TECHNISCHE
UNIVERSITÄT
WIEN

VIENNA
UNIVERSITY OF
TECHNOLOGY

DISSERTATION

Identification and Estimation of Finite Mixture Models

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines Doktors der technischen Wissenschaften unter der Leitung von

Univ.-Prof. Dipl.-Ing. Dr.techn. Friedrich Leisch
Institut für Statistik
Ludwig-Maximilians-Universität München

eingereicht an der Technischen Universität Wien
bei der Fakultät für Mathematik und Geoinformation

von

Dipl.-Ing. Bettina Grün
Matrikelnummer: 9725933
Hauptstraße 39
2244 Spannberg

Wien, im September 2006

Kurzfassung

Diese Dissertation beschäftigt sich mit verschiedenen Aspekten bei der Modellierung von finiten Mischmodellen wie Modellidentifizierbarkeit, Modelldiagnose und Software-Implementierung. Das Hauptaugenmerk liegt bei finiten Mischungen von generalisierten linearen Regressionsmodellen. Die Popularität dieser Modelle ist in den letzten Jahrzehnten enorm gestiegen, da die Schätzung durch die heutzutage stark gestiegene Rechenleistung bei Computern erleichtert wurde. Verschiedene Varianten existieren wie Modelle mit zufälligen Effekten für den Achsenabschnitt oder Modelle mit begleitenden Variablen, um die Gewichte der Komponenten zu charakterisieren.

Hinreichende Bedingungen für die Identifizierbarkeit werden gegeben, wobei finite Mischungen von multinomialen logit Modellen den interessantesten Spezialfall darstellen. Finite Mischungen von multinomialen Verteilungen sind nämlich im Gegensatz zu anderen komponentenspezifischen Verteilungen wie die Normal-, Poisson- oder Gammaverteilung nicht generisch identifizierbar.

Die Verwendung von Resampling-Methoden zur Modelldiagnose im Rahmen von frequentistischer Maximum Likelihood Schätzung wird diskutiert, und verschiedene mögliche Verwendungszwecke werden unterschieden. Die Anwendung wird an mehreren Beispielen veranschaulicht. Dieser Ansatz erweitert bzw. vereinigt frühere Anwendungen der Bootstrap-Methode (Münchhausen-Methode) zur Modelldiagnose. Mögliche Wege, das Label-Switching Problem zu lösen, das in diesem Zusammenhang ähnlich wie bei der Bayesianischen MCMC Schätzung auftritt, werden diskutiert.

Die Implementierung im R Paket **flexmix** wird beschrieben, indem die Grundprinzipien des Designs skizziert und Details der Implementierung diskutiert werden. Diese Details zu kennen ist notwendig für das Schreiben neuer Modelltreiber für die komponentenspezifischen Modelle und die begleitenden Variablenmodelle. Die Verwendung des Pakets wird an mehreren Beispielen mit künstlichen und echten Daten veranschaulicht. Zusätzlich werden auch Beispiele für das Schreiben neuer Modelltreiber gegeben.

Abstract

This thesis covers different aspects in finite mixture modelling such as model identifiability, model diagnostics and software implementation. The focus is on finite mixtures of generalized linear regression models. The popularity of these models has tremendously increased in the last decades as estimation was facilitated given the nowadays easily available computing power. Different variants exist such as random intercepts models or models including concomitant variables for characterizing the component weights.

Sufficient conditions for identifiability are given where finite mixtures of multinomial logit models are the most interesting special case. This is due to the fact that finite mixtures of multinomial distributions are not generically identifiable in contrast to other component specific distributions such as Gaussian, Poisson or gamma.

The use of resampling methods for model diagnostics in a frequentist maximum likelihood setting is discussed and different possible purposes distinguished. The application is illustrated on several examples. This approach extends or unifies previous applications of the bootstrap for model diagnostics. Possible ways to solve the label switching problem which also occurs in this setting similar as in Bayesian MCMC estimation are discussed.

The implementation in the R package **flexmix** is described outlining the design principles and discussing implementational details. To know these details is necessary for writing new drivers for the component specific models and concomitant variable models. The usage of the package is illustrated on several examples for artificial data and real world data sets. In addition, examples for writing new model drivers are given.

