the first and the sixth socio-demographic indicators.

5.4.2 Social Life and Income Condition (SLIC) indicators

Among the five indicators of the second pillar (SLIC1-SLIC5, half side of Table 5.5), SLIC1 and SLIC2 concern aspects of the Social Life while the other three indicators (SLIC3-SLIC5) include informations about the Households Income Condition. In particular, for each municipality, the Education indicator (SLIC1) and the Employment indicator (SLIC2) have been computed as follows:

$$\text{SLIC1} = \frac{\sum_{t=2018}^{2020} \text{PALM}_t}{\sum_{t=2018}^{2020} \text{TRP}_t},$$
$$\text{SLIC2} = \frac{\sum_{t=2018}^{2020} \text{WAPE}_t}{\sum_{t=2018}^{2020} \text{TRP}_t},$$

where the acronyms PALM_t, WAPE_t and TRP_t represent the number of People with At Least a Middle school diploma, the number of Working Age Population that is in Employment and the Total Resident Population at time t, respectively. Thus, SLIC1 and SLIC2 indicators describe the proportion of people with medium-high level education and the proportion of people in employment in a municipality, respectively (Table 5.5). For this reason, the higher the SLIC1

Table 5.6: Descriptive statistics of the quantitative variables

Indicator	Minimum	First quartile	Median	Average	Third quartile	Maximum
SD1	2.901	47.253	136.899	223.766	252.555	2780.676
SD2	−0.079	−0.022	−0.010	−0.013	−0.003	0.028
SD3	−0.086	−0.005	0.008	0.007	0.019	0.077
SD4	−0.340	−0.006	0.053	0.049	0.112	0.538
SD5	1.302	2.142	2.272	2.228	2.371	2.633
SD6	−88.500	3.750	18.380	62.880	52.000	2617.500
SLIC1	0.641	0.761	0.780	0.774	0.794	0.844
SLIC2	0.273	0.483	0.512	0.505	0.537	0.591
SLIC3	−0.487	0.023	0.038	0.034	0.052	0.259
SLIC4	0.263	0.321	0.352	0.368	0.408	0.729
SLIC5	0.105	0.161	0.176	0.178	0.196	0.297
HSHM1	1.039	1.150	1.248	1.617	1.654	10.755
HSHM2	−0.024	0.002	0.006	0.006	0.011	0.038
HSHM3	0.022	0.140	0.178	0.186	0.220	0.641
HSHM4	−0.130	0.111	0.159	0.197	0.250	1.210
HSHM5	1.645	3.285	3.759	4.132	4.570	22.452
HSHM6	−0.210	0.171	0.249	0.245	0.329	1.076
HSHM7	−0.331	0.154	0.278	0.314	0.405	2.662
HSHM8	0.006	0.012	0.016	0.016	0.021	0.033
HSHM9	−0.588	0.003	0.151	0.271	0.422	3.097
ISEE.20	0.000	0.042	0.061	0.058	0.070	0.200
RANK.21	0.000	0.010	0.017	0.018	0.025	0.103

indicator is, the higher the housing tension will be. In fact, a high level of education generally implies a high income and, consequently, a low difficulty in accessing the housing market. The minimum (64.1 %) and the maximum (84.4 %) values the SLIC1 indicator are reached by Farini (PC) and Bologna (BO) municipalities (Table 5.6). As far as the SLIC2 indicator is concerned, the housing tension is expected to increase if the number of people in employment increases (see the first column of Table 5.5). Two municipalities of the Piacenza province correspond to the lowest (Cerignala, PC) and the highest (Gossolengo, PC) proportion for the SLIC2 indicator. ISTAT sources have been considered for the SLIC1 and SLIC2 data (see the fourth column of Table 5.5).

The remaining SLIC indicators concern the Households Income Condition area. In particular, for each municipality, the Taxable Income indicator (SLIC3) and the Low Income Taxpayers indicator (SLIC4) have been computed as follows

$$\text{SLIC3} = \frac{\text{Taxable income per taxpayer in 2020}}{\text{Taxable income per taxpayer in 2016}} - 1,$$
$$\text{SLIC4} = \frac{\text{T0 in 2020}}{TT \text{ in 2020}},$$

where T0 and TT represent the number of Low Income (€0 - €15,000) Taxpayers (T0) and the Total number of Taxpayers (TT) in 2020, respectively. Thus, SLIC3 and SLIC4 represent the

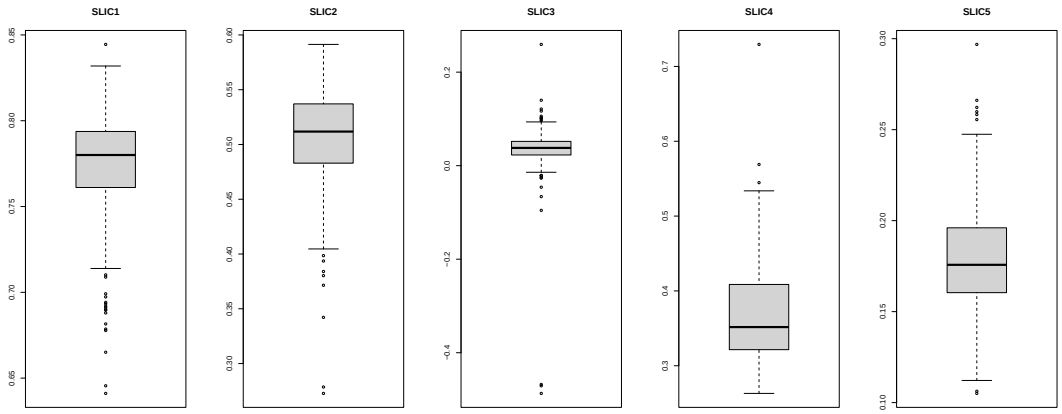

Figure 5.3: Boxplot of the SLIC indicators

change in Taxable Income per taxpayer computed between 2016 and 2020 and the proportion of Low Income Taxpayers. Finally, the Gini Index (SD5) is a summary measure of income inequality, which is commonly used to assess the degree of inequality in the distribution of income (Giorgi and Gigliarano, 2017). It is a concentration index that provides a numerical value between 0 and 1, with a 0 indicating perfect equality (all the individual households have the same income), while a value of 1 reflects maximal inequality among incomes (one person has all the income and all others have none). Thus, the closer the Gini index is to the value 1, the higher the housing tension will be (see the first column of Table 5.5). Minimum (0.105) and maximum (0.297) values of the Gini Index correspond to Jolanda di Savoia (FE) and Gazzola (PC) municipalities, respectively. There is no given expectation for the relationship between the SLIC3 indicator and housing tension; for the SLIC4 indicator, the higher the share of low-incomes is, the more housing deprivation should be observed (see the first column of Table 5.5). Riva del Po (FE) and Zerba (PC) are the municipalities with the minimum and the maximum values of the Taxable Income indicator (SLIC3), respectively; Zola Predosa (BO) and Goro (FE) municipalities are the ones for the SLIC4 indicator (Table 5.5). For the construction of the three latter (SLIC3-SLIC5) indicators, data have been taken by the Ministry of Economy and Finance source (Ministero dell'Economia e delle Finanze, MEF) (see the fourth column of Table 5.5). The box-plots of the SLIC indicators have been reported in Figure 5.3. From the latter figure it emerges that the presence of outliers for each indicator; furthermore, the distribution of the SLIC4 and SLIC5 indicators appear to be slightly skewed.

5.4.3 Housing Supply and Housing Market (HSHM) indicators

For the third pillar (Housing Supply and Housing Market indicators), as mentioned above, nine indicators (HSHM, bottom side of the second column of Table 5.5) have been employed. The HSHM1 (Housing Stock) and the HSHM2 (Change in Housing Stock) indicators concern the housing units area. In particular, the Housing Stock indicator is the proportion of the Total number of Housing Units (THU) of a certain municipality to the Total number of Resident Households (TRH) in that municipality; thus, it has been computed as follows:

$$\text{HSHM1} = \frac{\text{THU in 2020 (housing units)}}{\text{TRH in 2020 (hsd)}},$$

where THU is computed as the sum of the Housing Units by cadastral categories (A01, A02, A03, A04, A05, A06, A07, A08, A09, A11). Thus, Housing Stock (HSHM1) indirectly impacts housing deprivation (see the first column of Table 5.5). Casalecchio di Reno (BO) and Zerba (PC) are respectively two municipalities with the lowest and the highest proportion of Housing Units per household (units/hsd) (Table 5.6). As far as the HSHM2 indicator is concerned, it corresponds to the change in housing stock between 2016 and 2020 and it has been computed as follows:

$$\text{HSHM2} = \frac{\text{THU in 2020 (housing units)}}{\text{THU in 2016 (housing units)}} - 1.$$

For the latter indicator it would be natural to think that a positive change in housing stock means greater housing tension. Moreover, the minimum and the maximum changes in housing stock are observed respectively for Agazzano (PC) and Granarolo nell'Emilia (BO) municipalities (more details have been reported in Table 5.5). The HSHM3 and HSHM4 indicators regard rent area. For each municipality, three variables have been taken into consideration in order to construct the latter indicators: the Average Household Income (€/year) in 2020 (AHI2020), the Maximum Monthly Rent (€/month) in 2018 (MMR2018) and in 2020 (MMR2020); the MMR2018 and MMR2020 variables have been computed by considering a civil dwelling of 80 square meters (sq m). The HSM3 and HSHM4 indicators have been computed as follows:

$$\text{HSHM3} = \frac{(\text{MMR2020} * 12)\ (€/\text{year})}{\text{AHI2020}\ (€/\text{year})},$$
$$\text{HSHM4} = \frac{\text{MMR2020}\ (€/\text{month})}{\text{MMR2018}\ (€/\text{month})} - 1.$$

Thus, for each municipality, they measure the ratio of the Maximum Monthly Rent to the Average Household Income (€/year) and the Change in Maximum Monthly Rent between 2018 and 2020, respectively. Furthermore, both indicators (HSHM3 and HSHM4) are expected to have a direct impact on housing tension. As far as the Ratio of the rent to family income indicator is concerned, if a household pays a high percentage of their income for housing, housing problems are expected to grow up (see the first column of Table 5.5). The latter feature is especially true for the renter households paying over 30% of their income for housing. Thus, it is also true that a positive change in rent (HSM4 indicator) will increase the housing tension due to the fact that, usually, rising rent does not also match to an increase in income.

The HSHM5 and HSHM6 indicators concern housing prices area. For each municipality, they have been computed by taking into account for each municipality the Maximum Housing Price per square meters (€) in 2020 for a civil dwelling (MHP2020), the Average Housing Price per square meters (€) in 2017 (AHP2017) and 2020 (AHP2020) and, finally, the above mentioned AHI2020 variable. The resulting indicators have been constructed as follows:

$$\text{HSHM5} = \frac{(\text{MHP2020} * 80)\ (€)}{\text{AHI2020}\ (€)},$$
$$\text{HSHM6} = \frac{\text{AHP2020}\ (€)}{\text{AHP2017}\ (€)} - 1.$$

Thus, HSHM5 represents the number of annual income years necessary to purchase a house in 2020 (Family income indicator, see the first column of Table 5.5). For this reason, as the latter number increases, low-income households will face housing market access problems. As a consequence, as the renter households increase so too will the housing tension (see the first column of Table 5.5). Similar considerations can also be made for the HSHM6 indicator (Dwelling prices indicator) which represents the changes in Average Housing Price between 2017 and 2020. In fact, a positive change of the latter indicator means that housing problems will increase due to the increase of Housing prices. Finally, three indicators (HSHM7-HSHM9) from the property purchase and the Housing market dynamics area have also been considered. These indicators are the change in averages of the total Number of Normalized Transactions (NNT) computed between two different three-year periods, the average of the ratio of the number of housing units sold to the Total Housing Stock (Housing Stock dynamics index - IMI) and the change in averages of the property purchase computed between two different three-year periods, respectively.

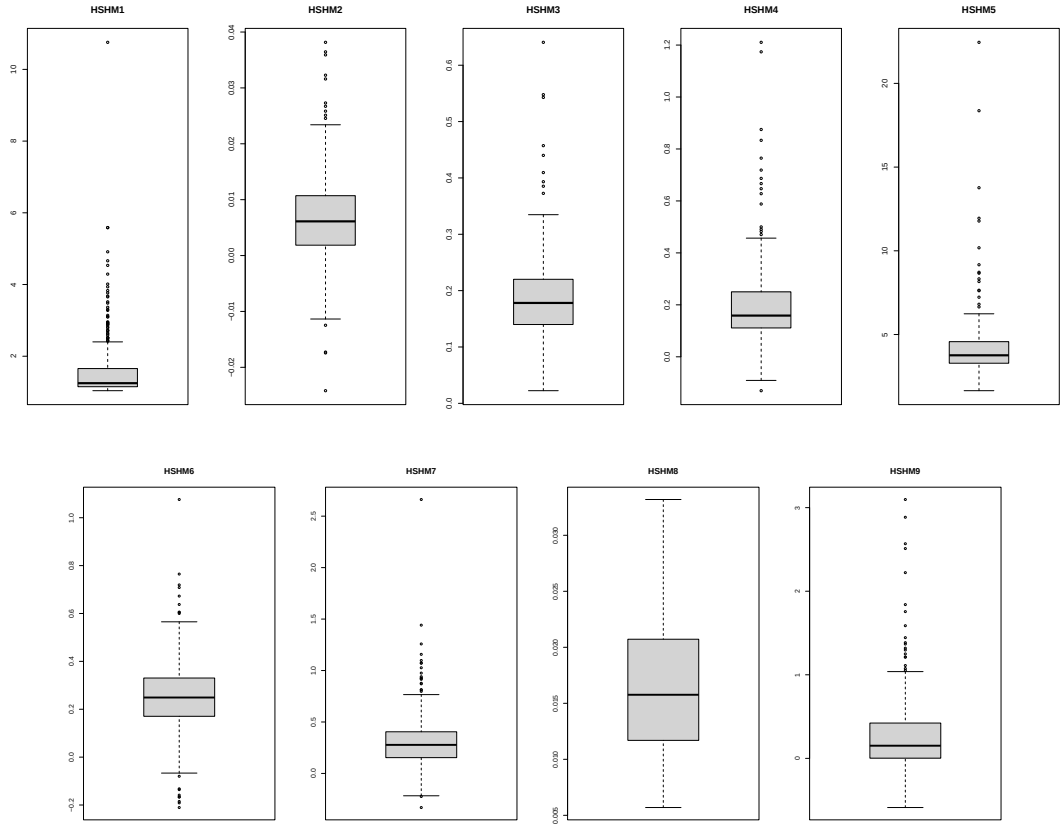

Figure 5.4: Boxplot of the HSHM indicators.

For each municipality, the aforementioned indicators have been computed as follows:

$$\begin{aligned}
\text{HSHM7} =& \frac{\text{Average(NNT in 2018, 2019 and 2020)}}{\text{Average(NNT in 2015, 2016 and 2017)}} - 1, \\
\text{HSHM8} =& \text{Average(IMI in 2016, 2017, 2018, 2019 and 2020)}, \\
\text{HSHM9} =& \frac{\text{Average(NNT in 2018, 2019 and 2020)}}{\text{Average(NNT in 2017, 2018 and 2019)}}.
\end{aligned}$$

As far as the HSHM7 indicator is concerned, a positive change in total Number of Normalized Transactions is expected to be associated with a greater housing tension. Instead, the HSHM8 indicator expresses the share of dwellings bought and sold in a given year and can be interpreted as the measure of the market's dynamism in light of the fact that a greater IMI value corresponds to a greater quota of homes bought and sold, net of the effect of the stock's size. Finally, a positive change in property purchase should increase housing tension. The minimum value of five HSHM indicators corresponds to municipalities coming from the Parma (PR) province. These municipalities are Corniglio (HSHM2), Terenzo (HSHM4), Bore (HSHM6) and Albareto (HSHM3 and HSHM9). The maximum value for these indicators are reached by municipalities

belonging to other provinces: Cervia (RA, HSHM2 and HSHM9), San Giovanni in Persiceto (BO, HSHM3), Riccione (RN, HSHM4) and Borghi (FC, HSHM6). The Borghi municipality also has the maximum value (2.662) for the NNT indicator (HSHM5) while Riolunato (MO) corresponds to the municipality with the highest negative change for the latter indicator. Finally, two municipalities of the Forlì-Cesena province correspond to the highest decrease (Sogliano al Rubicone, FC) and highest increase (Galeata, FC) in property purchase. Data representing the housing market vivacity are drawn from the Osservatorio del Mercato Immobiliare - Agenzia delle Entrate (OMI - Revenue Agency), which is a branch of the country's Inland Revenue Department. The box-plots of the HSHM indicators have been reported in Figure 5.4. The presence of outliers is evident for each HSHM indicator, with the exception of HSHM8 indicator. From the latter figure it also emerges that the distributions of some indicators appear to be skewed.

5.4.4 Housing tension indicators

Two indicators have been considered in order to obtain some information about housing tension: the proportion of the low income households and the proportion of households demanding for public residential housing (Edilizia Residenziale Pubblica, ERP). For the computation of the first indicator, it has been taken into consideration the number of households belonging to the Lowest Income Group (LIG, €0 - €17,154) of the Indicator of Equivalised Economic Situation (ISEE) in 2020. As far as the second indicator is concerned, the number of households in the Ranking of Housing Demand for the Households of the ERR in 2021[9] has been considered (RHDH). Then, for each municipality, the above mentioned indicators have been constructed as follows :

$$\text{ISEE.20} = \frac{\text{LIG (hsd)}}{\text{Resident households in 2020 (hsd)}},$$
$$\text{RANK.21} = \frac{\text{RHDH (hsd)}}{\text{Resident households in 2021 (hsd)}}.$$

Thus, the variables ISEE.20 and RANK.21 are useful indicators of housing tension in municipalities. A high value of ISEE.20 indicates a municipality with a high proportion of LIG (low-income households) relative to the total number of resident households. This suggests that the municipality is facing a housing tension issue. Similarly, a high value of RANK.21 indicates

[9]https://www.comune.bologna.it/bandi/graduatorie-definitive-contributo-affitto-anno-2021

Table 5.7: Descriptive statistics of the ISEE.20 and RANK.21 indicators by provinces.

Indicator	Province	Minimum	First quar.	Median	Average	Third quar.	Maximum
ISEE.20	Bologna	0.020	0.057	0.063	0.066	0.077	0.112
	Forlì-Cesena	0.029	0.061	0.069	0.074	0.084	0.130
	Ferrara	0.027	0.042	0.062	0.056	0.064	0.086
	Modena	0.011	0.043	0.064	0.061	0.080	0.104
	Piacenza	0.000	0.024	0.038	0.040	0.056	0.109
	Parma	0.000	0.027	0.058	0.051	0.069	0.111
	Ravenna	0.042	0.055	0.063	0.064	0.070	0.091
	Reggio Emilia	0.013	0.045	0.058	0.056	0.068	0.089
	Rimini	0.029	0.046	0.061	0.064	0.067	0.200
RANK.21	Bologna	0.003	0.013	0.018	0.019	0.024	0.047
	Forlì-Cesena	0.002	0.014	0.020	0.022	0.027	0.057
	Ferrara	0.000	0.004	0.009	0.010	0.014	0.029
	Modena	0.002	0.014	0.023	0.022	0.030	0.043
	Piacenza	0.000	0.004	0.010	0.015	0.022	0.103
	Parma	0.000	0.004	0.016	0.015	0.021	0.037
	Ravenna	0.010	0.016	0.019	0.020	0.025	0.029
	Reggio Emilia	0.004	0.014	0.019	0.019	0.024	0.048
	Rimini	0.000	0.011	0.017	0.018	0.027	0.046

that the number of households in that municipality seeking housing assistance is high, which is another indication of housing tension. Therefore, higher values of these two indicators are related to municipalities characterized by housing tension. In summary, ISEE.20 and RANK.21 can be used as important measures to identify and evaluate housing tension in municipalities. By using these indicators, policymakers and researchers can assess the extent of the problem and design appropriate policies and interventions to address it. To better describe the geographical distribution of RANK.21, some maps have been reported. In particular, Figure 5.5 shows the distribution of the proportion of households demanding for Public Residential Housing in 2021 for the overall considered municipalities in the ERR. Then, the same distribution is represented for the municipalities within the nine provinces (see Figure 5.6), and the classes of the ATA (see Figure 5.7), the ADA (see Figure 5.8), the Mountain (see Figure 5.9) and NSIA (see Figure 5.10) classifications. In the latter maps, the darker the red color is, the higher the proportion of households demanding for public residential housing is; the opposite is true for the blue color. Table 5.7 shows some descriptive statistics of the ISEE.20 and RANK.21 indicators computed by provinces. The lowest median values for such indicators are observed for the Piacenza and Ferrara provinces, respectively. As far as the highest values are considered, they correspond to Forlì-Cesena (6.9 %) and Modena (2.3 %) provinces for the ISEE.20 and RANK.21 variables, respectively. Some minimum values are equal to 0; specifically, Cerignale (PC) and Valmozzola (PR) municipalities for the first indicator and 14 municipalities for the second one. In particular, they correspond to six municipalities of the Piacenza province (Cerignale, Corte Brugnatella, Morfasso, Ottone, Piozzano and Zerba), five from the Parma province (Compiano, Monchio delle Corti, Tornolo, Valmozzola and Varsi), two from the Rimini province (Casteldelici, Sant'Agata

Table 5.8: Descriptive statistics of the ISEE.20 and RANK.21 indicators for each of the considered classification.