Danksagung

Während meiner Arbeit an dieser Dissertation wurde ich finanziell unterstützt (in chronologischer Reihenfolge) von U.P.P.E.R Network GmbH und dem Forschungsförderungsfonds für die gewerbliche Wirtschaft (FFF) im Rahmen des Projekts „E-direct Marketing Solution“, durch den Fond zur Förderung wissenschaftlicher Forschung (FWF) im Rahmen des SFB #010 „Selbstlernende Systeme und Modellierung in den Sozial- und Wirtschaftswissenschaften“, durch die Jubiläumssstiftung der Wirtschaftsuniversität Wien im Rahmen des Projekts „Statistical Computing with R“, durch den Australian Research Council (ARC) im Rahmen des Projekts „Advanced Issues in Market Segmentation Research“, und durch die Österreichische Akademie der Wissenschaften (ÖAW) durch ein DOC-FFORTE Stipendium.

Contents

Chapter 1

Introduction

This thesis is concerned with the identification and estimation of finite mixture models. Finite mixtures are a popular method for statistical modelling due to their flexibility and interpretational advantages for certain applications. The main focus is on mixtures of regression models. They have increased in popularity in the last decades and different special cases have been applied in a lot of areas. However, a thorough investigation of the properties of the general model class are still missing.

In this thesis general characteristics of this model class such as identifiability are discussed and sufficient conditions to guarantee identifiability are given. Identifiability is a theoretic concept and theoretic identifiability does not guarantee practical identifiability and the presence of theoretic identifiability problems does not exclude that these models can be reasonably used or that sensible results can be derived, such as in latent class analysis where in general only local identifiability can be guaranteed and the presence of different parameterizations for the same mixture is even sometimes known.

To complement the theoretical results and to allow data set and model specific conclusions model diagnostics are an essential tool. They can be used to check for identifiability problems. In addition the model fit can be assessed or a suitable model selected. Model diagnostics can rely on standard asymptotic theory or resampling methods can be applied. We focus on bootstrapping techniques which may provide additional insights given the asymptotic results. The application of resampling techniques is in general straightforward and these methods are easily implemented even though the methods are computationally intensive. However, other problems arise similar to those already known for Bayesian estimation, such as label switching.

The estimation of these models with the EM algorithm is implemented in the package **flexmix** (Leisch 2004) in R, an environment for statistical computing and graphics (R Development Core Team 2006). The package design and implementational details are described and the application illustrated. All computations in this thesis are made in R using packages **flexmix** (Leisch 2004) and **flexclust** (Leisch 2006).

1.1 Finite mixture models

Finite mixture models are a popular technique and are used in a lot of different areas, such as astronomy, biology, marketing or medicine. On the one hand they provide a flexible method for modelling unobserved heterogeneity and on the other hand they can be applied to approximate general distribution functions in a semi-parametric way. Finite mixture models have been known for more than 100 years. Early applications are given in Newcomb (1886) and Pearson (1894). An introduction and overview on mixture models are given in the following monographs: Everitt and Hand (1981); Titterington et al. (1985); McLachlan and Basford (1988); McLachlan and Peel (2000) and, recently, Frühwirth-Schnatter (2006).

The popularity of finite mixture models was damped at the beginning due to estimation difficulties where moment estimators and graphical methods were used. With the introduction of the Expectation Maximization (EM) algorithm (Dempster et al. 1977) maximum likelihood estimation became the most popular estimation method due to its advantages, such as easy, straightforward implementation and general applicability. The EM algorithm is in fact not a single algorithm, but a framework for maximum likelihood estimation where different component specific models can be specified. In addition there exist variations such as the stochastic EM (SEM; Diebolt and Ip 1996) or the classification EM (CEM; Celeux and Govaert 1992). These modification have been proposed to overcome known difficulties of the EM algorithm such as slow convergence or only convergence to a local optimum.

Bayesian estimation became feasible with the introduction of a Gibbs sampling scheme using data augmentation (Diebolt and Robert 1994). Even though this sampling scheme can be easily implemented, new problems have arisen with these MCMC methods such as label switching or empty components during sampling with improper priors. The convergence behavior of the MCMC sampler has also been criticized as the component specific estimates should be identical in the case of symmetric priors (Celeux et al. 2000). If this behavior is not observed it is obvious that the sampler has not equally visited all modes of the likelihood.

1.2 Overview of the thesis

This thesis focuses on different aspects of finite mixture modelling. Starting with model specification and notation it discusses identifiability and related problems, model diagnostics especially with the use of resampling methods and implementation of ML estimation in **flexmix**.