Indicator	Class.	Categories	Minimum	First quar.	Median	Average	Third quar.	Maximum
ISEE.20	ATA	yes	0.052	0.065	0.073	0.077	0.087	0.111
		no	0.000	0.040	0.059	0.055	0.069	0.200
	ADA	yes	0.021	0.059	0.067	0.069	0.079	0.111
		no	0.000	0.034	0.056	0.053	0.068	0.200
	Mountains	yes	0.000	0.029	0.048	0.050	0.069	0.130
		no	0.000	0.051	0.062	0.062	0.071	0.200
	SNAI	UB	0.008	0.054	0.063	0.063	0.070	0.200
		I	0.018	0.050	0.062	0.062	0.073	0.130
		PO	0.065	0.071	0.086	0.087	0.103	0.112
		P	0.000	0.026	0.039	0.043	0.061	0.100
		UP	0.000	0.011	0.019	0.022	0.036	0.048
RANK.21	ATA	yes	0.001	0.020	0.026	0.025	0.030	0.047
		no	0.000	0.009	0.017	0.017	0.023	0.103
	ADA	yes	0.001	0.017	0.022	0.023	0.029	0.047
		no	0.000	0.007	0.014	0.016	0.022	0.103
	Mountains	yes	0.000	0.005	0.012	0.014	0.020	0.057
		no	0.000	0.013	0.018	0.020	0.027	0.103
	SNAI	UB	0.001	0.015	0.020	0.021	0.027	0.103
		I	0.000	0.010	0.016	0.017	0.022	0.057
		PO	0.014	0.024	0.028	0.028	0.031	0.047
		P	0.000	0.003	0.009	0.011	0.019	0.039
		UP	0.000	0.003	0.005	0.006	0.008	0.030

Feltria) and, finally, one from the Ferrara province (Goro). Thus, the latter municipalities correspond to the darker red areas in Figure 5.6. Furthermore, minimum and maximum values of ISEE.20 and RANK.21 within two provinces correspond to the same municipalities: Camugnano and Bologna within the BO province; Verghereto and Galeata within the FC province. Furthermore, the highest proportion (20.0 %) of households with a low income is reached by San Giovanni in Marignano (RN); the highest proportion (10.3 %) for the RANK.21 variable is reached by the Alseno (PC) municipality (the darker blue area in Figure 5.5). Table 5.8 shows some descriptive statistics of the ISEE.20 and RANK.21 indicators for each of the four classifications illustrated in Section 5.3. As far as ISEE.20 is concerned, the highest median values are reached by the municipalities classified as ATA (7.3 %) and ADA (6.7 %); furthermore, no mountains (6.2 %) and Poles municipalities (8.6 %) are the categories with the highest median values for the Mountains and SNAI classifications, respectively. The same also holds for RANK.21 (see the third column of Table 5.8). Thus, the above mentioned municipalities (San Giovanni in Marignano and Alseno) correspond to the following categories for the considered classifications: no ATA, no ADA, no Mountains and UB. Thus, the municipalities of the ERR seem to be characterized by heterogenity in their housing deprivation.

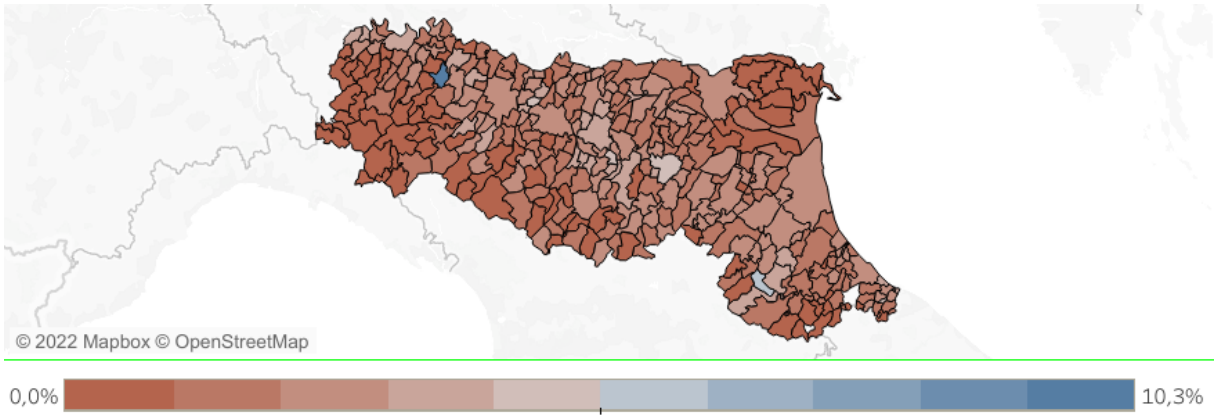

Figure 5.5: Map of the RANK.21 variable in the ERR.

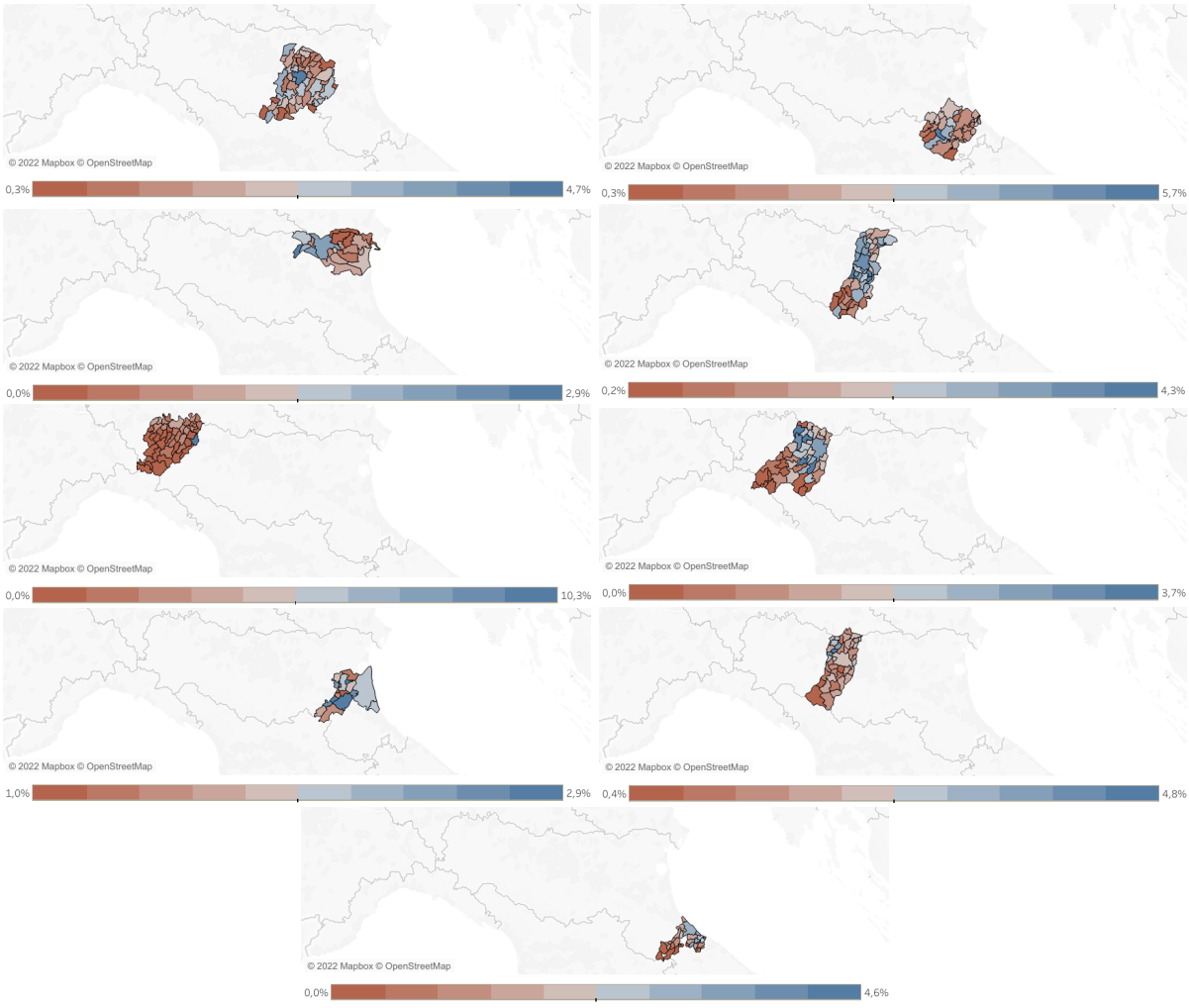

Figure 5.6: Map of the RANK.21 variable in the ERR by provinces (from the left to the right - BO, FC, FE, MO, PC, PR, RA, RE, RN).

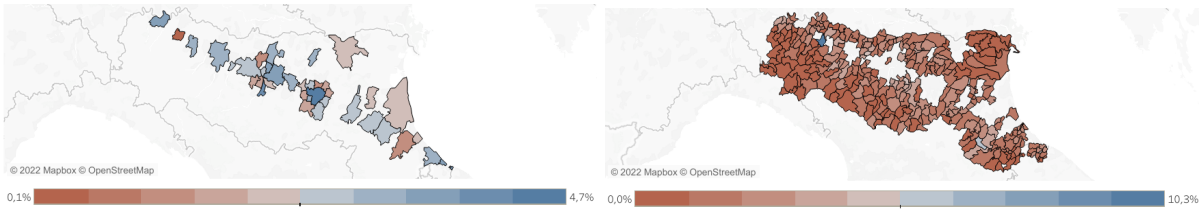

Figure 5.7: Map of the RANK.21 variable in the ERR by ATA classification (from the left to the right - yes/no).

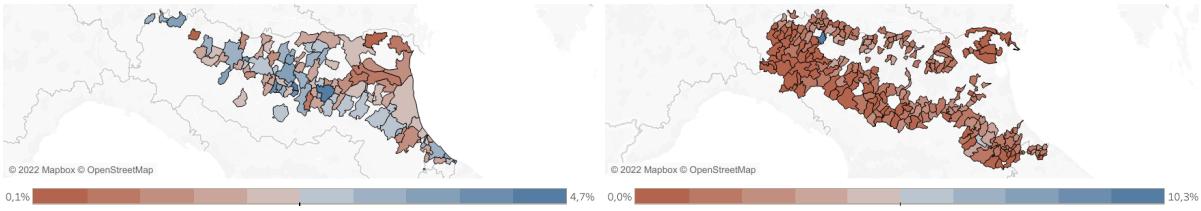

Figure 5.8: Map of the RANK.21 variable in the ERR by ADA classification (from the left to the right - yes/no).

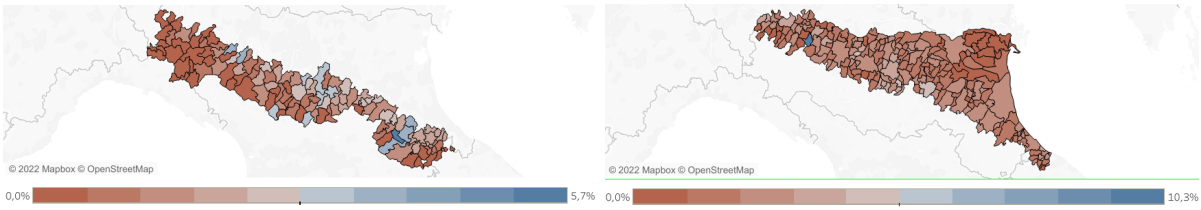

Figure 5.9: Map of the RANK.21 variable in the ERR by the Mountain classification (from the left to the right - yes/no).

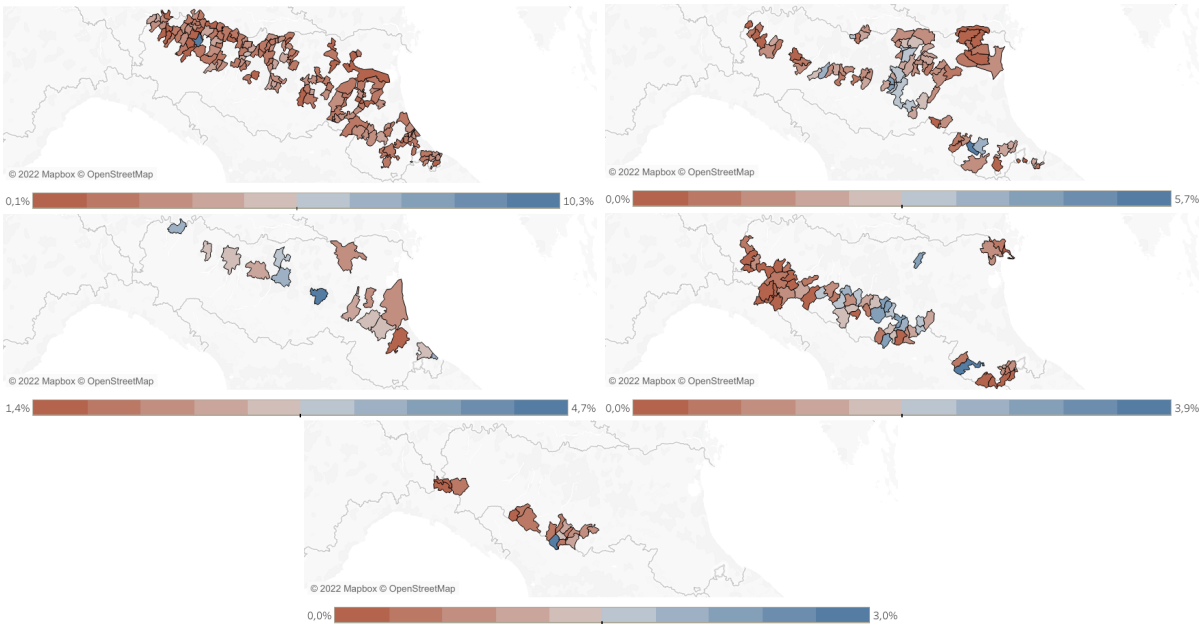

Figure 5.10: Map of the RANK.21 variable in the ERR by the NSIA classification (from the left to the right - UB, I, PO, P, UP).

5.5 Aim of this study

This research aims to study the dependence of housing tension in the municipalities of the ERR on the three types of indicators described in the previous section. From the preliminary analysis of the dataset, the municipalities of the ERR seem to be characterized by unobserved heterogeneity. Thus, in order to manage the possible presence of unknown clusters in the municipalities while performing multivariate regression analysis, mixtures of regression models have been considered. However, it also appears to be reasonable to study the housing deprivation in the municipalities of the ERR through robust methods able to manage the possible presence of outliers, i.e. municipalities whose features noticeably deviate from those registered for the other municipalities. Such municipalities may negatively impact on both the estimation of the regression coefficients and the prediction of the responses. Furthermore, in this research, it may be relevant to specify a system of M=2 regression equations (one equation for each response) with equation-dependent vectors of regressors (i.e., vectors which do not necessarily contain the same regressors for the two responses). In this way, the regressors can be different among dependent variables. For this reason, an approach based on seemingly unrelated regression models (Park, 1993; Srivastava and Giles, 1987) have been exploited, which is able to take into consideration both multivariate correlated responses and to allow each response to depend on its own vector of covariates. In order to perform the analysis, the following vectors of variables have been considered: $\boldsymbol{Y}$=(ISEE.20, RANK.21), $\boldsymbol{X}$=(SD1-SD6, SLIC1-SLIC5, HSHM1-HSHM9). The two dependent variables have bounded support within [0,1]. In order to provide for continuous values in the $(-\infty, +\infty)$ range and to manage values restricted to a finite interval, the most widely accepted solution is the simple logit transformation, originally proposed by Johnson (1949), defined as follows:

$$logit(p) = \ln\Big(\frac{p}{1-p}\Big) = \ln(p) - \ln(1-p), \quad \text{for } p \in (0,1) \tag{5.1}$$

where p may represent, in this analysis, either the proportion of the low income households or the proportion of households demanding for public residential housing within any municipality. However, this transformation cannot be employed in analyses of datasets where p=0 or p=1. To deal with this drawback, Anscombe (2014) and Berkson (1955) proposed the use of the empirical logit trasformation, which is a modified version of (5.1) in which the p=0 and p=1 proportions

are transformed as follows:

$$logit(p) = \ln\Big(\frac{p+t}{1-p+t}\Big), \tag{5.2}$$

where $t = \frac{0.5}{n}$ and n is the number of observations over which p is computed. Thus, the empirical logit transformation in (5.2) is a two-steps transformation defined as follows:

i) for proportions where p is strictly greater than 0 and less than 1, the simple logit transformation is applied as in (5.1);

ii) for proportions where p=0 and p=1, the empirical logit transformation is applied as in (5.2).

However, the latter transformation is sensitive to the value of n, especially when n is low. In fact, in some situations, it may happen that the transformed values obtained by applying (5.2) could become greater than the transformed value obtained with (5.1). A possible solution proposed by Warton and Hui (2011) is to add a small value t (by experimenting with different values) to the proportion p in (5.1). Thus, this approach can be considered as a modification of the empirical logistic transform (see (5.2)).

For each dependent variable examined in this analysis, the t costant has been computed as the smallest positive proportion registered among the municipalities multiplied by 0.5. In particular, the smallest non-zero proportions correspond to the values p=0.0018 and p=0.0008 for ISEE.20 and RANK.21, respectively; thus, the values of the transformed dependent variables, ISEE.20T and RANK.21T, have been obtained using t=0.0009 and t=0.0004, respectively. Figure 5.11 shows the scatterplots of the transformed dependent variables (ISEE.20T and RANK.21T) and the same untransformed variable. From the latter figure appears evident the logarithmic trend of the transformed values for both the dependent variables. Table 5.9 shows some descriptive statistics (minimum, first and third quartile, average and maximum) of the two untransformed (ISEE.20 and RANK.21) and transformed dependent variables (ISEE.20T and RANK.21T). From the latter table it emerges that the values of both the transformed dependent variables are negative because for all municipalities the proportions are lower than 0.5; in particular, the zero values of the ISEE.20 and RANK.21 dependent variables have been transformed in -6.995 and -7.851 values, respectively. Thus, in fact, the response vector is $\boldsymbol{Y}$=(Y_1=ISEE.20T, Y_2=RANK.21T).

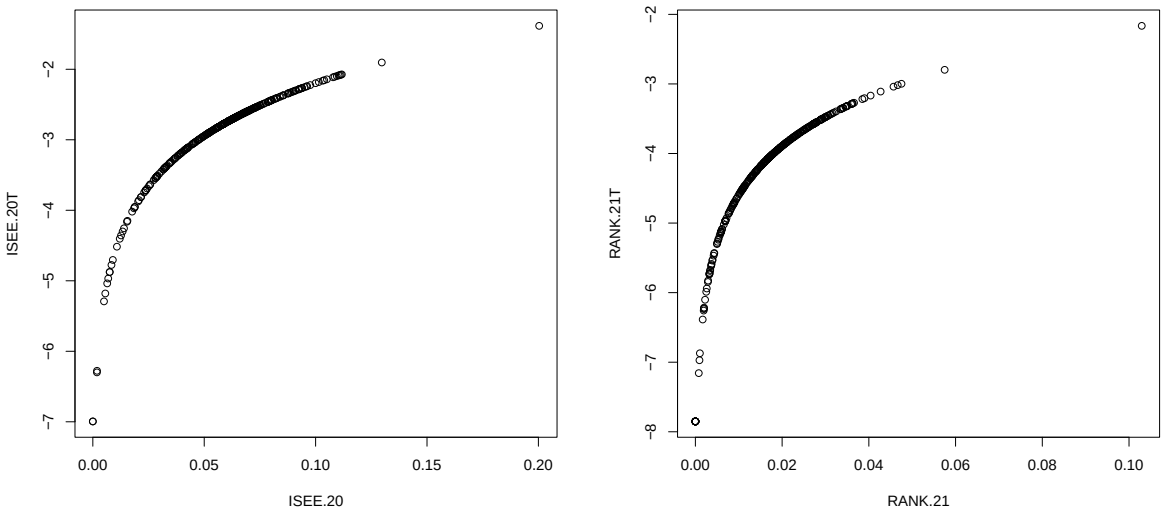

Figure 5.11: Scatterplots of the transformed dependent variables (ISEE.20T and RANK.21T)

Table 5.9: Descriptive statistics of the untransformed (ISEE.20 and RANK.21) and transformed dependent variables (ISEE.20T and RANK.21T)

	ISEE.20	RANK.21	ISEE.20T	RANK.21T
Minimum	0.000	0.000	−6.995	−7.851
First quartile	0.042	0.011	−3.117	−4.578
Median	0.061	0.017	−2.740	−4.049
Average	0.058	0.018	−2.934	−4.333
Third quartile	0.070	0.025	−2.581	−3.658
Maximum	0.200	0.103	−1.384	−2.165

5.6 Methods

For the analysis, the Mixtures of Contaminated Seemingly Gaussian regressions (MCSG) models defined in the equation 2.1 of Chapter 2 of this thesis have been considered, together with MSG models (Galimberti and Soffritti, 2020). Furthermore, a new class of models has been developed. To introduce the latter class, let's start from the hierarchical representation for the MCSG model (see Chapter 2):

$$\mathbf{Y}_i|(\mathbf{x}_i, Z_{ik}=1, U_{ik}=u_{ik}) \sim \begin{cases} N_M\left(\mathbf{y}_i|\mathbf{x}_i; \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*, \boldsymbol{\Sigma}_k\right) & \text{if } u_{ik}=1, \\ N_M\left(\mathbf{y}_i|\mathbf{x}_i; \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^*, \eta_k\boldsymbol{\Sigma}_k\right) & \text{if } u_{ik}=0, \end{cases}$$

$$U_{ik}|Z_{ik}=1 \sim \text{Bernoulli}(\alpha_k), \;\; \mathbf{Z}_i \sim \text{Multinomial}(\pi_1, \ldots, \pi_k),$$

with Bernoulli(α_k) and Multinomial($\pi_1, \ldots, \pi_k$) denoting a Bernoulli distribution with success probability equal to α_k and a K-dimensional multinomial distribution with probabilities

$\pi_1, \ldots, \pi_K$, respectively (details about the latter quantities have been reported in Chapter 2). Thus, the distribution of $\mathbf{y}_i$ depends on the covariates $\mathbf{x}_i$ and the latent variables $\mathbf{z}_i$ and $\mathbf{u}_i$. Furthermore, the latent cluster membership variable $\mathbf{z}_i$ is assumed to be independent on $\mathbf{x}_i$. The new class of models introduced in this section can be obtained by assuming that the latent cluster membership variable $\mathbf{z}_i$ depends on $\mathbf{c}_i$, where $\mathbf{c}_i = (c_{i1}, \ldots, c_{iV})$ is a vector composed of the values of V concomitant variables, which may eventually coincide with some variables of the vector $\mathbf{x}_i$. As a consequence, both the mixing weights and the model parameters of the K component densities of the mixture depend on covariates. To distinguish the roles of such covariates, the terms *concomitant* gating network variables and *explanatory* expert network variables (Gormley and Murphy, 2011) are commonly used, where these expressions are generally employed to denote the covariates which affect the mixing weights ($\mathbf{c}_i$) and the covariates which affect the parameters of the component densities ($\mathbf{x}_i$), respectively. The seemingly unrelated contaminated Gaussian linear clusterwise regression model with concomitant variables (cMCSG) of order K can be introduced, as follows:

$$f(\mathbf{y}_i|\mathbf{x}_i;\boldsymbol{\psi}) = \sum_{k=1}^{K} \pi_k(\mathbf{c}_i) h\left(\mathbf{y}_i;\boldsymbol{\theta}_k(\mathbf{x}_i)\right), \tag{5.3}$$

where $\pi_k(\mathbf{c}_i)$ are the mixing weights of the kth sub-population with $\pi_k(\mathbf{c}_i) > 0$ and $\sum_{k=1}^{K} \pi_k(\mathbf{c}_i) = 1$; when $K \geq 2$, they are allowed to depend on covariates $\mathbf{c}_i$. These mixing weights are modeled using multinomial logistic regression as follows:

$$\pi_k(\mathbf{c}_i) = \frac{exp(\tilde{\mathbf{c}}_i'\boldsymbol{\gamma}_k)}{1 + \sum_{k=2}^{K} exp(\tilde{\mathbf{c}}_i'\boldsymbol{\gamma}_k)} \quad k = 2, \ldots, K. \tag{5.4}$$

with the first component $\boldsymbol{\gamma}_1 = (0, \ldots, 0)'$ as baseline; $\boldsymbol{\gamma}_k$ is a $(V+1)$-dimensional vector of regression parameters for the concomitant variables, and $\tilde{\mathbf{c}}_i = (1, \mathbf{c}_i)$; furthermore, $h\left(\mathbf{y}_i;\boldsymbol{\theta}_k(\mathbf{x}_i)\right)$ is the same defined in equation 2.2 of Chapter 2 but with a modified notation which highlights that the parameter $\boldsymbol{\theta}_k$ depends on $\mathbf{x}_i$; thus, $\boldsymbol{\theta}_k(\mathbf{x}_i)$ is the same quantity represented by $\boldsymbol{\theta}_k$ in Chapter 2. As a consequence of the assumptions just described, the following hierarchical

representation for $\mathbf{Y}_i|\mathbf{x}_i$ can be obtained in association with the new class of cMCSG models:

$$\begin{aligned}
&\mathbf{Z}_i = (Z_{i1}, \ldots, Z_{ik}, \ldots, Z_{iK}) \sim \textit{Multinomial}(1, \pi_1(\mathbf{c}_i), \ldots, \pi_K(\mathbf{c}_i)),\\
&P(Z_{ik} = 1|\mathbf{c}_i) = \pi_k(\mathbf{c}_i),\\
&U_{ik}|Z_{ik} = 1 \sim \textit{Bernoulli}(\alpha_k),\\
&\mathbf{Y}_i|\mathbf{x}_i, Z_{ik} = 1, U_{ik} = u_{ik} \sim \begin{cases} N_M\Big(\boldsymbol{\mu}_{ik}(\mathbf{x}_i; \boldsymbol{\beta}_k^*), \boldsymbol{\Sigma}_k\Big) & \textit{if} \;\; u_{ik} = 1,\\ N_M\Big(\boldsymbol{\mu}_{ik}(\mathbf{x}_i; \boldsymbol{\beta}_k^*), \eta_k\boldsymbol{\Sigma}_k\Big) & \textit{if} \;\; u_{ik} = 0, \end{cases}
\end{aligned}$$

where $\mathbf{z}_i$ has now a multinomial distribution with a single trial and probabilities equal to $\pi_k(\mathbf{c}_i)$, $\boldsymbol{\mu}_{ik}(\mathbf{x}_i; \boldsymbol{\beta}_k^*)$ is the conditional expected value of $\mathbf{Y}_i|\mathbf{X} = \mathbf{x}_i$ in the kth sub-population (see equation (2.3) of Chapter 2). Thus, the complete-data likelihood function is now equal to

$$\begin{aligned}
L_c(\boldsymbol{\psi}) = \prod_{i=1}^{I}\prod_{k=1}^{K}\Big\{\pi_k(\mathbf{c}_i)\Big[\alpha_k\phi_M\Big(\mathbf{y}_i; \boldsymbol{\mu}_k(\mathbf{x}_i; \boldsymbol{\beta}_k^*), \boldsymbol{\Sigma}_k\Big)\Big]^{u_{ik}}\\
\Big[(1-\alpha_k)\phi_M\Big(\mathbf{y}_i; \boldsymbol{\mu}_k(\mathbf{x}_i; \boldsymbol{\beta}_k^*), \eta_k\boldsymbol{\Sigma}_k\Big)\Big]^{1-u_{ik}}\Big\}^{z_{ik}}.
\end{aligned}$$

Up to an additive constant, the complete-data log-likelihood function employed in the ECM algorithm for the computation of the parameter estimates can be expressed as follows:

$$\begin{aligned}
\ell_c(\boldsymbol{\psi}) = \sum_{i=1}^{I}\sum_{k=1}^{K} z_{ik}\Big[\ln\pi_k(\mathbf{c}_i) + u_{ik}\ln\alpha_k + (1-u_{ik})\ln(1-\alpha_k) - \frac{1}{2}\ln|\boldsymbol{\Sigma}_k| +\\
-\Big(\frac{M}{2}\ln\eta_k\Big)(1-u_{ik}) - \frac{1}{2}\Big(u_{ik} + \frac{1-u_{ik}}{\eta_k}\Big)\delta^2_{\boldsymbol{\Sigma}_k}\Big(\mathbf{y}_i, \boldsymbol{\mu}_k(\mathbf{x}_i; \boldsymbol{\beta}_k^*)\Big)\Big],
\end{aligned}$$

where $\delta^2_{\boldsymbol{\Sigma}_k}(\mathbf{y}_i, \boldsymbol{\mu}_k(\mathbf{x}_i; \boldsymbol{\beta}_k^*))$ is the squared Mahalanobis distance defined in equation (2.5) of the Chapter 2. Given the current parameter value $\boldsymbol{\psi}^{(h)}$, in the E-step on the hth iteration of the ECM algorithm the estimated posterior probabilities that the ith observation come from the kth sub-population and the same observation is a typical point of such a sub-population are now equal to:

$$\begin{aligned}
\hat{z}_{ik}^{(h)} &= \frac{\pi_k^{(h)}(\mathbf{c}_i)h\left(\mathbf{y}_i|\mathbf{x}_i; \boldsymbol{\theta}_k^{(h)}(\mathbf{x}_i)\right)}{f\left(\mathbf{y}_i|\mathbf{x}_i; \boldsymbol{\psi}^{(h)}\right)}\\
\hat{u}_{ik}^{(h)} &= \frac{\alpha_k^{(h)}\phi\Big(\mathbf{y}_i|\mathbf{x}_i; \tilde{\mathbf{x}}_i^{*\prime}\boldsymbol{\beta}_k^{*(h)}, \boldsymbol{\Sigma}_k^{(h)}\Big)}{h\left(\mathbf{y}_i|\mathbf{x}_i; \boldsymbol{\theta}_k^{(h)}(\mathbf{x}_i)\right)},
\end{aligned}$$

respectively, where $f(\mathbf{y}|\mathbf{x};\boldsymbol{\psi}) = \sum_{k=1}^{K}\pi_k(\mathbf{c}_i)h\left(\mathbf{y}|\mathbf{x};\boldsymbol{\theta}_k(\mathbf{x}_i)\right)$. The update for $\pi_k^{(h)}(\mathbf{c}_i)$ is now computed as follows:

$$\pi_k^{(h+1)}(\mathbf{c}_i) = P(\hat{z}_{ik}^{(h+1)} = 1) = \frac{exp(\mathbf{c}_i^{'}\boldsymbol{\gamma}_k)}{1+\sum_{k=2}^{K}exp(\mathbf{c}_i^{'}\boldsymbol{\gamma}_k)} \quad k = 2,\ldots,K. \tag{5.5}$$

It is worth noting that in the absence of concomitant variables, the update for the equation (5.5) is the one defined in Section (2.2.4), i.e. $\pi_k^{(h+1)} = \frac{1}{I}\sum_{i=1}^{I}\hat{z}_{ik}^{(h)}$. As far as the updtates for the other parameters are concerned, they are equal to the ones defined in equations (2.8), (2.9), (2.10) and (2.11). The strategies for the initialisation of $\boldsymbol{\psi}$ and convergence criteria are similar to the ones illustrated in Section 2.2.5 of Chapter 2. The only difference is that, in order to obtain starting values for the component weights $\pi_k(\mathbf{c}_i)$, the function `MoEClust` (Murphy and Murphy, 2020) has been employed. In particular, this latter function has been employed to fit a Gaussian mixture model with V concomitant variables and K components to the sample residuals of a seemingly unrelated linear regression model (Srivastava and Giles, 1987) through the package `systemfit` (Henningsen and Hamann, 2007) in the `R` environment (R Core Team, 2021). Hence, the gating network parameters $\boldsymbol{\gamma}_k^{(h)}$ are updated through the function `multinom` from the `nnet` package (Venables and Ripley, 2013) with the dependent variables given by the a posteriori probability estimates $\hat{z}_{ik}^{(h)}$. The parameter vector of model (5.3) is given by $\boldsymbol{\psi} = (\boldsymbol{\psi}_1,\ldots,\boldsymbol{\psi}_k,\ldots,\boldsymbol{\psi}_K)$, where $\boldsymbol{\psi}_k = (\boldsymbol{\gamma}_k, \boldsymbol{\theta}_k(\mathbf{x}_i))$. As concerns the formula for the implementation of the Bayesian information criterion defined in Section (2.2.6), the number of free parameters is now equal to $n_{\boldsymbol{\psi}} = 2K + K(P^* + M) + K\frac{M(M+1)}{2} + (V+1)(K-1)$. For $\alpha_k \to 1$ or $\eta_k \to 1$ $\forall k$, model (5.3) reduces to a mixture of seemingly unrelated Gaussian regressions model with concomitant variables (here denoted with the acronym cMSG) of order K. Such a mixture can be seen as an extension of the MSG model (Galimberti and Soffritti, 2020). It is worth noting that, in the absence of concomitant variables and when the vectors of predictors selected for the two dependent variables coincide (i.e., $\mathbf{X}_1 = \mathbf{X}_2$), the resulting model belongs to either the Mixtures of multivariate Contaminated Gaussian regressions models (MCG, Mazza and Punzo (2020)) or the Mixtures of multivariate gaussian Regressions Models (MRM, Jones and McLachlan (1992)).

5.7 Results

As above mentioned, the analysis has been carried out through MSG, MCSG, cMSG and cMCSG models. For the models defined by (5.3), the vector of the concomitant variables employed in the analysis is $\boldsymbol{c}$=(TRP, TRF). Models from MSG and MCSG classes have been estimated for $K \in \{1, 2, 3, 4, 5\}$ and each of the parameterisations for the covariance matrices reported in Table 3.1 of Chapter 3. As far as cMSG and cMCSG models are concerned, they have been estimated for $K \in \{2, 3, 4, 5\}$ and without imposing any constraint on the covariance matrices (i.e., with the `VVV` parameterisation).

In every scenario where predictive modeling or data analysis is being performed, it is crucial to implement an effective variable selection strategy. Without an appropriate variable selection strategy, in fact, some variables may have a negligible effect on the outcomes and the model may suffer from overfitting (Chowdhury and Turin, 2020). Fewer variables in the model translate to less computational time and less complexity, and are also preferred according to the principle of parsimony, which prioritizes simpler models with fewer variables over complex ones. Complex models that contain many variables make the model more dependent on the observed data, while simpler models are easier to interpret, generalize, and use in practice. However, it is also crucial to ensure that no relevant variables are excluded from the selected model, which may impact the accuracy and reliability of the model. Therefore, selecting the right set of variables is important to balance the practicality and simplicity of the model with the need to include essential variables for accurate and meaningful results (Chowdhury and Turin, 2020). Thus, it is important to possibly embed variable selection in the analysis. This is also the case of the MSG, MCSG, cMSG and cMCSG models, where the selection of the appropriate variables to be included in the regression equations is one important challenge. In general, an exhaustive search for the optimal subset of predictors should be carried out. However, it is often impractical or impossible to exhaustively search the entire space of possible subsets of predictors to be included in the model, particularly in the presence of complex and high-dimensional problems. Therefore, various non-exhaustive strategies have been proposed in the literature. Such methods include stepwise selection, forward selection and backward elimination (Crawford and Hoel (1972); Mallows (1973); Miller (1984); Sutter and Kalivas (1993)), interactive variable selection (Lindgren et al. (1994); Miller (1991)), automatic variable selection (Breiman (1996); Forina et al. (1986)), cyclic subspace regression (Bakken et al. (1999); Jolliffe and Cadima (2016)),

and many other. However, some methods have the drawback of failing to select regressors that may be of no value when used alone but offer useful information when combined. For instance, forward selection adds regressors until a specific selection criterion is minimized or maximized, but it cannot remove a regressor once it has been included. The same applies to backward elimination, where a removed regressor cannot be reinserted. Thus, they can fail to find the global optimum. Furthermore, when the number of regressors is high, these traditional selection methods can be computationally expensive and lead to unstable results. To overcome this disadvantage, penalized likelihood methods (Jianqing and Runze (1999); Tibshirani (1996)) can be used, which are statistical techniques used for variable selection in regression models and that are able to manage a large number of regressors, by producing stable and interpretable models. It involves modifying the likelihood function of a multivariate regression model by adding a penalty term to the likelihood function. The penalty term is designed to shrink some of the estimated regression coefficients towards zero, resulting in a more parsimonious model with fewer regressors. Furthermore, there are various types of probabilistic optimization techniques that can be used to solve subset selection issue through stochastic iterative algorithms. Therefore, optimization algorithms and techniques are developed to efficiently search the space and identify near-optimal solutions. These algorithms consider different subsets of regressors and evaluate their performance through an objective function (for example, the Akaike Information Criterion (Akaike, 1974) and the Bayesian Information Criterion (Schwarz, 1978). After assessing the performance of different subsets of regressors using an objective function, these algorithms generate new subsets from the existing ones. Most techniques in this category employ genetic algorithms (Michalewitz (1996); Goldberg (1989)) or simulated annealing (Kirkpatrick et al., 1983) as a search algorithm.

In the analyses of this chapter, for each model class, the backward elimination technique and a genetic algorithm have been employed in order to select the relevant regressors for each regression equation. In particular, the genetic algorithm exploits principles and operators of the biological evolution of a species (see, for example, Goldberg (1989), Chatterjee et al. (1996), Scrucca (2016)). The algorithm employed in this analysis is similar to a genetic algorithm introduced in Galimberti et al. (2018). It follows these steps:

- the chromosomes (ordered sequences of genes) that compose an initial population are randomly generated and examined; each gene can take on a value of either 0 or 1. These genes can be thought of as Bernoulli variables with a success probability of q. The chro-

mosomes themselves are randomly generated by independently drawing realizations from this Bernoulli distribution. This means that each gene in a chromosome is randomly assigned a value of 0 or 1 with equal probability. The purpose of generating chromosomes in this way is to explore a wider range of potential solutions to the problem being optimized, rather than being limited to a pre-specified set of starting solutions; furthermore, its fitness is evaluated;

- in order to generate novel populations composed of chromosomes characterised by improved fitness values, an iterative evolution process is performed, based on three genetic operators; in particular,
  - the crossover operator, which is a random process of genome recombination that applies to pairs of chromosomes to create new offspring; the process involves selecting two parent chromosomes, and then randomly selecting a crossover point along their sequence of genes. The genetic material from one parent before the crossover point is combined with the genetic material from the other parent after the crossover point to create two new offspring chromosomes. The uniform distribution is used in the crossover process, where the crossover point is randomly selected with equal probability anywhere along the chromosome. This ensures that the offspring chromosomes are diverse and have genetic material from both parent chromosomes.
  - the mutation operator, which is a random alteration of a gene in a chromosome by flipping its value from 0 to 1 or from 1 to 0; a Bernoulli distribution is used to randomly generate the new value for the mutated gene, with success probability of w.
  - the selection operator, which is a weighted random sampling from the initial population with weights proportional to the chromosomes' fitness; thus, the chromosomes selected in this way reproduce and their offspring will compose a novel generation, obtained after crossover and mutation; in this technique, each chromosome's fitness value is used as the weight or probability of selection. The higher the fitness value of a chromosome, the higher its weight or probability of being selected. This means that the fittest chromosomes have a higher chance of being selected than the less fit ones. To perform weighted random sampling, the algorithm generates a random number between 0 and 1, and then iterates through the chromosomes in the popu-

lation, adding up the weights until the cumulative sum exceeds the random number. The chromosome corresponding to the weight that causes the sum to exceed the random number is then selected. This process is repeated until the desired number of chromosomes have been selected to create the next generation of solutions.

- the chromosomes of the resulting novel generation are assigned their fitness, and the evolution process repeats; the algorithm stops when a maximum number of population has been generated.

In this approach, each model is represented as a chromosome and its fitness is measured by the BIC. For each dependent variable, the examined chromosome has a binary gene for each candidate regressor in $\mathbf{X}$, where 1 and 0 values denote whether any given candidate regressor has been selected or not. For MCSG and MSG models, the chromosome also contains an additional gene which can take values from the set $\{1, \ldots, 14\}$ so as to distinguish the 14 parsimonious parameterisations for the covariance matrices. The genetic algorithm has been devised so as to explore subspaces of the model space associated with each model class in which the value of K is fixed. It is important to note that the process of selecting a model in this latter framework can be quite complex, especially when dealing with high-dimensional datasets (here, the total number of candidate regressors for each regression equation is 21). The effectiveness of a genetic algorithm depends on the extent of exploration of the model space. However, there is no general rule for choosing the appropriate population size and number of generations for a genetic algorithm.

The genetic algorithm has been implemented in `R` through the package `GA` (Scrucca, 2013). The values of q and w that have been used in the analyses are equal to 0.5. Each execution requires the specification of two tuning parameters: dimension of the examined population (N) and maximum number of generations to be examined (d_{max}). For each model class and each examined value of K, twelve independent executions of this algorithm have been performed, one for each combination of the following values for the tuning parameters: N=200, 300, 400, 500; d_{max}=30, 40, 50. Thus, for each model class, the number of models that have been examined is about 400000.

As far as the analyses with the backward elimination technique are concerned, for each value of K and each parsimonious parameterisation, the process starts with fitting a model with all the candidate regressors included in both the regression equations and, thus, the Bayesian Information Criterion (BIC) is computed; in this approach, after fixing a regression equation,

the regressors are iteratively removed one by one from the other equation. Each model is then refitted without the removed variable and the Bayesian Information Criterion (BIC) is again computed. Finally, the BIC of the original model is compared with the BICs of the models obtained by removing a single regressor. The process of variable elimination is then repeated, starting from this improved model. Through iterative removal of variables and re-evaluation of the BIC, the goal is to identify a model with a higher BIC, indicating a better fit to the data. The iterative process continues until no further removal of regressors leads to a higher BIC. This final model represents the optimal combination of variables (limited to the examined models) for the regression analysis. Thus, using the BIC as a guide for variable elimination helps to identify the most important predictors and simplifies the regression equation, potentially improving its interpretability and predictive performance. It is worth noting that the backward elimination technique, while widely used, may not always result in the selection of the best model, as it can fail to find the global optimum. In contrast, the genetic algorithm technique is a modern approach to model selection that can efficiently search for the optimal model by exploring a large search space of possible models.