Chapter 2 presents the class of mixture models which are covered in this thesis. The focus is on mixtures of distributions also known as model-based clustering and mixtures of generalized linear models (GLMs; McCullagh and Nelder 1989) where each of the components is from the same parametric distribution family for all components. The notation of this class of mixture models is introduced. The estimation within a frequentist maximum likelihood setting is described.

Chapter 3 discusses identifiability problem encountered when fitting finite mixture models. Trivial and generic identifiability problems are distinguished. Even though trivial identifiability problems can be theoretically easily prevented by imposing constraints, they might still cause difficulties in estimation and diagnostics. Sufficient conditions for (generic) identifiability are presented for mixtures of GLMs and mixtures of multinomial logit models. These conditions are a generalization of those given in Hennig (2000) for mixtures of Gaussian regression models.

Chapter 4 focuses on model diagnostics using resampling techniques. Bootstrap methods are popular methods for assessing the model fit (Davison and Hinkley 1997). They can either be used if standard asymptotic theory is not available, to difficult to derive or to complement these results. The application of these methods to assess the model fit and determine different "genuine" modes is outlined and illustrated on several examples.

Chapter 5 discusses the implementation in R. Package **flexmix** (Leisch 2004; Grün and Leisch 2006a,b) provides functionality for ML estimation with the EM algorithm. In addition functionality for (automatic) model selection and for inspecting the fitted model is available.

Chapter 6 summarizes the main findings of the thesis. The Appendix contains the mathematical details of the proof for the sufficient identifiability conditions for mixtures of multinomial logit models (Appendix A) and illustrates the wide applicability of the models and the use of the package **flexmix** on various examples (Appendix B).

Chapter 2

Model specification and estimation

This chapter introduces the model class covered in this thesis and the notation used. Models for model-based clustering, i.e. finite mixtures of distributions, are described as well as finite mixtures of generalized linear models. The most important distributions are covered for this case: Gaussian, Poisson, gamma and binomial/ multinomial. For convenience a distinction between the Gaussian, Poisson and gamma distribution and the multinomial distribution is made. The estimation of these models is outlined where focus is given on maximum likelihood estimation with the EM algorithm.

2.1 Model specification

A general class of finite mixture models is given by:

$$H(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}, \boldsymbol{\Theta}) = \sum_{s=1}^{S} \pi_s(\boldsymbol{w}, \boldsymbol{\alpha}) F_s(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\vartheta}_s),$$

where $H(\cdot|\cdot)$ denotes the mixture distribution and $\boldsymbol{\Theta}$ the vector of all parameters. The dependent variables are $\boldsymbol{y}$, the independent variables $\boldsymbol{x}$ and the concomitant $\boldsymbol{w}$. F_s is the component specific distribution function. The component specific parameters are given by $\boldsymbol{\vartheta}_s$. In the following it is assumed that the mixture density exists and is denoted by $h(\cdot|\cdot)$ and that the component specific density functions are given by $f_s(\cdot|\cdot)$.

For the component weights π_s it holds that

$$\sum_{s=1}^{S} \pi_s(\boldsymbol{w}, \boldsymbol{\alpha}) = 1 \quad \text{and} \quad \pi_s(\boldsymbol{w}, \boldsymbol{\alpha}) > 0, \, \forall s, \boldsymbol{\alpha}, \boldsymbol{w} \tag{2.1}$$

where $\boldsymbol{\alpha}$ are the parameters of the concomitant variable model. It should be noted that in this definition the component weights are strictly positive and no empty components are allowed.

Different concomitant variable models are possible to determine the component weights (Dayton and Macready 1988). The mapping function only has to fulfill condition (2.1). If the concomitant variables $\boldsymbol{w}$ are categorical variables they induce different groups in the observations and different component weights can be estimated for each of the groups. This case as well as the case where numeric concomitant variables are present is covered by using multinomial logit models for the π_s given by

$$\pi_s(\boldsymbol{w}, \boldsymbol{\alpha}) = \frac{e^{\boldsymbol{w}'\boldsymbol{\alpha}_s}}{\sum_{u=1}^{S} e^{\boldsymbol{w}'\boldsymbol{\alpha}_u}}$$

with $\boldsymbol{\alpha} = (\boldsymbol{\alpha}'_s)_{s=1,\ldots,S}$ and $\boldsymbol{\alpha}_1 \equiv \boldsymbol{0}$, i.e. the first component is the baseline.