Table 5.10 and Table 5.11 report the models which best fitted the analysed dataset according to the BIC for each examined value of K within each model class by using the genetic algorithm and the backward elimination technique. Overall, it seems that the best trade-off between the fit and complexity can be obtained using the MCSG model with $K = 2$ clusters of municipalities (BIC=-764.6) identified using the genetic algorithm (see Table 5.10). The convergence of the ECM algorithm for the parameter estimation of the latter model has been reached after 198 iterations. As far as the other model classes and the genetic algorithm analysis are concerned, the best MSG and cMSG models have $K = 4$ clusters (BIC=-779.0 and BIC=-793.6, respectively), while cMCSG models of order $K = 3$ should be preferred (BIC=-779.3) (see Table 5.10). The analysis with the backward elimination technique has shown that the best MSG and MCSG models have $K = 2$ clusters (BIC=-875.2 and BIC=-810.1, respectively) while the best cMSG and cMCSG models have $K = 5$ clusters and the same BIC (-803.8). It is worth noting that in the context of the 18 summaries listed in Table 5.10 and Table 5.11, in only one instance (cMCSG model class and K=5), using the backward elimination technique resulted in the selection of a model with a higher BIC than the one obtained through the genetic algorithm. As far as the overall best model is concerned, the proportion of the low income households is regressed on three socio-demographic indicators (Population Density, Change in household and Change in

Table 5.10: Selected regressors, maximised log-likelihood $\ell(\hat{\boldsymbol{\psi}})$ and values of BIC for the best models within the classes MSG, MCSG, cMSG and cMCSG in the analysis of housing tension in the municipalities of the ERR through genetic algorithm.

Model class	K	Acronym	$\mathbf{X}_1$	$\mathbf{X}_2$	$\ell(\hat{\boldsymbol{\psi}})$	n_{ψ}	BIC	Model class	K	Acronym	$\mathbf{X}_1$	$\mathbf{X}_2$	$\ell(\hat{\boldsymbol{\psi}})$	n_{ψ}	BIC
MSG	1	EEE	SD5, SD6, SLIC4, HSHM3	SD3, SD5, SD6, SLIC4, HSHM5	−543.6	14	−1168.2	MCSG	1	EEE	SD1, SD3, SD6, SLIC5, HSHM1, HSHM2	SD1, SD3, SLIC3, SLIC4, HSHM1, HSHM8	−376.9	19	−863.8
MSG	2	VVV	SD5, SD6, HSHM1	SD5, SD6, SLIC4, HSHM5	−369.4	25	−883.6	MCSG	2	VVV	SD1, SD3, SD4, SLIC1, HSHM1, HSHM2, HSHM6, HSHM7, HSHM8,	SD1, SD3, SD4, SD5, SLIC2, SLIC5, HSHM1, HSHM2, HSHM5, HSHM9	−228.8	53	−764.6
MSG	3	VVV	SD2, SD4, SD5, SLIC2, HSHM1, HSHM8, HSHM9	SD1, SD3, SD4, SD6, SLIC2, SLIC4, HSHM1, HSHM5, HSHM6	−220.8	65	−818.2	MCSG	3	VVV	SD2, SD4, SD5, SLIC2, HSHM1, HSHM8 HSHM9	SD1, SD3, SD4, SD6, SLIC2, SLIC4, HSHM1, HSHM5, HSHM6,	−220.8	71	−852.9
MSG	4	VVV	SD2, SD5, SD6, SLIC1, HSHM1	SD2, SD5, SD6, SLIC4, HSHM3, HSHM6, HSHM7, HSHM8, HSHM9	−160.7	79	−779.0	MCSG	4	VVV	SD1, SD3, SD5, SD6, SLIC5, HSHM1, HSHM2, HSHM6	SD2, SD3, SD4, SD5, SLIC1, SLIC3, SLIC4, HSHM2, HSHM3, HSHM4, HSHM5, HSHM8	−82.8	111	−808.6
MSG	5	VVV	SD4, SD5, SLIC3, SLIC1, HSHM1	SD1, SD2, SD3, SLIC4, HSHM3, HSHM6, SLIC5, HSHM1, HSHM3, HSHM5, HSHM6, HSHM8, HSHM9	−42.6	124	−803.6	MCSG	5	VVV	SD2, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, HSHM1, HSHM4, HSHM6, HSHM9	SD1, SD3, SD4, SD6, SLIC5, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM9	33.7	164	−882.7
cMSG	2	VVV	SD1, SD4, SD5, HSHM1	SD1, SD4, SD5, SLIC4, HSHM5	−357.4	31	−894.3	cMCSG	2	VVV	SD5, SD6	SD1, SLIC2	−320.6	25	−786.0
cMSG	3	VVV	SD5, SLIC3, SLIC4, HSHM1, HSHM2, HSHM4, HSHM7	SD1, SD3, SLIC1, SLIC2, SLIC5, HSHM1, HSHM3, HSHM5, HSHM6, HSHM7, HSHM9	−198.0	75	−830.6	cMCSG	3	VVV	SD1, SD2, SD4 SLIC1, SLIC5, HSHM1, HSHM2, HSHM4, HSHM6, HSHM9	SD2, HSHM1, HSHM2, HSHM5, HSHM6,	−181.1	72	−779.3
cMSG	4	VVV	SD4, SD5, SLIC3, SLIC4, SLIC5, HSHM1, HSHM3	SD1, SD2, SD4, SD5, SLIC1, SLIC2, SLIC4, HSHM2, HSHM4, HSHM6, HSHM7, HSHM8, HSHM9	−81.1	109	−793.6	cMCSG	4	VVV	SD1, SD3, SD4, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, HSHM1, HSHM5, HSHM7, HSHM8	SD2, SD4, SD5, SLIC1, SLIC2, SLIC4, HSHM2, HSHM4, HSHM5, HSHM6, HSHM9	−20.6	133	−811.6
cMSG	5	VVV	SD4, SLIC1, SLIC3, SLIC5, HSHM1, HSHM3, HSHM4, HSHM5, HSHM7, HSHM9	SD1, SD2, SD3, SD5, SD6, HSHM1, SLIC2, SLIC4, HSHM2, HSHM6, HSHM8	14.2	142	−794.3	cMCSG	5	VVV	SD1, SD3, SD5, SD6, SLIC2, SLIC3, HSHM1, HSHM6, HSHM8,	SD2, SD4, SD6, SLIC1, SLIC2, SLIC4, HSHM2, HSHM3, HSHM4, HSHM7, HSHM9	16.5	147	−818.7

Table 5.11: Selected regressors, maximised log-likelihood $\ell(\hat{\boldsymbol{\psi}})$ and values of BIC for the best models within the classes MSG, MCSG, cMSG and cMCSG in the analysis of housing tension in the municipalities of the ERR through backward elimination selection.

Model class	K	Acronym	$\mathbf{X}_1$	$\mathbf{X}_2$	$\ell(\hat{\boldsymbol{\psi}})$	n_{ψ}	BIC
MSG	1	VVV	SD5, SD6, SLIC4, HSHM3	SD1, SD4, SD5, SLIC4, HSHM3, HSHM8	−540.7	15	−1168.3
MSG	2	VEE	SD1, SD4, SD5 HSHM1	SD1, SD4, SLIC5, SLIC4	−379.7	25	−904.3
MSG	3	VVV	SD1, SD2, SD3, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, HSHM1, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM9	−251.0	122	−916.3
MSG	4	VEE	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM7, HSHM8, HSHM9	−73.7	173	−875.2
MSG	5	VVE	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, HSHM1, HSHM2, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	175.8	220	−922.8
cMSG	2	VVV	SD4, SD5, SD6, SLIC2, SLIC4, HSHM1, HSHM2, HSHM3	SD1, SD3, SD5, SLIC2, SLIC4, HSHM1, HSHM3, HSHM8	−329.6	45	−919.9
cMSG	3	VVV	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	−75.9	135	−933.9
cMSG	4	VVV	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD6, SLIC1, SLIC2, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	61.9	181	−924.8
cMSG	5	VVV	SD1, SD2, SD3, SD4, SD5, SD6, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM9	241.1	222	−803.8
MCSG	1	EEE	SD4, SD6, SLIC4, SLIC5, HSHM1, HSHM2 HSHM3	SD1, SD3, SLIC4, HSHM1, HSHM3, HSHM8	−378.9	20	−873.7
MCSG	2	VEE	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	−147.3	89	−810.1
MCSG	3	EEV	SD1, SD2, SD3, SD4, SD5, SD6, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	−47.7	133	−903.9
MCSG	4	VEI	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	−79.5	180	−884.8
MCSG	5	VVE	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM6, HSHM7, HSHM8, HSHM9	200.2	230	−931.9
cMCSG	2	VVV	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8	SD1, SD2, SD3, SD4, SD5, HSHM1, SLIC2, SLIC3, SLIC4, HSHM1, HSHM2, HSHM3, HSHM2, HSHM3, HSHM4, HSHM5, HSHM9	−189.2	81	−847.6
cMCSG	3	VVV	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM6, HSHM7, HSHM8	−78.8	144	−991.8
cMCSG	4	VVV	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM6, HSHM9	104.4	185	−862.9
cMCSG	5	VVV	SD1, SD2, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9	SD1, SD3, SD4, SD5, SD6, SLIC1, SLIC2, SLIC3, SLIC4, SLIC5, SLIC6, HSHM1, HSHM2, HSHM3, HSHM4, HSHM5, HSHM6, HSHM7, HSHM8, HSHM9,	299.0	242	−803.8

Table 5.12: Estimates of $\boldsymbol{\pi}$, $\boldsymbol{\alpha}$, $\boldsymbol{\eta}$ and $\boldsymbol{\Sigma}$ under the overall best model for the analysis of housing tension in the municipalities of the ERR.

$\hat{\boldsymbol{\psi}}$	$k=1$	$k=2$
$\hat{\pi}_k$	0.055	0.945
$\hat{\alpha}_k$	0.999	0.829
$\hat{\eta}_k$	1.000	12.316
$\hat{\boldsymbol{\Sigma}}_k$	$\begin{pmatrix} 0.676 & 0.034 \\ 0.034 & 0.002 \end{pmatrix}$	$\begin{pmatrix} 0.056 & 0.057 \\ 0.057 & 0.157 \end{pmatrix}$

foreigners), one Social Life and Income Condition indicator (Education) and five Housing Supply and Housing market (Housing Stock, Change in Housing Stock, Dwelling prices, NNT and IMI) indicators. Thus, $P_1 = 9$ regressors have been selected for the first equation of the regression model. The selected regressors for the proportion of households demanding for public residential housing are the same three SD indicators selected by the proportion of the low income households together with Household Size, two SLIC (Taxable Income and Gini Index) and four HSHM indicators (Housing Stock, Change in Housing Stock, Family Income and Property purchase). Thus, $P_2 = 10$. It is worth noting that the majority of the selected indicators belong to the SD and HSHM macro-areas. The estimates of $\boldsymbol{\pi}$, $\boldsymbol{\alpha}$, $\boldsymbol{\eta}$ and $\boldsymbol{\Sigma}$ of the overall best MCSG model are reported in Table 5.12, while the estimates of the regression coefficients have been reported in the fourth column of Table 5.13, together with the estimates of their standard errors (column 6) computed by the parametric bootstrap approach. In particular, 100 bootstrap samples have been generated from the selected model. Thus, they have been utilized to compute 100 estimates of the parameters for the selected model. The standard deviation of such estimates has been employed as the estimated standard errors of the estimated regression coefficients. Furthermore, tests for the hypotheses H_0: β_{kmp}=0 for $k = 1, 2$, $m = 1, 2$, $p = 1, \ldots, P_m$, have been run under an asymptotic normal distribution using the z_{kmp} statistics, where $z_{kmp} = \frac{\hat{\beta}_{kmp}}{se(\hat{\beta}_{kmp})}$, with $se(\hat{\beta}_{kmp})$ denoting the estimated standard error of $\hat{\beta}_{kmp}$. Some estimates of the regression coefficients for both the dependent variables are not consistent with the expectation for the effects reported in the first column of Table 5.10. However, using $\alpha = 0.05$, all these regression coefficients seem to be not significantly different from 0 according to the p-values obtained using bootstrap approach (see the bold entries in column 8 of Table 5.13). Thus, as far as the first cluster is concerned, only the number of inhabitants per square kilometer (SD1) and the proportion of people in employment (SLIC2) regressors result to be significantly different from 0. The estimated effects of the population density on both the dependent variables are positive within both clusters detected by the model (see $\hat{\beta}_{111}$, $\hat{\beta}_{121}$, $\hat{\beta}_{211}$ and $\hat{\beta}_{221}$ in the fifth

Table 5.13: $\hat{\beta}_{kmp}$, estimated standard errors, z_{kmp} values and p-values obtained using the bootstrap.

Regr.	k	m	p	$\hat{\beta}_{kmp}$	se($\hat{\beta}_{kmp}$)	z_{kmp}	p-value
1 (Intercept)	1	1	0	−16.546	11.447	−1.445	0.148
SD1	1	1	1	0.007	0.003	2.188	0.029
SD3	1	1	2	−10.071	18.090	−0.557	**0.578**
SD4	1	1	3	1.674	2.657	0.630	0.529
SLIC1	1	1	4	14.189	14.182	1.000	0.317
HSHM1	1	1	5	0.170	0.605	0.281	**0.779**
HSHM2	1	1	6	21.949	36.627	0.599	**0.549**
HSHM6	1	1	7	−0.155	1.281	−0.121	**0.904**
HSHM7	1	1	8	−2.045	1.231	−1.662	0.097
HSHM8	1	1	9	85.630	74.353	1.152	0.249
1 (Intercept)	1	2	0	−10.399	6.646	−1.565	0.118
SD1	1	2	1	0.007	0.003	2.345	0.019
SD3	1	2	2	27.251	20.775	1.312	0.190
SD4	1	2	3	1.112	3.597	0.309	0.757
SD5	1	2	4	−6.289	3.497	−1.798	**0.072**
SLIC2	1	2	5	35.072	15.057	2.329	0.020
SLIC5	1	2	6	−14.588	12.934	−1.128	**0.259**
HSHM1	1	2	7	0.296	0.760	0.389	**0.697**
HSHM2	1	2	8	−75.651	44.154	−1.713	0.087
HSHM5	1	2	9	0.584	0.316	1.848	0.065
HSHM9	1	2	10	0.127	0.717	0.177	0.860
1 (Intercept)	2	1	0	−3.247	0.446	−7.278	0.000
SD1	2	1	1	0.0002	0.0001	3.732	0.000
SD3	2	1	2	2.348	1.094	2.147	0.032
SD4	2	1	3	0.827	0.187	4.410	0.000
SLIC1	2	1	4	0.985	0.576	1.709	0.087
HSHM1	2	1	5	−0.282	0.026	−10.979	0.000
HSHM2	2	1	6	−8.633	2.060	−4.190	0.000
HSHM6	2	1	7	−0.097	0.081	−1.191	**0.233**
HSHM7	2	1	8	−0.007	0.058	−0.129	**0.897**
HSHM8	2	1	9	4.670	4.195	1.113	0.266
1 (Intercept)	2	2	0	−5.815	0.568	−10.235	0.000
SD1	2	2	1	0.0004	0.0001	4.221	0.000
SD3	2	2	2	7.857	1.821	4.314	0.000
SD4	2	2	3	0.806	0.324	2.488	0.013
SD5	2	2	4	0.377	0.185	2.036	0.042
SLIC2	2	2	5	1.480	1.027	1.441	0.150
SLIC5	2	2	6	2.161	0.963	2.244	0.025
HSHM1	2	2	7	−0.296	0.050	−5.893	0.000
HSHM2	2	2	8	−9.378	3.124	−3.002	0.003
HSHM5	2	2	9	0.004	0.014	0.263	0.792
HSHM9	2	2	10	0.033	0.057	0.579	0.563

column of Table 5.13). Furthermore, as far as cluster 1 is concerned, it emerges also that the estimated effects of the Employment regressor ($\hat{\beta}_{125}$) are positive (and particularly strong) for the proportion of households demanding for public residential housing (RANK.21T). As far as cluster 2 is concerned, the estimates of the regression equation for the proportion of the low income households show that ISEE.20T is negatively affected by Housing Stock ($\hat{\beta}_{215}$) and Change in Housing Stock ($\hat{\beta}_{216}$) and positively affected by Change in household ($\hat{\beta}_{212}$) and Change in foreigners ($\hat{\beta}_{213}$). Similar results have been obtained with reference to the regression equation for the proportion of households demanding for public residential housing (RANK.21T) (see $\hat{\beta}_{227}$, $\hat{\beta}_{228}$, $\hat{\beta}_{222}$ and $\hat{\beta}_{223}$), from which it also emerges that Household Size ($\hat{\beta}_{224}$) and Gini Index ($\hat{\beta}_{226}$) positively affect the proportion of households demanding for public residential housing. The parameter estimates demonstrate that the analysed dataset is characterised both by heterogeneity over municipalities and by the presence of atypical observations. This latter feature seems to characterise only the second cluster of municipalities ($\hat{\alpha}_2 = 0.829$ and $\hat{\eta}_2 = 12.316$). By using the estimates of the conditional variances and covariances, it results that the estimated correlation coefficients between the two dependent variables in the two clusters of municipalities (0.925 and 0.597) are considerably different. The two clusters determined according to the highest estimated posterior probabilities of the selected model are composed of 18 and 310 municipalities, respectively. According to the rule for the intra-class distinction between typical observations and mild outliers illustrated in Section 2.2.4 of Chapter 2, the first cluster only contain typical observations. This is a consequence of the estimates $\hat{\alpha}_1 = 0.999$ and $\hat{\eta}_1 = 1.000$ (see Table 5.12). This latter result is also evident from the estimated sample residuals $\mathbf{y}_i - \hat{\boldsymbol{\mu}}_1(\mathbf{x}_i; \hat{\boldsymbol{\beta}}_1^*)$ for the 18 municipalities belonging to the first cluster (see the scatterplot on the left side of Figure 5.12). A further proof is given by the low values of the estimated distances $\hat{d}_{i1}^2$ for the municipalities of the first cluster, which are between 0.500 and 7.131. Table 5.14 reports the complete list of the 18 municipalities classified as typical in the first cluster, together with the information concerning the ADA, ATA, Mountains and NSIA classifications. Among the 310 municipalities of the second cluster, 40 have resulted to be mild outliers. Such outliers correspond to municipalities from the nine provinces that have a "no" category for the ATA classification (see Table 5.15); as far as the NSIA classification for the latter 40 municipalities is concerned, it is worth noting that none of these belong to the Poles (main centres); furthermore, 37 of these municipalities are also classified as having no ADA, with the exception of Comacchio, Copparo (FE) and San Giovanni in Persiceto (BO). The scatterplot with the estimated sample

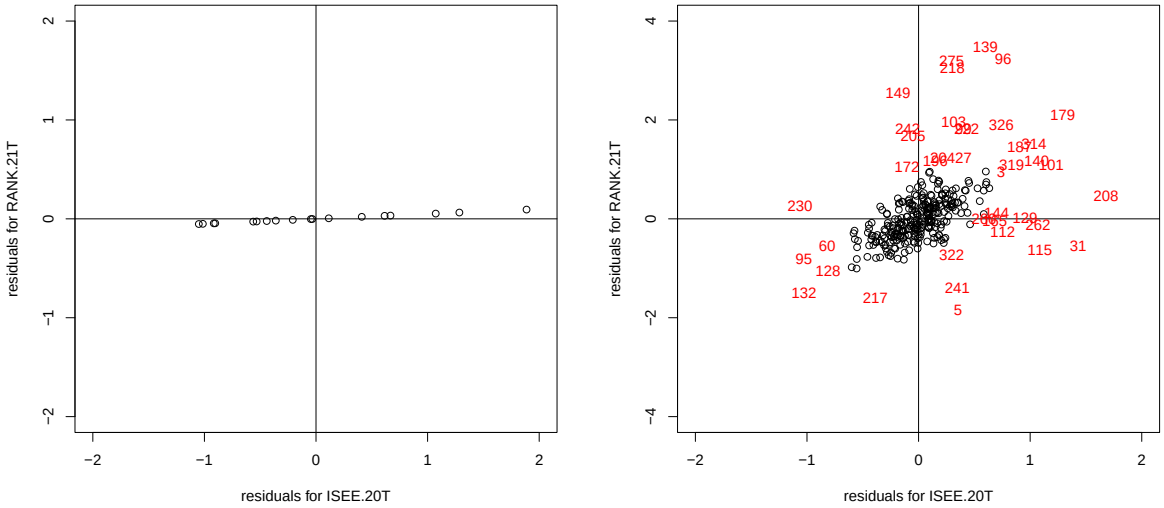

Figure 5.12: Scatterplots of the estimated residuals for the municipalities assigned to the first (left) and second (right) clusters detected by the overall best model for the analysis of housing tension in the municipalities of the ERR. Black circle correspond to typical municipalities, while outliers (red points) of the second scatterplot are labelled with the number of the corresponding municipalities.

Table 5.14: List of municipalities classified of Cluster 1 in the analysis of housing tension in the municipalities of the ERR.

Municipality	Prov.	ADA	ATA	Mountains	NSIA
Albareto	PR	No	No	Yes	P
Baricella	BO	No	No	No	I
Besenzone	PC	No	No	No	UB
Castel del Rio	BO	No	No	Yes	I
Casteldelci	RN	No	No	Yes	P
Cerignale	PC	No	No	Yes	UP
Fiorenzuola d'Arda	PC	Yes	Yes	No	UB
Gazzola	PC	No	No	No	UB
Medicina	BO	Yes	No	No	UB
Monterenzio	BO	No	No	Yes	P
Montiano	FC	No	No	No	UB
Morfasso	PC	No	No	Yes	P
San Giovanni in Maragnano	RN	No	No	No	UB
San Pietro in Cerro	PC	No	No	No	UB
Tornolo	PR	No	No	No	P
Valmozzola	PR	No	No	No	P
Varsi	PR	No	No	No	P
Zerba	PC	No	No	Yes	UP

Table 5.15: List of municipalities classified as outliers in Cluster 2 in the analysis of housing tension in the municipalities of the ERR.

ID.	Municipality	Prov.	ADA	ATA	Mountains	NSIA
3	Albinea	RE	No	No	No	UB
5	Alseno	PC	No	No	No	UB
27	Bobbio	PC	No	No	Yes	P
31	Bore	PR	No	No	Yes	P
60	Casola Valsenio	RA	No	No	Yes	I
95	Comacchio	FE	Yes	No	No	I
96	Compiano	PR	No	No	No	P
99	Copparo	FE	Yes	No	No	I
101	Corniglio	PR	No	No	No	P
103	Corte Brugnatella	PC	No	No	Yes	P
112	Farini	PC	No	No	Yes	P
115	Ferriere	PC	No	No	Yes	UP
128	Fornovo di Taro	PR	No	No	No	UB
129	Frassinoro	MO	No	No	Yes	UP
132	Galeata	FC	No	No	Yes	I
139	Goro	FE	No	No	No	P
140	Gossolengo	PC	No	No	No	UB
144	Gropparello	PC	No	No	Yes	I
149	Jolanda di Savoia	FE	No	No	No	I
165	Masi Torello	FE	No	No	No	UB
172	Mesola	FE	No	No	No	P
179	Monchio delle Corti	PR	No	No	No	UP
187	Montefiorino	MO	No	No	Yes	P
196	Mordano	BO	No	No	No	UB
204	Ostellato	FE	No	No	No	I
205	Ottone	PC	No	No	Yes	UP
208	Palanzano	PR	No	No	No	UP
217	Pievepelago	MO	No	No	Yes	UP
218	Piozzano	PC	No	No	Yes	I
222	Polesine Zibello	PR	No	No	No	UB
230	Premilcuore	FC	No	No	Yes	P
241	Riolunato	MO	No	No	Yes	UP
242	Riva del Po	FE	No	No	No	I
260	San Giorgio Piacentino	PC	No	No	No	UB
262	San Giovanni in Persiceto	BO	Yes	No	No	I
275	Sant'Agata Feltria	RN	No	No	Yes	I
314	Vernasca	PC	No	No	Yes	UB
319	Vigarano Mainarda	FE	No	No	No	UB
322	Villa Minozzo	RE	No	No	Yes	P
326	Ziano Piacentino	PC	No	No	No	I

Table 5.16: List of municipalities classified as typical in Cluster 2 in the analysis of housing tension in the municipalities of the ERR.

Municipality	Prov.	ADA	ATA	Mountains	NSIA	Municipality	Prov.	ADA	ATA	Mountains	NSIA
Agazzano	PC	No	No	No	I	Alfonsine	RA	Yes	No	No	UB
Alta Val Tidone	PC	No	No	Yes	P	Alto Reno Terme	BO	No	No	Yes	P
Anzola dell'Emilia	BO	Yes	Yes	No	UB	Argelato	BO	No	No	No	UB
Argenta	FE	Yes	No	No	UB	Bagnacavallo	RA	Yes	No	No	UB
Bagnara di Romagna	RA	No	No	No	UB	Bagno di Romagna	FC	No	No	Yes	I
Bagnolo in Piano	RE	No	No	No	UB	Baiso	RE	No	No	Yes	P
Bardi	PR	No	No	Yes	P	Bastiglia	MO	No	No	No	UB
Bedonia	PR	No	No	Yes	P	Bellaria-Igea Marina	RN	Yes	No	No	UB
Bentivoglio	BO	No	No	No	I	Berceto	PR	No	No	Yes	P
Bertinoro	FC	Yes	No	No	UB	Bettola	PC	No	No	Yes	I
Bibbiano	RE	Yes	No	No	UB	Bologna	BO	Yes	Yes	No	PO
Bomporto	MO	Yes	No	No	UB	Bondeno	FE	Yes	No	No	I
Boretto	RE	No	No	No	I	Borghi	FC	No	No	Yes	I
Borgo Tossignano	BO	No	No	Yes	UB	Borgo Val di Taro	PR	No	No	Yes	P
Borgonovo Val Tidone	PC	No	No	No	UB	Brescello	RE	No	No	No	UB
Brisighella	RA	No	No	Yes	UB	Budrio	BO	Yes	No	No	I
Busseto	PR	No	No	No	UB	Cadelbosco di Sopra	RE	Yes	No	No	UB
Cadeo	PC	No	No	No	UB	Calderara di Reno	BO	Yes	Yes	No	UB
Calendasco	PC	No	No	No	UB	Calestano	PR	No	No	Yes	I
Campagnola Emilia	RE	No	No	No	UB	Campegine	RE	No	No	No	UB
Campogalliano	MO	No	Yes	No	UB	Camposanto	MO	No	No	No	I
Camugnano	BO	No	No	Yes	P	Canossa	RE	No	No	Yes	I
Caorso	PC	No	No	No	UB	Carpaneto Piacentino	PC	No	No	No	UB
Carpi	MO	Yes	Yes	No	PO	Carpineti	RE	No	No	Yes	P
Casalecchio di Reno	BO	Yes	Yes	No	UB	Casalfiumanese	BO	No	No	Yes	UB
Casalgrande	RE	Yes	Yes	No	I	Casina	RE	No	No	Yes	P
Castel Bolognese	RA	No	No	No	UB	Castel D'Aiano	BO	No	No	Yes	P
Castel di Casio	BO	No	No	Yes	P	Castel Guelfo di Bologna	BO	No	No	No	UB
Castel Maggiore	BO	Yes	Yes	No	UB	Castel San Giovanni	PC	Yes	No	No	UB
Castel San Pietro Terme	BO	Yes	No	No	UB	Castelfranco Emilia	MO	Yes	Yes	No	I
Castellarano	RE	Yes	No	No	I	Castell'Arquato	PC	No	No	No	UB
Castello D'Argile	BO	No	No	No	I	Castelnovo di Sotto	RE	No	No	No	UB
Castelnovo Ne' Monti	RE	Yes	No	Yes	P	Castelnuovo Rangone	MO	Yes	No	No	UB
Castelvetro di Modena	MO	Yes	No	No	I	Castelvetro Piacentino	PC	No	No	No	UB
Castenaso	BO	Yes	Yes	No	UB	Castiglione dei Pepoli	BO	No	No	Yes	P
Castrocaro Terme e Terra del Sole	FC	No	No	No	UB	Cattolica	RN	Yes	Yes	No	UB
Cavezzo	MO	No	No	No	UB	Cavriago	RE	No	No	No	UB
Cento	FE	Yes	Yes	No	P	Cervia	RA	Yes	No	No	UB
Cesena	FC	Yes	Yes	No	PO	Cesenatico	FC	Yes	Yes	No	UB
Civitella di Romagna	FC	No	No	Yes	I	Codigoro	FE	Yes	No	No	P
Coli	PC	No	No	Yes	P	Collecchio	PR	Yes	No	No	UB
Colorno	PR	No	No	No	UB	Concordia sulla Secchia	MO	No	No	No	UB
Conselice	RA	No	No	No	UB	Coriano	RN	Yes	No	No	UB
Correggio	RE	Yes	Yes	No	UB	Cortemaggiore	PC	No	No	No	UB
Cotignola	RA	No	No	No	UB	Crevalcore	BO	Yes	No	No	I
Dovadola	FC	No	No	Yes	UB	Dozza	BO	No	No	No	UB
Fabbrico	RE	No	No	No	UB	Faenza	RA	Yes	Yes	No	PO
Fanano	MO	No	No	Yes	P	Felino	PR	No	No	No	UB
Ferrara	FE	Yes	Yes	No	PO	Fidenza	PR	Yes	Yes	No	PO
Finale Emilia	MO	Yes	No	No	I	Fiorano Modenese	MO	Yes	No	No	UB
Fiscaglia	FE	No	No	No	I	Fiumalbo	MO	No	No	Yes	P
Fontanelice	BO	No	No	Yes	UB	Fontanellato	PR	No	No	No	UB
Fontevivo	PR	No	No	No	UB	Forli'	FC	Yes	Yes	No	PO
Forlimpopoli	FC	Yes	No	No	UB	Formigine	MO	Yes	Yes	No	UB
Fusignano	RA	No	No	No	UB	Gaggio Montano	BO	No	No	Yes	P
Galliera	BO	No	No	No	I	Gambettola	FC	Yes	No	No	UB
Gattatico	RE	No	No	No	UB	Gatteo	FC	No	No	No	UB
Gemmano	RN	No	No	No	UB	Gragnano Trebbiense	PC	No	No	No	UB
Granarolo dell'Emilia	BO	Yes	Yes	No	UB	Grizzana Morandi	BO	No	No	Yes	P
Gualtieri	RE	No	No	No	I	Guastalla	RE	Yes	No	No	I
Guiglia	MO	No	No	Yes	P	Imola	BO	Yes	Yes	No	PO
Lagosanto	FE	No	No	No	P	Lama Mocogno	MO	No	No	Yes	P
Langhirano	PR	Yes	No	No	I	Lesignano de' Bagni	PR	No	No	No	I
Lizzano in Belvedere	BO	No	No	Yes	P	Loiano	BO	No	No	Yes	P
Longiano	FC	No	No	No	UB	Lugagnano Val d'Arda	PC	No	No	No	UB
Lugo	RA	Yes	Yes	No	PO	Luzzara	RE	No	No	No	I
Maiolo	RN	No	No	Yes	P	Malalbergo	BO	No	No	No	UB
Maranello	MO	Yes	No	No	UB	Marano sul Panaro	MO	No	No	Yes	P
Marzabotto	BO	No	No	Yes	I	Massa Lombarda	RA	Yes	No	No	UB
Medesano	PR	Yes	No	No	UB	Medolla	MO	No	No	No	I
Meldola	FC	No	No	Yes	UB	Mercato Saraceno	FC	No	No	Yes	UB
Minerbio	BO	No	No	No	I	Mirandola	MO	Yes	No	No	I
Misano Adriatico	RN	Yes	No	No	UB	Modena	MO	Yes	Yes	No	PO
Modigliana	FC	No	No	No	UB	Molinella	BO	Yes	No	No	I
Mondaino	RN	No	No	No	I	Monghidoro	BO	No	No	Yes	P
Monte San Pietro	BO	Yes	No	Yes	UB	Montecchio Emilia	RE	Yes	Yes	No	UB
Montechiarugolo	PR	Yes	No	No	UB	Montecreto	MO	No	No	Yes	P
Montefiore Conca	RN	No	No	No	I	Montegridolfo	RN	No	No	No	I
Montescudo-Monte Colombo	RN	No	No	No	UB	Montese	MO	No	No	Yes	P
Monticelli d'Ongina	PC	No	No	No	UB	Monzuno	BO	No	No	Yes	I
Morciano di Romagna	RN	No	No	No	UB	Neviano degli Arduini	PR	No	No	No	P
Noceto	PR	Yes	No	No	UB	Nonantola	MO	Yes	No	No	UB
Novafeltria	RN	No	No	Yes	P	Novellara	RE	Yes	No	No	UB
Novi di Modena	MO	Yes	No	No	UB	Ozzano dell'emilia	BO	Yes	No	No	I
Palagano	MO	No	No	Yes	P	Parma	PR	Yes	Yes	No	PO
Pavullo nel Frignano	MO	Yes	No	Yes	P	Pellegrino Parmense	PR	No	No	No	I
Pennabilli	RN	No	No	Yes	P	Piacenza	PC	Yes	Yes	No	PO
Pianello Val Tidone	PC	No	No	No	I	Pianoro	BO	Yes	Yes	Yes	I
Pieve di Cento	BO	No	No	No	I	Podenzano	PC	No	No	No	UB
Poggio Renatico	FE	No	No	No	UB	Poggio Torriana	RN	No	No	Yes	I
Polinago	MO	No	No	Yes	P	Ponte dell'Olio	PC	No	No	No	I
Pontenure	PC	No	No	No	UB	Portico e San Benedetto	FC	No	No	Yes	I
Portomaggiore	FE	Yes	No	No	I	Poviglio	RE	No	No	No	UB
Predappio	FC	No	No	Yes	UB	Prignano sulla Secchia	MO	No	No	Yes	P
Quattro Castella	RE	Yes	No	No	I	Ravarino	MO	No	No	No	I
Ravenna	RA	Yes	Yes	No	PO	Reggio nell'Emilia	RE	Yes	Yes	No	PO
Reggiolo	RE	No	No	No	UB	Riccione	RN	Yes	Yes	No	PO
Rimini	RN	Yes	Yes	No	PO	Rio Saliceto	RE	No	No	No	UB
Riolo Terme	RA	No	No	Yes	UB	Rivergaro	PC	No	No	No	UB
Rocca San Casciano	FC	No	No	Yes	I	Roccabianca	PR	No	No	No	UB
Rolo	RE	No	No	No	UB	Roncofreddo	FC	No	No	Yes	UB
Rottofreno	PC	Yes	No	No	UB	Rubiera	RE	Yes	Yes	No	UB
Russi	RA	Yes	No	No	UB	Sala Baganza	PR	No	No	No	UB
Sala Bolognese	BO	No	No	No	I	Salsomaggiore Terme	PR	Yes	No	No	UB
Saludecio	RN	No	No	No	UB	San Benedetto Val di Sambro	BO	No	No	Yes	P
San Cesario sul Panaro	MO	No	No	No	I	San Clemente	RN	No	No	No	UB
San Felice sul Panaro	MO	Yes	No	No	I	San Giorgio di Piano	BO	No	No	No	I
San Lazzaro di Savena	BO	Yes	Yes	No	UB	San Leo	RN	No	No	Yes	P
San Martino in Rio	RE	No	No	No	UB	San Mauro Pascoli	FC	Yes	No	No	UB
San Pietro in Casale	BO	Yes	No	No	I	San Polo D'Enza	RE	No	No	No	I
San Possidonio	MO	No	No	No	UB	San Prospero	MO	No	No	No	UB
San Secondo Parmense	PR	No	No	No	UB	Santa Sofia	FC	No	No	Yes	P
Sant'Agata Bolognese	BO	No	No	No	I	Sant'Agata Sul Santerno	RA	No	No	No	UB
Santarcangelo di Romagna	RN	Yes	No	No	UB	Sant'Ilario d'Enza	RE	Yes	No	No	UB
Sarmato	PC	No	No	No	UB	Sarsina	FC	No	No	Yes	UB
Sasso Marconi	BO	Yes	Yes	Yes	UB	Sassuolo	MO	Yes	Yes	No	UB
Savignano sul Panaro	MO	No	No	No	I	Savignano sul Rubicone	FC	Yes	No	No	UB
Scandiano	RE	Yes	Yes	No	I	Serramazzoni	MO	No	No	Yes	P
Sestola	MO	No	No	Yes	P	Sissa Trecasali	PR	No	No	No	UB
Sogliano al Rubicone	FC	No	No	Yes	I	Solarolo	RA	No	No	No	UB
Soliera	MO	Yes	No	No	UB	Solignano	PR	No	No	No	I
Soragna	PR	No	No	No	UB	Sorbolo Mezzani	PR	Yes	No	No	UB
Spilamberto	MO	Yes	No	No	I	Talamello	RN	No	No	Yes	P
Terenzo	PR	No	No	No	P	Terre del Reno	FE	Yes	No	No	I
Tizzano Val Parma	PR	No	No	No	P	Toano	RE	No	No	Yes	P
Torrile	PR	No	No	No	UB	Traversetolo	PR	No	No	No	I
Travo	PC	No	No	Yes	I	Tredozio	FC	No	No	Yes	I
Tresignana	FE	No	No	No	I	Valsamoggia	BO	Yes	No	Yes	I
Varano de' Melegari	PR	No	No	No	I	Ventasso	RE	No	No	Yes	UP
Vergato	BO	No	No	Yes	P	Verghereto	FC	No	No	Yes	P
Verucchio	RN	Yes	No	Yes	UB	Vetto	RE	No	No	Yes	P
Vezzano sul Crostolo	RE	No	No	No	UB	Viano	RE	No	No	Yes	I
Vignola	MO	Yes	No	No	I	Vigolzone	PC	No	No	No	UB
Villanova sull'Arda	PC	No	No	No	UB	Voghiera	FE	No	No	No	UB
Zocca	MO	No	No	Yes	P	Zola Predosa	BO	Yes	Yes	No	UB

Table 5.17: Joint classification of the 328 municipalities based on the ATA classification and the clusters determined according to the highest estimated posterior probabilities of the overall best model.

ATA	$k = 1$	$k = 2$	Row Total
No	17	272	289
	0.059	0.941	0.881
Yes	1	38	39
	0.026	0.974	0.119
Column Total	18	310	328

Table 5.18: Joint classification of the 328 municipalities based on the ADA classification and the clusters determined according to the highest estimated posterior probabilities of the overall best model.

ADA	$k = 1$	$k = 2$	Row Total
No	16	211	227
	0.070	0.930	0.692
Yes	2	99	101
	0.020	0.980	0.308
Column Total	18	310	328

Table 5.19: Joint classification of the 328 municipalities based on the Mountain classification and the clusters determined according to the highest estimated posterior probabilities of the overall best model.

Mountains	$k = 1$	$k = 2$	Row Total
No	11	215	226
	0.049	0.951	0.689
Yes	7	95	102
	0.069	0.931	0.311
Column Total	18	310	328

Table 5.20: Joint classification of the 328 municipalities based on the NSIA classification and the clusters determined according to the highest estimated posterior probabilities of the overall best model.

NSIA	$k = 1$	$k = 2$	Row Total
UB	7	144	151
	0.046	0.954	0.460
I	2	80	82
	0.024	0.976	0.250
P	7	54	61
	0.115	0.885	0.186
PO	0	16	16
	0.000	1.000	0.049
UP	2	16	18
	0.111	0.889	0.055
Column Total	18	310	328

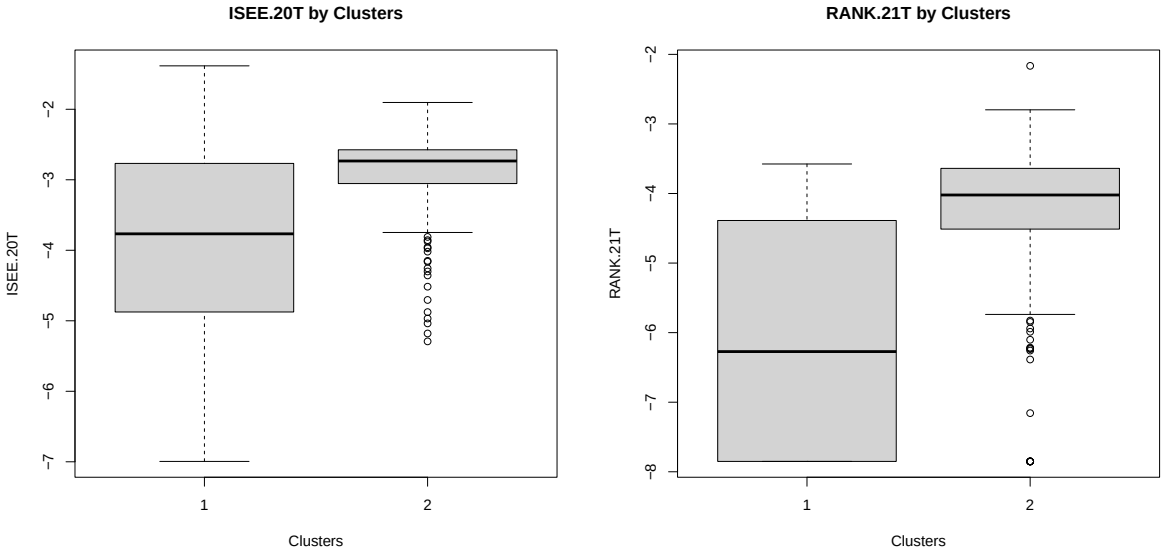

Figure 5.13: Boxplots of the two dependent variables (ISEE.20T and RANK.21T) for the clusters of municipalities determined according to the highest estimated posterior probabilities of the overall best model.

Table 5.21: Descriptive statistics of the two dependent variables (ISEE.20T and RANK.21T) for the clusters of municipalities determined according to the highest estimated posterior probabilities of the overall best model.

Dependent var.	k	Minimum	First quar.	Median	Average	Third quar.	Maximum
ISEE.20T	1	−6.995	−4.850	−3.767	−4.036	−2.783	−1.384
	2	−5.292	−3.053	−2.734	−2.870	−2.575	−1.904
RANK.21T	1	−7.851	−7.851	−6.272	−6.094	−4.481	−3.576
	2	−7.851	−4.510	−4.021	−4.231	−3.639	−2.165

residuals $\mathbf{y}_i - \hat{\boldsymbol{\mu}}_2(\mathbf{x}_i; \hat{\boldsymbol{\beta}}_2^*)$ for all the municipalities of the second cluster (see the right side of Figure 5.12) shows that, for the majority of the 40 mild outlying municipalities, the reason for the outlyingness detected by the model has been an overestimation of the proportions for either dependent variables. The values of the estimated distances $\hat{d}_{i2}^2$ for the municipalities that have been classified as typical are between 0.004 and 8.488; the minimum and maximum of the same distances for the outlying municipalities are 9.53 and 89.54, respectively. Table 5.15 reports the list of the municipalities identified as outliers in the second cluster, while those typical have been reported in Table 5.16. Tables 5.17-5.20 report the contingency tables obtained from the classifications reported in Section 5.3 and the classification determined according to the highest estimated posterior probabilities by the overall best model. The latter tables also contain the row percentages computed by dividing, for each group, the number of municipalities having one of the category for the a priori classifications (cell value) by the total number of municipalities classified in that category (cell's row total). From such tables it seems that there is no association

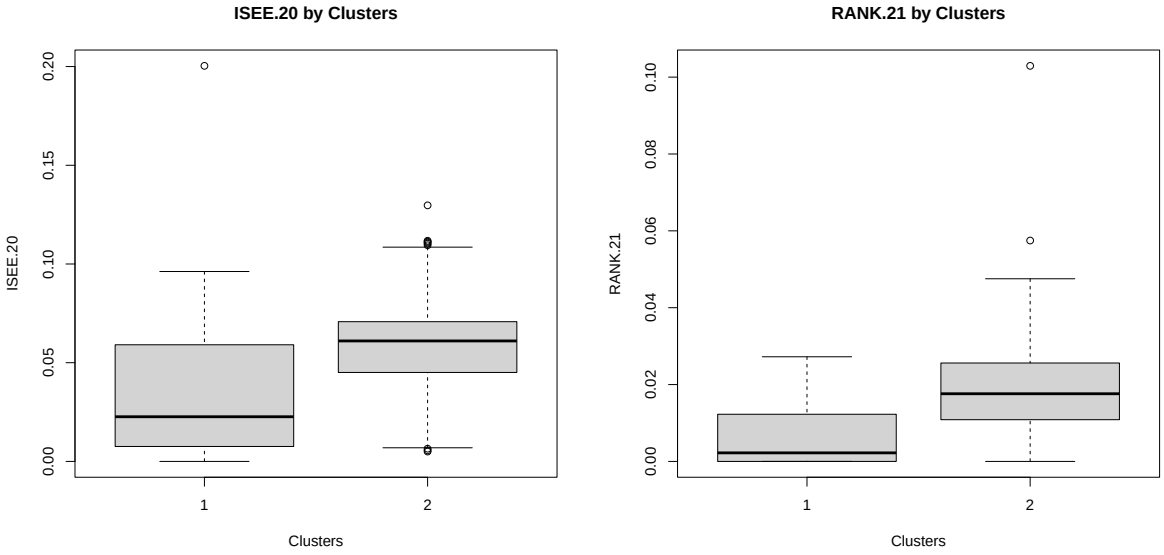

Figure 5.14: Boxplots of the ISEE.20 and RANK.21 variables (untransformed variables) for the clusters of municipalities determined according to the highest estimated posterior probabilities of the overall best model.

Table 5.22: Descriptive statistics of the ISEE.20 and RANK.21 variables (untransformed variables) for the clusters of municipalities determined according to the highest estimated posterior probabilities of the overall best model.

Untransformed dep. var.	k	Minimum	First quar.	Median	Average	Third quar.	Maximum
ISEE.20	1	0.000	0.008	0.023	0.041	0.058	0.200
	2	0.005	0.045	0.061	0.059	0.071	0.130
RANK.21	1	0.000	0.000	0.002	0.006	0.011	0.028
	2	0.000	0.011	0.018	0.018	0.026	0.103

between the partition of municipalities detected by the model and the a priori classifications determined by the institutions. This result is confirmed by the values of the adjusted Rand index (Hubert and Arabie, 1985), which are negative or close to zero for all the joint classifications just mentioned. Figure 5.13 and Table 5.21 show respectively the boxplots and some descriptive statistics of the two dependent variables for the clusters of municipalities determined according to the highest estimated posterior probabilities of the overall best model. It seems that the distributions of the two dependent variables among groups are different. Furthermore, the second cluster of municipalities shows higher median values of ISEE.20T (−2.870) and RANK.21T (−4.231). The second cluster also seems to be more homogeneous, although it also contains some outlying municipalities. Consequently, the municipalities of the second cluster result to be affected by a larger evidence of housing tension. Furthermore, Figure 5.14 and Table 5.22 show the same information for the untransformed dependent variables (ISEE.20 and RANK.21). The median values of ISEE.20 and RANK.21 in the second cluster of municipalities are respectively

equal to 0.061 and 0.018. These latter values result to be greater than that of cluster 1.

5.8 Conclusions

A study on the housing tension in the municipalities of the Emilia-Romagna region has been performed in this chapter, with the aim of helping the Regional Observatory of the housing system to better comprehend the factors that may have a strong impact on housing tension in the municipalities of the region and to detect the existence of clusters of municipalities with different levels of housing tension. To this end, a new family of seemingly unrelated clusterwise linear regression models has been developed as an extension of the model described in Chapter 2. In particular, this class allows the mixing weights to depend on some concomitant variables. The latter class, together with several types of Gaussian mixture-based linear regresion models previously proposed in the literature, has been employed to study the dependence of housing tension in the municipalities of the ERR on some indicators provided by the region. The choice of the regressors to be considered in the specification of the linear predictors of the two examined responses has been carried out through a genetic algorithm and a backward elimination technique. The overall best model has suggested the presence of two clusters of municipalities. In such a model some regressors are absent from both regression equations (SD2, SD6, SLIC3, SLIC4, HSHM3, HSHM4), while some are common to both equations (SD2, SD6, SLIC3, SLIC4, HSHM3, HSHM4). Additionally, certain regressors (SLIC1, HSHM6, HSHM7, HSHM8) are relevant in explaining the ISEE.20T but are absent in the regression equation of the second dependent variable. Conversely, SD5, SLIC2, SLIC5, HSHM5, and HSHM9 are found to be relevant for the RANK.21T dependent variable, but they do not have an effect on ISEE.20T. Therefore, using seemingly unrelated clusterwise regression models has allowed for the specification of regression equations in which the two variables which describes the housing tension in the municipalities of the ERR depend on different sets of covariates. The cluster characterized by a greater association with the housing tension also shows the presence of some outlying municipalities. Thus, the municipalities of this cluster seem to be the ones that need more public housing policies. An avenue of future research is represented by the specification of seemingly unrelated clusterwise regression models explicitly accounting for space-dependent observations.

Declarations

Funding

This study has been funded by the University of Bologna, Italy.

Competing interests

The authors have no competing interests to declare that are relevant to the content of this article.

Data availability

The data that support the findings of this study are available from Emilia-Romagna region. They have been used under license for the current study. They are not publicly available to preserve individuals' privacy under the European General Data Protection Regulation.

Code availability

The R code developed by the authors for the implementation of the ECM algorithm illustrated in Section 5.6 is available from the corresponding author upon request.

Bibliography

Akaike H (1974) A new look at the statistical model identification. IEEE Trans Autom Control 19(6): 716–723

Anscombe F J (1956) On estimating binomial response relations. Biometrika 43(3/4): 461–464

Bakken GA, Houghton TP, Kalivas JH (1999) Cyclic subspace regression with analysis of wavelength-selection criteria. Chemometr Intell Lab Syst 45: 225–239

Barca F, Casavola P, Lucatelli S (2014) Strategia nazionale per le Aree interne: Definizione, obiettivi, strumenti e governance. In Materiali Uval. Ministero dell'Economia e delle Finanze: Roma, Italia

Berkson J (1955) Maximum Likelihood and Minimum X2 Estimates of the Logistic Function. J Am Stat Assoc 50: 130–62

Bogdon AS, Can A (1997) Indicators of local housing affordability: Comparative and spatial approaches. Real Estate Economics 25(1): 43–80

Bonafede G, Napoli G (2021) Housing Affordability for Urban Regions. In Urban Regionalisation Processes (pp. 205-233). Springer, Cham

Borg I (2015) Housing deprivation in Europe: On the role of rental tenure types. Housing, Theory and Society 32(1): 73–93

Breiman L (1996) Heuristics of instability and stabilization in model selection. Ann Statist 24(6): 2350–2383

Chatterjee S, Laudato M, Lynch LA (1996) Genetic algorithms and their statistical applications: an introduction. Comput Stat Data Anal 22: 633–651

Chowdhury MZI, Turin TC (2020) Variable selection strategies and its importance in clinical prediction modelling. FMCH 8(1)

Crawford GM, Hoel PG (1972) Model selection via stepwise regression. Comm Stat - Theory Methods 1(1): 85–94

Forina M, Drava C, De La Pezuela C (1986) VI CAC (Chemometrics in Analytical Chemistry Conference), Tarragona, Abstract PII-29

Galimberti G, Manisi A, Soffritti G (2018) Modelling the role of variables in model-based cluster analysis. Stat Comp, 28(1): 145–169

Galimberti G, Soffritti G (2020) Seemingly unrelated clusterwise linear regression. Adv Data Anal Classif 14(2): 235–260

Giorgi GM, Gigliarano C (2017) The Gini concentration index: a review of the inference literature. J Econ Surv, 31(4): 1130–1148

Goldberg DE (1989) Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading

Gormley IC, Murphy TB (2011) Mixture of experts modelling with social science applications. In: Mengersen K, Robert C, Titterington DM (eds) Mixtures: estimation and applications, chapter 9. Wiley, New York, pp 101–121

Gray P, Campbell L (2001) Managing Social Housing in N. Ireland. In Housing in Northern Ireland-and comparisons with the Republic of Ireland (pp. 93-111). Chartered Institute of Housing

Hancock KE (1993). "Can pay? Won't pay?" or Economic Principles of "Affordability". Urban studies, 30(1): 127–145

Henningsen A, Hamann JD (2007) **systemfit**: a package for estimating systems of simultaneous equations in R. J Stat Softw 23(4): 1–40

Hubert L, Arabie P (1985) Comparing partitions. J Classif 2(1): 193–218

Jianqing F, Runze Liu R (1999) Variable selection via penalized likelihood. J Am Stat Assoc 94(446): 1321–1330

Johnson N (1949) Systems of frequency curves generated by methods of translation. Biometrika 36: 149–76

Jolliffe IT, Cadima J (2016) Cyclic subspace projection methods for large-scale regression problems. J R Stat Soc B 78(1): 169–190

Jones PN, McLachlan GJ (1992) Fitting finite mixture models in a regression context. Aust J Stat 34(2): 233–240

Kirkpatrick S, Gelati CD, Vechi MP (1983) Optimization by simulated annealing. Science 20: 671–680

Kutty N (1999) Determinants of structural adequacy of dwellings. J Hous Built Environ, 10(1): 27–43

Mazza A, Punzo A (2020) Mixtures of multivariate contaminated normal regression models Stat Pap 61(2): 787–822

Lindgren F, Geladi P, Rannar S, Wold S (1994) Interactive Variable Selection (IVS) for PLS. Part 1: theory and algorithms. J Chemom 8: 349–363

Mallows CL (1973) Some comments on Cp. Technometrics 15(4): 661–675

Michalewitz Z (1996) Genetic Algorithms+Data Structures=Evolution Programs, 3rd ed., Springer-Verlag, Berlin

Miller AJ (1984) Selection of subsets of regression variables. J R Stat Soc A 147(3): 389–425

Miller AJ (1991) Subset selection in regression (2nd ed.). Chapman and Hall

Ministry of Economic Development (2014) A Strategy for Inner Areas in Italy: Definition, Objectives, Tools and Governance, Materiali UVAL, Rome, available at: http://old2018.agenziacoesione.gov.it/opencms/export/sites/dps/it/documentazione/servizi/materiali_uval/Documenti/MUVAL_31_Aree_interne_ENG.pdf

Moretto V, Elia G, Schirinzi S, Vizzi R, Ghiani G (2021) A knowledge visualization approach to identify and discovery inner areas: a pilot application in the province of Lecce. Eur Plann Stud 29(9): 1819–1842

Murphy K, Murphy TB (2020) Gaussian parsimonious clustering models with covariates and a noise component. Adv Data Anal Class 14(2): 293–325

Park T (1993) Equivalence of maximum likelihood estimation and iterative two-stage estimation for seemingly unrelated regression models. Commun Stat Theory Methods 22(8): 2285–2296

Perrone G, Soffritti G (2023) Seemingly unrelated clusterwise linear regression for contaminated data. Stat Pap 64: 883–921. https://doi.org/10.1007/s00362-022-01344-6

R Core Team (2021) `R`: a language and environment for statistical computing. `R` Foundation for Statistical Computing, Vienna, Austria. http://www.R-project.org

Scrucca L (2013) `GA`: a package for genetic algorithms in R. J. Stat. Softw. 53(4): 1–37

Scrucca L (2016) Genetic algorithms for subset selection in model-based clustering. In: Celebi, M.E., Aydin, K. (eds.) Unsupervised Learning Algorithms, pp. 55–70. Springer, Berlin

Schwarz G (1978) Estimating the dimension of a model. Ann Stat 6(2):461–464

Soffritti G, Galimberti G (2011) Multivariate linear regression with non-normal errors: a solution based on mixture models. Stat Comput 21(4): 523–536

Srivastava VK, Giles DEA (1987) Seemingly unrelated regression equations models. Marcel Dekker, New York

Sutter JM, Kalivas JH (1993) Comparison of Forward selection, Backward elimination and Generalised Simulated Annealing for variable selection. Microchem J 47: 60–66

Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc B 58(1): 267–288

Townsend P (1979) Poverty in the United Kingdom. London: Allen Lane and Penguin Books

Venables WN, Ripley BD (2013) Modern applied statistics with S-PLUS. Springer. Sci Bus Media

Warton DI, Hui FK (2011) The arcsine is asinine: the analysis of proportions in ecology. Ecology 92(1): 3–10

Appendix A

R functions

This section contains the R functions which have been developed in this project and employed in the analyses reported in Chapters 2-3 (MCSUN) and in Chapter 4 (SuCNCW)[1].

A.1 MCSUN

```
#-----------------------------------------------------------------------------
# Parsimonious Mixtures of Seemingly Unrelated Contaminated Normal Regression Models
#-----------------------------------------------------------------------------
MCSUN<-function(formula.list,data=list(),k,tol=rep(10^(-8), 3), //
       iter.max=c(500,500,10),modelnameY){

library(mclust)
library(mvtnorm)
library(Matrix)
library(matrixcalc)
library(systemfit)
library(tictoc)
library(ContaminatedMixt)

start_time <- Sys.time()
tic()
```

[1]The presence of "//" at the end of the row indicates that the corresponding command continues in the following row.

```
M<-length(formula.list)

#------------------
# Regressors Matrix
#------------------

v_list<-character(0)
y_list<-character(0)
for (i in 1:M){
tf<-terms(formula.list[[i]],data=data)
v_list<-c(v_list,attr(tf,"term.labels"))
y_list<-c(y_list,all.vars(formula.list[[i]])[1])
}
P.star<-length(v_list)
v_unique<-unique(v_list)
P<-length(v_unique)

d<-matrix(NA,M,P)
colnames(d)<-v_unique
d<-as.data.frame(d)
pm<-NULL
for (m in 1:M){
tf_i<-terms(formula.list[[m]],data=data)
v_list_i<-attr(tf_i,"term.labels")
  pm[m]<-length(v_list_i)
d[m,]<-is.element(v_unique, v_list_i)*1
}
D<-d<-as.matrix(d)
D[D == 0] <- NA
colnames(D)<-v_unique
rownames(D)<-y_list
```

```
#-------------------------
# Npar by parameterisation
#-------------------------

if (modelnameY == "EII") nparcov.Y = 1
else if (modelnameY == "VII") nparcov.Y = k
else if (modelnameY == "EEI") nparcov.Y = M
else if (modelnameY == "VEI") nparcov.Y = M+k-1
else if (modelnameY == "EVI") nparcov.Y = M*k-k+1
else if (modelnameY == "VVI") nparcov.Y = M*k
else if (modelnameY == "EEE") nparcov.Y = M*(M+1)/2
else if (modelnameY == "EEV") nparcov.Y = k*M*(M+1)/2-(k-1)*M
else if (modelnameY == "VEV") nparcov.Y = k*M*(M+1)/2-(k-1)*(M-1)
else if (modelnameY == "VVV") nparcov.Y = k*M*(M+1)/2
else if (modelnameY == "EVE") nparcov.Y = M*(M+1)/2+(k-1)*(M-1)
else if (modelnameY == "VVE") nparcov.Y = M*(M+1)/2+(k-1)*M
else if (modelnameY == "VEE") nparcov.Y = M*(M+1)/2+(k-1)
else if (modelnameY == "EVV") nparcov.Y = k*M*(M+1)/2-(k-1)
else stop("modelname or covtype for the responses is not correctly defined")

#-------------------------------------------------
# Ordering the data frame and Missing Data control
#-------------------------------------------------
v_all<-c(y_list,v_unique)
data_O<-data[,v_all]

missing<-is.na(data_O)
if (sum(missing)!=0) //
    stop("Function MGlm can not deal with missing values")

#---------------------------------
```

```
# Initialization of the parameters
#---------------------------------

modello<-systemfit(formula.list,data=data_O)
residui<-residuals(modello)

 #--------------------
 # Prior Probabilities
 #--------------------

 mclust.init<-mclustBIC(residui,G=k,modelNames=modelnameY)
 if (is.na(mclust.init)==TRUE) {
   mclust.init<-mclustBIC(residui,G=k)
   mclust.init<-summary(mclust.init,residui,G=k)
 } else {
   mclust.init<-mclustBIC(residui,G=k)
   mclust.init<-summary(mclust.init,residui,G=k,modelNames=modelnameY)
 }

 pro.init<-1
 if (k>1) pro.init<-mclust.init$parameters$pro

y<-as.matrix(data_O[,1:M])
colnames(y)<-y_list
I<-length(y[,1])

Dcost<-NULL
#---------------------------------------
# Parameters of the Response distribution
#---------------------------------------
b.init.k<-matrix(0,P.star+M,k)
Sigma_init_Y.X<-array(0,c(M,M,k))
```

```
for (h in 1:k){
mod.YX.h<-try(systemfit(formula.list,data=data_O //
          [mclust.init$classification==h,]),silent=TRUE)
if (class(mod.YX.h)=="try-error") mod.YX.h<-systemfit(formula.list, //
          data=data_O[sample(x=I,size=I*pro.init[h]),])
b.init.k[,h]<-as.vector(mod.YX.h$coefficients)
Sigma_init_Y.X[,,h]<-mod.YX.h$residCov
}
rownames(b.init.k)<-names(mod.YX.h$coefficients)
dimnames(Sigma_init_Y.X)<-list(y_list,y_list)

if (modelnameY == "EII") //
  val = try(msEII(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "VII") //
  val = try(msVII(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "EEI") //
  val = try(msEEI(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "VEI") //
  val = try(msVEI(Sk=Sigma_init_Y.X, ng=pro.init, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "EVI") //
  val = try(msEVI(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "VVI") //
  val = try(msVVI(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "EEE") //
  val = try(msEEE(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "EEV") //
  val = try(msEEV(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "VEV") //
  val = try(msVEV(Sk=Sigma_init_Y.X, ng=pro.init, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VVV") //
```

```
  val = try(msVVV(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "EVE") //
  val = try(msEVE(Sk=Sigma_init_Y.X, ng=pro.init, D0=Dcost, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VVE") //
  val = try(msVVE(Sk=Sigma_init_Y.X, ng=pro.init, D0=Dcost, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VEE") //
  val = try(msVEE(Sk=Sigma_init_Y.X, ng=pro.init, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "EVV") //
  val = try(msEVV(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else stop("modelname or covtype for the responses is not correctly defined")

Sigma_init_Y.X<-val$sigma

#-------------------------
# Contamination parameters
#-------------------------

alpha<-rep(0.999,k)
eta<-rep(1.001,k)

#-------------------------------------
# Initialization of the Log Likelihood
#-------------------------------------

x.l<-list()
for (m in 1:M){
     x.l[[m]]<-lm(formula.list[[m]],data_O,x=TRUE)$x
}
```

```
Xpp<-bdiag(x.1)

priorphi.ih<-matrix(0,I,k)

h_1<-matrix(0,I,k)
h_m<-matrix(0,I,k)

for (h in 1:k){
 mu_Y.X<- Xpp%*%b.init.k
 h_1[,h]<-(alpha[h]*dmvnorm.new(y,matrix(mu_Y.X[,h],I,M), //
          as.matrix(Sigma_init_Y.X[,,h])))
 h_m[,h]<-dCN(y,matrix(mu_Y.X[,h],I,M),as.matrix(Sigma_init_Y.X[,,h]), //
          alpha[h],eta[h])
 priorphi.ih[,h]<-pro.init[h]*h_m[,h]
}

zeri<-sum(priorphi.ih==0)>0
loglik<- sum(log(apply(priorphi.ih,1,sum)))
loglik.iterECM<-loglik
iterazioni.lc2.iterECM<-0
stopECM<-FALSE
iterECM<-0

C.noninvertibile<-0
SigmaY.noninvertibile<-0

Alpha_new<-alpha
Eta_new<-eta
Beta_init<-b.init.k
Beta_new<-Beta_init
SigmaY.X_init<-Sigma_init_Y.X
SigmaY.X_new<-SigmaY.X_init
```

```
#--------------
# ECM ALGORITHM
#--------------

while (stopECM==FALSE){

 #---------
 # E-STEP
 #---------
  #Computation of the posterior probabilities

  p.ih<-priorphi.ih/apply(priorphi.ih,1,sum)
  u.ih<-h_1/h_m

 #--------
 # CM1 STEP
 #--------
  #Computation of the prior probabilities

  pro.new<-apply(p.ih,2,sum)/I
  num.ih<-p.ih*u.ih

  #Computation of alpha
  Alpha_new<-apply(num.ih,2,sum)/apply(p.ih,2,sum)
  controllo.Alpha<-rep(0.5,k)
  controllo1.Alpha<-rbind(Alpha_new,controllo.Alpha)
  Alpha_new<-apply(controllo1.Alpha,2,max)

 #-----------------------------------------------
 # Iterative computation of Beta_new e SigmaY.X_new
 #-----------------------------------------------
```

```
  iterCM<-0
  stopCM<-0

  while (stopCM==FALSE){

   #Initialization of the matrices
   C<-list()
   N<-list()
   F<-list()

   #Weights
   w.ih<-u.ih+t(t(matrix(1,I,k)-u.ih)/Eta_new)

   for (h in 1:k){
    F[[h]]<- crossprod(Xpp,suppressMessages((solve(SigmaY.X_new[,,h]) //
              %x%Matrix(diag(p.ih[,h]*w.ih[,h]),sparse=TRUE) )))
    C[[h]]<-F[[h]]%*%Xpp
    N[[h]]<-F[[h]]%*% as.vector(y)

#Is matrix C nonsingular?
    check.C<-eigen(as.matrix(C[[h]]),symmetric=TRUE)
    invertibilitaC<-check.C$values[(P.star+M)]/check.C$values[1] //
                    > 1/(10^50)

if (invertibilitaC==FALSE){
     #ECM algorithm stops
     stopCM<-TRUE
     stopECM<-TRUE
     C.noninvertibile<-1
    } else {
#Computation of Beta
     Beta_new[,h]<-solve(as.matrix(C[[h]]),tol=1/(10^50))%*% //
```

```
                    as.matrix(N[[h]])
    }
   }

   if (C.noninvertibile==0) {
   check.invertibilita.SigmaY<-rep(0,k)

for (h in 1:k){
 U<-matrix((as.vector(y)-(Xpp%*%as.matrix(Beta_new[,h]))),M,I,byrow=TRUE)%*% //
    diag(p.ih[,h]*w.ih[,h])%*%matrix((as.vector(y)- //
    (Xpp%*%as.matrix(Beta_new[,h]))),I,M)
 SigmaY.X_new[,,h]<- U/sum(p.ih[,h])
                                  }

if (modelnameY == "EII") //
 val2 = try(msEII(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "VII") //
 val2 = try(msVII(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "EEI") //
 val2 = try(msEEI(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "VEI") //
 val2 = try(msVEI(Sk=SigmaY.X_new, ng=pro.new, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "EVI") //
 val2 = try(msEVI(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "VVI") //
 val2 = try(msVVI(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "EEE") //
 val2 = try(msEEE(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "EEV") //
 val2 = try(msEEV(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "VEV") //
```

```
 val2 = try(msVEV(Sk=SigmaY.X_new, ng=pro.new, eplison= tol[3], //
        max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VVV") //
 val2 = try(msVVV(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "EVE") //
 val2 = try(msEVE(Sk=SigmaY.X_new, ng=pro.new, D0=Dcost, eplison= tol[3], //
        max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VVE") //
 val2 = try(msVVE(Sk=SigmaY.X_new, ng=pro.new, D0=Dcost, eplison= tol[3], //
        max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VEE") //
 val2 = try(msVEE(Sk=SigmaY.X_new, ng=pro.new, eplison= tol[3], //
        max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "EVV") //
 val2 = try(msEVV(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else stop("modelname or covtype for the responses is not correctly defined")

 if (length(val2)>1){
  SigmaY.X_new<-val2$sigma
   for (h in 1:k){
    if (M==1) {
      contr.1<-SigmaY.X_new[,,h]< 1/(10^10)
      contr.2<-try(solve(SigmaY.X_new[,,h]),silent=TRUE)
      contr.3<-(Eta_new[h]*SigmaY.X_new[,,h])< 1/(10^10)
      contr.4<-try(solve(Eta_new[h]*SigmaY.X_new[,,h]),silent=TRUE)
      check.invertibilita.SigmaY[h]<- (contr.1==TRUE | class(contr.2)=="try-error" | //
                                       contr.3==TRUE | class(contr.4)=="try-error")
               } else {
      check.ha<-eigen(SigmaY.X_new[,,h],symmetric=TRUE)
      check.hb<-eigen(Eta_new[h]*SigmaY.X_new[,,h],symmetric=TRUE)
      contr.1<-check.ha$values[M]/check.ha$values[1] < 1/(10^10)
      contr.2<-try(solve(SigmaY.X_new[,,h]),silent=TRUE)
```

```
      contr.3<-check.hb$values[M]/check.hb$values[1] < 1/(10^10)
      contr.4<-try(solve(Eta_new[h]*SigmaY.X_new[,,h]),silent=TRUE)
               }
      }
}

if (sum(check.invertibilita.SigmaY)!=0){
           # the ECM algorithm stops
       stopCM<-TRUE
       stopECM<-TRUE
    SigmaY.noninvertibile<-1
      } else {
       iterCM<-iterCM+1
       criterio<-sqrt(sum(( Beta_init- Beta_new)^2) + //
             sum((SigmaY.X_init-SigmaY.X_new)[upper.tri(diag(M),diag=TRUE)]^2))
       nparametri.criterio<-k*(P.star+M)+k*M*(M+1)/2
       if ((criterio/nparametri.criterio<tol[2]) || (iterCM==iter.max[2])){
          stopCM<-TRUE
         iterazioni.lc2.iterECM<-c(iterazioni.lc2.iterECM,iterCM)
      } else {
         Beta_init<- Beta_new
         SigmaY.X_init<-SigmaY.X_new
      }
}
}
        #-------------------------
}       # End of WHILE for B and SY
        #-------------------------

nume.ih<-matrix(0,I,k)
deno.ih<-p.ih*(matrix(1,I,k)-u.ih)
```

```
  #---------------------------------
  # Computation of Eta- STEP CM2
  #---------------------------------

if (C.noninvertibile==0 & SigmaY.noninvertibile==0) {
for (h in 1:k){
mu_Y.X<- Xpp%*%Beta_new
nume.ih[,h]<-deno.ih[,h]*mahalanobis.new(y,matrix(mu_Y.X[,h],I,M), //
             as.matrix(SigmaY.X_new[,,h]))
}
etanew<-apply(nume.ih,2,sum)/(M*apply(deno.ih,2,sum))
controllo.Eta<-rep(1,k)
controllo1.Eta<-rbind(etanew,controllo.Eta)
Eta_new<-apply(controllo1.Eta,2,max)
}
if (C.noninvertibile==0 & SigmaY.noninvertibile==0) {

#---------------------------------
# Updating the log-likelihood
#---------------------------------
for (h in 1:k){

mu_Y.X<- Xpp%*%Beta_new
h_1[,h]<-(Alpha_new[h]*dmvnorm.new(y,matrix(mu_Y.X[,h],I,M), //
         as.matrix(SigmaY.X_new[,,h])))
xxx<-try(dCN(y,matrix(mu_Y.X[,h],I,M),as.matrix(SigmaY.X_new[,,h]), //
         Alpha_new[h],Eta_new[h]),silent=TRUE)
if (class(xxx)=="try-error") {
xxx<-dCNmod(y,matrix(mu_Y.X[,h],I,M),as.matrix(SigmaY.X_new[,,h]), //
     Alpha_new[h],Eta_new[h])
}
h_m[,h]<-xxx
```

```
priorphi.ih[,h]<-pro.new[h]*h_m[,h]
}

zeri<-c(zeri,sum(priorphi.ih==0)>0)
loglik.new<- sum(log(apply(priorphi.ih,1,sum)))
loglik.iterECM<-c(loglik.iterECM,loglik.new)
iterECM<-iterECM+1
if (iterECM>2){
   #--------------------
   # Stopping criterion
   #--------------------
   a<-(loglik.iterECM[iterECM+1]-loglik.iterECM[(iterECM)])/ //
      (loglik.iterECM[(iterECM)] - loglik.iterECM[(iterECM-1)])
   if (a=="NaN") a<-0
   loglik.inf<-loglik.iterECM[(iterECM-1)]+(1/(1-a))* //
               (loglik.iterECM[iterECM]-loglik.iterECM[(iterECM-1)])
   if (k==1) loglik.inf<-loglik.iterECM[iterECM]
   improvement<-loglik.inf-loglik.new
   if ((abs(loglik.inf-loglik.new)<tol[1]) || //
      (iterECM==iter.max[1])) stopECM<-TRUE
   else loglik<-loglik.new
   } else {
   loglik<-loglik.new
   }
 }
   #-------------------------
}  # End of the ECM Algorithm
   #-------------------------

cat("numero tot. di iterazioni ECM = ", iterECM, "\n")

EM.ok<-C.noninvertibile+SigmaY.noninvertibile
```

```
if (EM.ok == 0) {

#-----------
#    BIC
#-----------
npar<- (3*k)-1 +  k*(P.star+M) + nparcov.Y
bic<- 2*loglik-npar*log(I)
#-----------
#    ICL
#-----------
massimo<-c()
cl<-apply(p.ih,1,which.max)

for (i in 1:I){
massimo[i]<-p.ih[i,cl[i]]
}

icl_mazza<-bic+2*sum(log(massimo))

  entropia.post<-function(z) {
  contr<-log(z)*z
  contr.ok<-contr!="NaN"
  -sum(contr[contr.ok])
  }

icl_baek<-bic-2*sum(apply(p.ih,1,entropia.post))

   #----------
   # Ordering
   #----------
pos<-sort.list(pro.new)
```

```
Alpha<-Alpha_new[pos]

Eta<-Eta_new[pos]

B<-Beta_new[,pos]

SigmaY.X<-SigmaY.X_new[,,pos]

pi<-sort(pro.new)

pih<-p.ih[,pos]

uih<-u.ih[,pos]

end_time <- Sys.time()

tempo<-toc()

total_time<-end_time - start_time

total_time2<-tempo$toc-tempo$tic


      } else {


Alpha<-NA

Eta<-NA

B<-NA

SigmaY.X<-NA

pi<-NA

pih<-NA

uih<-NA

loglik<-NA

loglik.iterEM<-NA

npar<-NA

bic<-NA

cl<-NA

icl_baek<-NA

icl_mazza<-NA

total_time<-NA

total_time2<-NA


}
```

```
out<-list(M=M,k=k,P=P,P.star=P.star, D=D,pi=pi,B=B,SigmaY.X=SigmaY.X, //
     loglik.iter=loglik.iterECM,loglik=loglik,npar=npar, BIC=bic, //
     modelnameY=modelnameY, ICL_MAZZA=icl_mazza, ICL_BAEK=icl_baek, //
     Alpha=Alpha, Eta=Eta, pih=pih, uih=uih,cl=cl,time=total_time, //
     time2=total_time2)
class(out)<-"MCSUN"
out
}
```

A.2 SuCNCW

```
#-----------------------------------------------------------------------------
# Parsimonious seemingly unrelated contaminated normal cluster-weighted models
#-----------------------------------------------------------------------------
CMW<-function(formula.list,data=list(),k,tol=rep(10^(-6),3), //
     iter.max=c(500,1,10),v_input=c(),modelnameY,modelnameX){

library(mclust)
library(mvtnorm)
library(Matrix)
library(matrixcalc)
library(systemfit)
library(tictoc)
library(ContaminatedMixt)

start_time <- Sys.time()
tic()

M<-length(formula.list)

if (M==0) stop("No response variables have been selected for the analysis")

#--------------------
# Regressors Matrix
#--------------------

v_list<-character(0)
y_list<-character(0)

for (i in 1:M){
 tf<-terms(formula.list[[i]],data=data)
```

```
 v_list<-c(v_list,attr(tf,"term.labels"))
 yi<-attr(tf,"variables")[[2]]
 class(yi)<-"character"
 y_list<-c(y_list,yi)
}

P.star<-length(v_list)
v_unique<-unique(c(v_list,v_input))
P<-length(v_unique)

if (P==0) stop("Only response variables have been selected for the analysis")

#-------------------------
# Npar by parameterisation
#-------------------------

if (modelnameX == "EII") nparcov.X <- 1
else if (modelnameX == "VII") nparcov.X <- k
else if (modelnameX == "EEI") nparcov.X <- P
else if (modelnameX == "VEI") nparcov.X <- P+k-1
else if (modelnameX == "EVI") nparcov.X <- P*k-k+1
else if (modelnameX == "VVI") nparcov.X <- P*k
else if (modelnameX == "EEE") nparcov.X <- P*(P+1)/2
else if (modelnameX == "EEV") nparcov.X <- k*P*(P+1)/2-(k-1)*P
else if (modelnameX == "VEV") nparcov.X <- k*P*(P+1)/2-(k-1)*(P-1)
else if (modelnameX == "VVV") nparcov.X <- k*P*(P+1)/2
else if (modelnameX == "EVE") nparcov.X <- P*(P+1)/2+(k-1)*(P-1)
else if (modelnameX == "VVE") nparcov.X <- P*(P+1)/2+(k-1)*P
else if (modelnameX == "VEE") nparcov.X <- P*(P+1)/2+(k-1)
else if (modelnameX == "EVV") nparcov.X <- k*P*(P+1)/2-(k-1)
else stop("modelname or covtype for the predictors is not correctly defined")
```

```
if (modelnameY == "EII") nparcov.Y <- 1
else if (modelnameY == "VII") nparcov.Y <- k
else if (modelnameY == "EEI") nparcov.Y <- M
else if (modelnameY == "VEI") nparcov.Y <- M+k-1
else if (modelnameY == "EVI") nparcov.Y <- M*k-k+1
else if (modelnameY == "VVI") nparcov.Y <- M*k
else if (modelnameY == "EEE") nparcov.Y <- M*(M+1)/2
else if (modelnameY == "EEV") nparcov.Y <- k*M*(M+1)/2-(k-1)*M
else if (modelnameY == "VEV") nparcov.Y <- k*M*(M+1)/2-(k-1)*(M-1)
else if (modelnameY == "VVV") nparcov.Y <- k*M*(M+1)/2
else if (modelnameY == "EVE") nparcov.Y <- M*(M+1)/2+(k-1)*(M-1)
else if (modelnameY == "VVE") nparcov.Y <- M*(M+1)/2+(k-1)*M
else if (modelnameY == "VEE") nparcov.Y <- M*(M+1)/2+(k-1)
else if (modelnameY == "EVV") nparcov.Y <- k*M*(M+1)/2-(k-1)
else stop("modelname or covtype for the responses is not correctly defined")

d<-matrix(NA,M,P)
colnames(d)<-v_unique
d<-as.data.frame(d)
pm<-NULL
for (m in 1:M){
tf_i<-terms(formula.list[[m]],data=data)
v_list_i<-attr(tf_i,"term.labels")
  pm[m]<-length(v_list_i)
d[m,]<-is.element(v_unique, v_list_i)*1
}
D<-d<-as.matrix(d)
D[D == 0] <- NA
colnames(D)<-v_unique
rownames(D)<-y_list

#-----------------------------------------------
```

```
# Ordering the data frame and Missing Data control
#-------------------------------------------------
v_all<-c(y_list,v_unique)
data_O<-data[,v_all]

missing<-is.na(data_O)
if (sum(missing)!=0) stop("Function CMW can not deal with missing values")

#---------------------------------
# Initialization of the parameters
#---------------------------------

 #--------------------
 # Prior Probabilities
 #--------------------
  mclust.init<-mclustBIC(residui,G=k,modelNames=c(modelnameY,modelnameX))
 if (sum(is.na(mclust.init))==2) {
   mclust.init<-mclustBIC(data_O,G=k)
   mclust.init<-summary(mclust.init,data_O,G=k)
 } else {
   mclust.init<-mclustBIC(residui,G=k)
   mclust.init<-summary(mclust.init,data_O,G=k, //
                modelNames=c(modelnameY,modelnameX))
 }

 pro.init<-1
 if (k>1) pro.init<-mclust.init$parameters$pro

 y<-as.matrix(data_O[,1:M])
 colnames(y)<-y_list
 I<-length(y[,1])
```

```
 #----------------------------------------
 # Parameters of the Regressor distribution
 #----------------------------------------

 mean_joint<-mclust.init$parameters$mean
 var_joint<-mclust.init$parameters$variance$sigma
 Dcost<-NULL

 mean_init_X<-matrix(mean_joint[(M+1):(M+P),],P,k)
 Sigma_init_X<-array(var_joint[(M+1):(M+P),(M+1):(M+P),],c(P,P,k))
 rownames(mean_init_X)<-v_unique
 dimnames(Sigma_init_X)<-list(v_unique,v_unique)

#----------------------------------------
# Parameters of the Response distribution
#----------------------------------------
b.init.k<-matrix(0,P.star+M,k)
Sigma_init_Y.X<-array(0,c(M,M,k))
for (h in 1:k){
mod.YX.h<-try(systemfit(formula.list,data=data_O //
          [mclust.init$classification==h,]),silent=TRUE)
if (class(mod.YX.h)=="try-error") mod.YX.h<-systemfit(formula.list, //
          data=data_O[sample(x=I,size=I*pro.init[h]),])
b.init.k[,h]<-as.vector(mod.YX.h$coefficients)
Sigma_init_Y.X[,,h]<-mod.YX.h$residCov
}
rownames(b.init.k)<-names(mod.YX.h$coefficients)
dimnames(Sigma_init_Y.X)<-list(y_list,y_list)



if (modelnameY == "EII") //
  val <- try(msEII(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
```

```
else if (modelnameY == "VII") //
  val <- try(msVII(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "EEI") //
  val <- try(msEEI(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "VEI") //
  val <- try(msVEI(Sk=Sigma_init_Y.X, ng=pro.init, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "EVI") //
  val <- try(msEVI(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "VVI") //
  val <- try(msVVI(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "EEE") //
  val <- try(msEEE(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "EEV") //
  val <- try(msEEV(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "VEV") //
  val <- try(msVEV(Sk=Sigma_init_Y.X, ng=pro.init, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VVV") //
 val <- try(msVVV(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else if (modelnameY == "EVE") //
 val <- try(msEVE(Sk=Sigma_init_Y.X, ng=pro.init, D0=Dcost, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VVE") //
 val <- try(msVVE(Sk=Sigma_init_Y.X, ng=pro.init, D0=Dcost, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VEE") //
 val <- try(msVEE(Sk=Sigma_init_Y.X, ng=pro.init, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "EVV") //
 val <- try(msEVV(Sk=Sigma_init_Y.X, ng=pro.init),silent=TRUE)
else stop("modelname or covtype for the responses is not correctly defined")
```

```
   Sigma_init_Y.X<-val$sigma

if (modelnameX == "EII") //
 val <- try(msEII(Sk=Sigma_init_X, ng=pro.init),silent=TRUE)
else if (modelnameX == "VII") //
 val <- try(msVII(Sk=Sigma_init_X, ng=pro.init),silent=TRUE)
else if (modelnameX == "EEI") //
 val <- try(msEEI(Sk=Sigma_init_X, ng=pro.init),silent=TRUE)
else if (modelnameX == "VEI") //
 val <- try(msVEI(Sk=Sigma_init_X, ng=pro.init, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "EVI") //
 val <- try(msEVI(Sk=Sigma_init_X, ng=pro.init),silent=TRUE)
else if (modelnameX == "VVI") //
 val <- try(msVVI(Sk=Sigma_init_X, ng=pro.init),silent=TRUE)
else if (modelnameX == "EEE") //
 val <- try(msEEE(Sk=Sigma_init_X, ng=pro.init),silent=TRUE)
else if (modelnameX == "EEV") //
 val <- try(msEEV(Sk=Sigma_init_X, ng=pro.init),silent=TRUE)
else if (modelnameX == "VEV") //
 val <- try(msVEV(Sk=Sigma_init_X, ng=pro.init, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "VVV") //
 val <- try(msVVV(Sk=Sigma_init_X, ng=pro.init),silent=TRUE)
else if (modelnameX == "EVE") //
 val <- try(msEVE(Sk=Sigma_init_X, ng=pro.init, D0=Dcost, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "VVE") //
 val <- try(msVVE(Sk=Sigma_init_X, ng=pro.init, D0=Dcost, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "VEE") //
```

```
 val <- try(msVEE(Sk=Sigma_init_X, ng=pro.init, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "EVV") //
 val <- try(msEVV(Sk=Sigma_init_X, ng=pro.init),silent=TRUE)
else stop("modelname or covtype for the predictors is not correctly defined")

Sigma_init_X<-val$sigma

#-------------------------
# Contamination parameters
#-------------------------

alpha<-rep(0.999,k)
eta<-rep(1.001,k)
alpha_x<-rep(0.999,k)
eta_x<-rep(1.001,k)

#-----------------------------------------------
# Initialization of the Log Likelihood
#-----------------------------------------------
y<-as.matrix(data_O[,1:M])
colnames(y)<-y_list
x<-as.matrix(data_O[,(M+1):(M+P)])
colnames(x)<-v_unique
I<-length(y[,1])

int<-rep(1,length.out=I)
x.l<-list()



for (m in 1:M){
    if (pm[m]==0) {
```

```
      x.l[[m]]<-int
           } else {
      x.sel<-t(t(x)*D[m,])
  x.l[[m]]<-cbind(int,x.sel[, !apply(is.na(x.sel), 2, all)])
      }
}

Xpp<-bdiag(x.l)

h_1<-matrix(0,I,k)
h_m<-matrix(0,I,k)

h_2<-matrix(0,I,k)
h_p<-matrix(0,I,k)

priorphi.ih<-matrix(0,I,k)

mu_Y.X<- Xpp%*%b.init.k


for (h in 1:k){
h_1[,h]<-(alpha[h]*dmvnorm.new(y,matrix(mu_Y.X[,h],I,M), //
          as.matrix(Sigma_init_Y.X[,,h])))
h_2[,h]<-(alpha_x[h]*dmvnorm.new(x,matrix(mean_init_X[,h],I,P,byrow=TRUE), //
          sigma=as.matrix(Sigma_init_X[,,h])))
h_m[,h]<-dCN(y,matrix(mu_Y.X[,h],I,M),as.matrix(Sigma_init_Y.X[,,h]), //
          alpha[h],eta[h])
h_p[,h]<-dCN(x,matrix(mean_init_X[,h],I,P,byrow=T),as.matrix(Sigma_init_X[,,h]), //
             alpha_x[h],eta_x[h])
priorphi.ih[,h]<-pro.init[h]*h_m[,h]* h_p[,h]
}
```

```
zeri<-sum(priorphi.ih==0)>0
loglik<- sum(log(apply(priorphi.ih,1,sum)))
loglik.iterECM<-loglik
iterazioni.lc2.iterECM<-0
stopECM<-FALSE
iterECM<-0

Alpha_new_x<-alpha_x
Eta_new_x<-eta_x
C.noninvertibile<-0
SigmaY.noninvertibile<-0
Alpha_new<-alpha
Eta_new<-eta
Beta_init<-b.init.k
Beta_new<-Beta_init
SigmaY.X_init<-Sigma_init_Y.X
SigmaY.X_new<-SigmaY.X_init
SigmaX.noninvertibile<-0
muX_new<-mean_init_X
SigmaX_new<-Sigma_init_X

#--------------
# ECM ALGORITHM
#--------------
while (stopECM==FALSE){

#---------
# E-STEP
#---------
#computation of the posterior probabilities

p.ih<-priorphi.ih/apply(priorphi.ih,1,sum)
```

```
u.ih<-h_1/h_m
v.ih<-h_2/h_p

#---------
# CM1 STEP
#---------
    #Computation of the prior probabilities
    pro.new<-apply(p.ih,2,sum)/I

    num.ih.x<-p.ih*v.ih
    Alpha_new_x<-apply(num.ih.x,2,sum)/apply(p.ih,2,sum)
    controllo.Alpha.x<-rep(0.5,k)
    controllo1.Alpha.x<-rbind(Alpha_new_x,controllo.Alpha.x)
    Alpha_new_x<-apply(controllo1.Alpha.x,2,max)

    q.ih<-v.ih+t(t(matrix(1,I,k)-v.ih)/Eta_new_x)

    #Computation of muX
    for(h in 1:k){
      u<-sweep(x,1,p.ih[,h]*q.ih[,h],"*")
      muX_new[,h]<-apply(u,2,sum)/sum(p.ih[,h]*q.ih[,h])
    }

    #Computation of SigmaX
    for(h in 1:k){
      uu<-matrix((t(x)-muX_new[,h])%*%diag(p.ih[,h]*q.ih[,h]) //
          %*%t(t(x)-muX_new[,h]),P,P)
      rownames(uu)<-v_unique
      colnames(uu)<-v_unique
      SigmaX_new[,,h]<- matrix(uu/sum(p.ih[,h]),P,P)
    }
```

```
if (modelnameX == "EII") //
  val <- try(msEII(Sk=SigmaX_new, ng=pro.new),silent=TRUE)
else if (modelnameX == "VII") //
  val <- try(msVII(Sk=SigmaX_new, ng=pro.new),silent=TRUE)
else if (modelnameX == "EEI") //
  val <- try(msEEI(Sk=SigmaX_new, ng=pro.new),silent=TRUE)
else if (modelnameX == "VEI") //
  val <- try(msVEI(Sk=SigmaX_new, ng=pro.new, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "EVI") //
  val <- try(msEVI(Sk=SigmaX_new, ng=pro.new),silent=TRUE)
else if (modelnameX == "VVI") //
  val <- try(msVVI(Sk=SigmaX_new, ng=pro.new),silent=TRUE)
else if (modelnameX == "EEE") //
  val <- try(msEEE(Sk=SigmaX_new, ng=pro.new),silent=TRUE)
else if (modelnameX == "EEV") //
  val <- try(msEEV(Sk=SigmaX_new, ng=pro.new),silent=TRUE)
else if (modelnameX == "VEV") //
  val <- try(msVEV(Sk=SigmaX_new, ng=pro.new, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "VVV") //
  val <- try(msVVV(Sk=SigmaX_new, ng=pro.new),silent=TRUE)
else if (modelnameX == "EVE") //
  val <- try(msEVE(Sk=SigmaX_new, ng=pro.new, D0=Dcost, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "VVE") //
  val <- try(msVVE(Sk=SigmaX_new, ng=pro.new, D0=Dcost, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "VEE") //
  val <- try(msVEE(Sk=SigmaX_new, ng=pro.new, eplison= tol[3], //
  max.iter=iter.max[3]),silent=TRUE)
else if (modelnameX == "EVV") //
```

```
  val <- try(msEVV(Sk=SigmaX_new, ng=pro.new),silent=TRUE)
else stop("modelname or covtype for the predictors is not correctly defined")


check.invertibilita.X<-rep(0,k)


if (length(val)>1){
SigmaX_new<-val$sigma
for (h in 1:k){
if (P==1) {
 contr.X1<-SigmaX_new[,,h]< 1/(10^10)
 contr.X2<-try(solve(SigmaX_new[,,h]),silent=TRUE)
 contr.X3<-(Eta_new_x[h]*SigmaX_new[,,h])< 1/(10^10)
 contr.X4<-try(solve(Eta_new_x[h]*SigmaX_new[,,h]),silent=TRUE)
 check.invertibilita.X[h]<- (contr.X1==TRUE | class(contr.X2)=="try-error" | //
                             contr.X3==TRUE | class(contr.X4)=="try-error")
          } else {
 check.Xha<-eigen(SigmaX_new[,,h],symmetric=TRUE)
 check.Xhb<-eigen(Eta_new_x[h]*SigmaX_new[,,h],symmetric=TRUE)
 contr.X1<-sum(check.Xha$values[P]/check.Xha$values[1] < 1/(10^10)) //
       +sum(check.Xha$values<10^(-20))
 contr.X2<-try(solve(SigmaX_new[,,h]),silent=TRUE)
 contr.X3<-sum(check.Xhb$values[P]/check.Xhb$values[1] < 1/(10^10)) //
          +sum(check.Xhb$values<10^(-20))
 contr.X4<-try(solve(Eta_new_x[h]*SigmaX_new[,,h]),silent=TRUE)
           check.invertibilita.X[h]<-(contr.X1==1 | class(contr.X2)[1]=="try-error" | //
 class(contr.X2)[2]=="try-error" | contr.X3==1 | //
 class(contr.X4)[1]=="try-error" | class(contr.X4)[2]=="try-error")
         }
   }
 }

if (sum(check.invertibilita.X)!=0){
```

```
stopCM<-TRUE
stopECM<-TRUE
SigmaX.noninvertibile<-1
} else {

     num.ih<-p.ih*u.ih

#aggiornamento di alpha
Alpha_new<-apply(num.ih,2,sum)/apply(p.ih,2,sum)
controllo.Alpha<-rep(0.5,k)
controllo1.Alpha<-rbind(Alpha_new,controllo.Alpha)
Alpha_new<-apply(controllo1.Alpha,2,max)

#-------------------------------------------------
# iterative computation of Beta_new e SigmaY.X_new
#-------------------------------------------------
iterCM<-0
stopCM<-0

      while (stopCM==FALSE){


     # Initialization of the matrices
       C<-list()
       N<-list()
       F<-list()

#Weights
w.ih<-u.ih+t(t(matrix(1,I,k)-u.ih)/Eta_new)

for (h in 1:k){
F[[h]]<- crossprod(Xpp,suppressMessages((solve(SigmaY.X_new[,,h]) //
```

```
         %x%Matrix(diag(p.ih[,h]*w.ih[,h]),sparse=TRUE) ))
C[[h]]<-F[[h]]%*%Xpp
N[[h]]<-F[[h]]%*% as.vector(y)
# is matrix C nonsingular?
check.C<-eigen(as.matrix(C[[h]]),symmetric=TRUE)
invertibilitaC<-check.C$values[(P.star+M)]/check.C$values[1] //
                > 1/(10^50)

if (invertibilitaC==FALSE){
       # ECM algorithm stops
        stopCM<-TRUE
        stopECM<-TRUE
        C.noninvertibile<-1
       } else {
# computation of Beta
Beta_new[,h]<-solve(as.matrix(C[[h]]),tol=1/(10^50)) //
              %*% as.matrix(N[[h]])
}
}

if (C.noninvertibile==0) {

check.invertibilita.SigmaY<-rep(0,k)

for (h in 1:k){
U<-matrix((as.vector(y)-(Xpp%*%as.matrix(Beta_new[,h]))),M,I,byrow=TRUE) //
   %*% diag(p.ih[,h]*w.ih[,h])%*%matrix((as.vector(y)- //
   (Xpp%*%as.matrix(Beta_new[,h]))),I,M)
SigmaY.X_new[,,h]<- U/sum(p.ih[,h])
}

if (modelnameY == "EII") //
```

```
 val2 <- try(msEII(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "VII") //
 val2 <- try(msVII(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "EEI") //
 val2 <- try(msEEI(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "VEI") //
 val2 <- try(msVEI(Sk=SigmaY.X_new, ng=pro.new, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "EVI") //
 val2 <- try(msEVI(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "VVI") //
 val2 <- try(msVVI(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "EEE") //
 val2 <- try(msEEE(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "EEV") //
 val2 <- try(msEEV(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "VEV") //
 val2 <- try(msVEV(Sk=SigmaY.X_new, ng=pro.new, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VVV") //
 val2 <- try(msVVV(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
else if (modelnameY == "EVE") //
 val2 <- try(msEVE(Sk=SigmaY.X_new, ng=pro.new, D0=Dcost, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VVE") //
 val2 <- try(msVVE(Sk=SigmaY.X_new, ng=pro.new, D0=Dcost, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "VEE") //
 val2 <- try(msVEE(Sk=SigmaY.X_new, ng=pro.new, eplison= tol[3], //
 max.iter=iter.max[3]),silent=TRUE)
else if (modelnameY == "EVV") //
 val2 <- try(msEVV(Sk=SigmaY.X_new, ng=pro.new),silent=TRUE)
```

```
else stop("modelname or covtype for the responses is not correctly defined")


if (length(val2)>1){
SigmaY.X_new<-val2$sigma
for (h in 1:k){
if (M==1) {
contr.1<-SigmaY.X_new[,,h]< 1/(10^10)
contr.2<-try(solve(SigmaY.X_new[,,h]),silent=TRUE)
contr.3<-(Eta_new[h]*SigmaY.X_new[,,h])< 1/(10^10)
contr.4<-try(solve(Eta_new[h]*SigmaY.X_new[,,h]),silent=TRUE)
check.invertibilita.SigmaY[h]<- (contr.1==TRUE | class(contr.2)=="try-error" | //
    contr.3==TRUE | class(contr.4)=="try-error")
        } else {
check.ha<-eigen(SigmaY.X_new[,,h],symmetric=TRUE)
check.hb<-eigen(Eta_new[h]*SigmaY.X_new[,,h],symmetric=TRUE)
contr.1<-sum(check.ha$values[M]/check.ha$values[1] < 1/(10^10)) //
         +sum(check.ha$values<10^(-20))
contr.2<-try(solve(SigmaY.X_new[,,h]),silent=TRUE)
contr.3<-sum(check.hb$values[M]/check.hb$values[1] < 1/(10^10)) //
     +sum(check.hb$values<10^(-20))
contr.4<-try(solve(Eta_new[h]*SigmaY.X_new[,,h]),silent=TRUE)
         check.invertibilita.SigmaY[h]<-(contr.1==1 | class(contr.2)[1]=="try-error"| //
class(contr.2)[2]=="try-error" | contr.3==1 | //
class(contr.4)[1]=="try-error" | class(contr.4)[2]=="try-error")
        }
      }

}

if (sum(check.invertibilita.SigmaY)!=0){
       # the ECM algorithm stops
       stopCM<-TRUE
```

```
       stopECM<-TRUE
SigmaY.noninvertibile<-1
      } else {
       iterCM<-iterCM+1
       criterio<-sqrt(sum(( Beta_init- Beta_new)^2) + //
                 sum((SigmaY.X_init-SigmaY.X_new)[upper.tri(diag(M),diag=TRUE)]^2))
       nparametri.criterio<-k*(P.star+M)+k*M*(M+1)/2

       if ((criterio/nparametri.criterio<tol[2]) || (iterCM==iter.max[2])){
        stopCM<-TRUE
       iterazioni.lc2.iterECM<-c(iterazioni.lc2.iterECM,iterCM)
      } else {
       Beta_init<- Beta_new
       SigmaY.X_init<-SigmaY.X_new
      }
}
}
}
    #-------------------------
}   #END OF WHILE FOR B AND SY
    #-------------------------

nume.ih<-matrix(0,I,k)
nume.ih.x<-matrix(0,I,k)
deno.ih<-p.ih*(matrix(1,I,k)-u.ih)
deno.ih.x<-p.ih*(matrix(1,I,k)-v.ih)

  #---------------------------------
  #Computation of Eta- STEP CM2
  #---------------------------------

if (C.noninvertibile==0 & SigmaY.noninvertibile==0 & SigmaX.noninvertibile==0) {
```

```
  SigmaX.noninvertibile==0
  mu_Y.X<- Xpp%*%Beta_new

for (h in 1:k){
 nume.ih[,h]<-deno.ih[,h]*mahalanobis.new(y,matrix(mu_Y.X[,h],I,M), //
  as.matrix(SigmaY.X_new[,,h]))
 nume.ih.x[,h]<-deno.ih.x[,h]*mahalanobis.new(x,matrix(muX_new[,h],I,P,byrow=T), //
                as.matrix(SigmaX_new[,,h]))

 etanew.x<-apply(nume.ih.x,2,sum)/(P*apply(deno.ih.x,2,sum))
 controllo.Eta.x<-rep(1.001,k)
 controllo1.Eta.x<-rbind(etanew.x,controllo.Eta.x)
 Eta_new_x<-apply(controllo1.Eta.x,2,max)

 etanew<-apply(nume.ih,2,sum)/(M*apply(deno.ih,2,sum))
 controllo.Eta<-rep(1.001,k)
 controllo1.Eta<-rbind(etanew,controllo.Eta)
 Eta_new<-apply(controllo1.Eta,2,max)
}

if (C.noninvertibile==0 & SigmaY.noninvertibile==0 & SigmaX.noninvertibile==0) {


#---------------------------------
# Updating the log-likelihood
#---------------------------------
mu_Y.X<- Xpp%*%Beta_new

 for (h in 1:k){

  h_1[,h]<-(Alpha_new[h]*dmvnorm.new(y,matrix(mu_Y.X[,h],I,M), //
   as.matrix(SigmaY.X_new[,,h])))
```

```
  h_2[,h]<-(Alpha_new_x[h]*dmvnorm.new(x,matrix(muX_new[,h],I,P,byrow=T), //
            as.matrix(SigmaX_new[,,h])))
  qqq<-try(dCN(x,matrix(muX_new[,h],I,P,byrow=T),as.matrix(SigmaX_new[,,h]), //
       Alpha_new_x[h],Eta_new_x[h]),silent=TRUE)

 if (class(qqq)=="try-error" ) {
  qqq<-dCNmod(x,matrix(muX_new[,h],I,P,byrow=T),as.matrix(SigmaX_new[,,h]), //
       Alpha_new_x[h],Eta_new_x[h])
}

  xxx<-try(dCN(y,matrix(mu_Y.X[,h],I,M),as.matrix(SigmaY.X_new[,,h]), //
       Alpha_new[h],Eta_new[h]),silent=TRUE)

 if (class(xxx)=="try-error") {
  xxx<-dCNmod(y,matrix(mu_Y.X[,h],I,M),as.matrix(SigmaY.X_new[,,h]), //
   Alpha_new[h],Eta_new[h])
}

 h_p[,h]<-qqq
 h_m[,h]<-xxx
 priorphi.ih[,h]<-pro.new[h]*h_m[,h]*h_p[,h]
}

zeri<-c(zeri,sum(priorphi.ih==0)>0)
    loglik.new<- sum(log(apply(priorphi.ih,1,sum)))
loglik.iterECM<-c(loglik.iterECM,loglik.new)
iterECM<-iterECM+1
if (iterECM>2){
  #--------------------
  # stopping criterion
  #--------------------
```

```
  a<-(loglik.iterECM[iterECM+1]-loglik.iterECM[(iterECM)])/ //
      (loglik.iterECM[(iterECM)]- loglik.iterECM[(iterECM-1)])
  if (a=="NaN") a<-0
  loglik.inf<-loglik.iterECM[(iterECM-1)]+(1/(1-a))*(loglik.iterECM[iterECM]- //
              loglik.iterECM[(iterECM-1)])
  loglik.inf.22luglio<-loglik.iterECM[(iterECM)]+(1/(1-a)) //
                       *(loglik.iterECM[iterECM+1]-loglik.iterECM[(iterECM)])

  improvement3<-loglik.inf.22luglio-loglik.new
  if (k==1) loglik.inf<-loglik.iterECM[iterECM]
  improvement<-loglik.inf-loglik.new
  improvement2<-loglik.inf-loglik.iterECM[iterECM]

  if ((((loglik.inf-loglik.new)<tol[1]) & (loglik.inf-loglik.new)>0) //
  || (iterECM==iter.max[1])) stopECM<-TRUE  else loglik<-loglik.new
  } else {
  loglik<-loglik.new
  }
}
   #-------------------------
}  # End of the ECM Algorithm
   #-------------------------

EM.ok<-C.noninvertibile+SigmaY.noninvertibile+SigmaX.noninvertibile

EM.ok1<-sum(apply(SigmaY.X_new,3, function(x) eigen(x)$values)<10^(-20))+ //
        sum(apply(SigmaX_new,3, function(x) eigen(x)$values)<10^(-20))
EM.ok<-EM.ok+EM.ok1

if (EM.ok == 0) {

dimnames(SigmaX_new)<-list(v_unique,v_unique)
```

```
dimnames(SigmaY.X_new)<-list(y_list,y_list)
#-----------
#    BIC
#-----------
npar<- (5*k)-1 + (k*P) + nparcov.X + k*(P.star+M) + nparcov.Y
bic<- 2*loglik-npar*log(I)
#-----------
#    ICL
#-----------
massimo<-c()

#ICL1
cl<-apply(p.ih,1,which.max)

for (i in 1:I){
massimo[i]<-p.ih[i,cl[i]]
}

icl_mazza<-bic+2*sum(log(massimo))



#ICL2

entropia.post<-function(z) {
contr<-log(z)*z
contr.ok<-contr!="NaN"
-sum(contr[contr.ok])
}

icl_baek<-bic-2*sum(apply(p.ih,1,entropia.post))

#---------
```

```
# Ordering
#---------
pos<-sort.list(pro.new)

AlphaX<-Alpha_new_x[pos]
EtaX<-Eta_new_x[pos]
muX<-muX_new[,pos]
SigmaX<-SigmaX_new[,,pos]
Alpha<-Alpha_new[pos]
Eta<-Eta_new[pos]
B<-Beta_new[,pos]
SigmaY.X<-SigmaY.X_new[,,pos]
pi<-sort(pro.new)
end_time <- Sys.time()
tempo<-toc()
total_time<-end_time - start_time
total_time2<-tempo$toc-tempo$tic



cl<-rep(1,I)
pih<-priorphi.ih/apply(priorphi.ih,1,sum)

if (k>1) {
p.ih<-pih[,pos]
cl<-apply(pih,1,which.max)
u.ih<-u.ih[,pos]
v.ih<-v.ih[,pos]
h1.ih<-h_1[,pos]
hm.ih<-h_m[,pos]
}
} else {
```

```
Alpha<-NA
Eta<-NA
muX<-NA
SigmaX<-NA
AlphaX<-NA
EtaX<-NA
B<-NA
SigmaY.X<-NA
pi<-NA
p.ih<-NA
u.ih<-NA
v.ih<-NA
loglik<-NA
loglik.iterEM<-NA
npar<-NA
bic<-NA
cl<-NA
icl_baek<-NA
icl_mazza<-NA
total_time<-NA
total_time2<-NA
}

out<-list(M=M,k=k,P=P,P.star=P.star,h1.ih=h1.ih,hm.ih=hm.ih,muX=muX, //
     SigmaX=SigmaX,D=D,pi=pi,B=B,loglik.inf=loglik.inf,loglik.new=loglik.new, //
     SigmaY.X=SigmaY.X,loglik.iter=loglik.iterECM,loglik=loglik,npar=npar, //
     BIC=bic, modelnameX=modelnameX, modelnameY=modelnameY, //
     ICL_MAZZA=icl_mazza, ICL_BAEK=icl_baek, Alpha=Alpha, Eta=Eta, //
     AlphaX=AlphaX, EtaX=EtaX,C.noninvertibile=C.noninvertibile,SigmaX.noninvertibile= //
     SigmaX.noninvertibile,SigmaY.noninvertibile=SigmaY.noninvertibile, //
     v_unique=v_unique, p.ih=p.ih, u.ih=u.ih,v.ih=v.ih,cl=cl, //
     time=total_time,time2=total_time2)
```

```
class(out)<-"SuCNCW"

out

}
```