Finite mixtures where the component-specific densities are from different parametric families have been successfully used in the past, e.g., to model outliers (see Dasgupta and Raftery 1998). However, in the following only finite mixtures where the component specific densities are from the same parametric family are considered, i.e. $F_s \equiv F$ for notational simplicity.

If the component distributions are the same, we can distinguish between component specific parameters $\boldsymbol{\vartheta}_s$ which are equal over all components and those which vary between the components. Those parameters which are equal are referred to as fixed effects and the others as varying effects. This is similar to models with random effects (e.g., Diggle et al. 1994; Pinheiro and Bates 2000) the main difference is that in our case the grouping variable for the varying effects is unknown and has to be estimated. Thus the model is actually closer to the varying-coefficients modelling framework (Hastie and Tibshirani 1993), using convex combinations of discrete points as functional form for the varying coefficients. The covariates of the fixed effects are in the following denoted by $\boldsymbol{z}$, while those of the varying effects are denoted by $\boldsymbol{x}$. The dimension of the vectors is assumed to be U for $\boldsymbol{x}$ and V for $\boldsymbol{z}$.

Often repeated measurements are given for some individuals, i.e. the class membership is fixed for these observations. These observations are in general assumed to be independent given the class membership. The following notation takes this into account: T denotes the set of all individuals in the population and R_t is the index set of observations belonging to individual $t \in T$. The concomitant variables are assumed to be constant for each individual as they are used to determine specific component weights for each individual given its values of the concomitant variables. Let

$$\begin{aligned} \boldsymbol{X} &:= (\boldsymbol{x}_r' : r \in R_t, t \in T) \\ \boldsymbol{Z} &:= (\boldsymbol{z}_r' : r \in R_t, t \in T) \\ \boldsymbol{Y} &:= (\boldsymbol{y}_r' : r \in R_t, t \in T) \\ \boldsymbol{W} &:= (\boldsymbol{w}_t' : t \in T). \end{aligned}$$

Given these assumptions the mixture distribution for T individuals is denoted by:

$$H(\boldsymbol{Y}|\boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{\Theta}) = \prod_{t \in T} \sum_{s=1}^{S} \pi_s(\boldsymbol{w}_t, \boldsymbol{\alpha}) \prod_{r \in R_t} F(\boldsymbol{y}_r|\boldsymbol{x}_r, \boldsymbol{z}_r, \boldsymbol{\vartheta}_s).$$

2.1.1 Model-based clustering

The use of finite mixtures of distributions is referred to as model-based clustering, as in these models assumptions on the distributions of each of the clusters are made. This is in contrast to other clustering techniques such as k-means or hierarchical clustering (e.g., Kaufman and Rousseeuw 1990) which use a more heuristic approach. In this case $\boldsymbol{w}, \boldsymbol{x} \equiv \boldsymbol{1}$ and the dependent variable $\boldsymbol{y}$ is often multivariate.

Finite mixtures of Gaussian distributions are often used for numeric multivariate variables. The aim is either to find groups in the data or vector quantization, i.e. data compression. As for the full model with arbitrary variance-covariance matrices of the components the number of parameters is equal to $K(1 + p\frac{(p+3)}{2}) - 1$, where p denotes the dimension of the input data, different restrictions on the variance-covariance matrices have been proposed to get more parsimonious representations (see Celeux and Govaert 1995; Fraley and Raftery 2002). This includes restrictions to have equal variance-covariance matrices over the components or to allow only for diagonal matrices.

For multivariate binary or categorical data finite mixture models are often used to describe the data by several components under the assumption that for each component the different variables are independent. This is also often referred as latent class analysis (Goodman 1974). These models are very popular for applications in psychology or marketing in order to segment respondents into groups given questionnaire answers collected on a binary or ordinal scale or to account for correlation between the observations. An example is given in Section 4.1.2 for market basket analysis.

A combination of numeric and categorical variables often occurs in practice and under the assumption of independence multivariate mixture models can be estimated. This data is also referred to as mixed-mode data and finite mixtures for this kind of data are described in Everitt (1988) and Hunt and Jorgensen (1999).

2.1.2 Finite mixtures of GLMs: Gaussian, Poisson and gamma

In this case dependent and independent variables are given. Furthermore, F is from the exponential family of distributions and for each component a generalized linear model is fitted. These models are also referred to as GLIMMIX models (Wedel and DeSarbo 1995).

The component specific parameters are given by $\boldsymbol{\vartheta}_s = (\boldsymbol{\beta}'_s, \phi_s)$ where $\boldsymbol{\beta}_s$ are the regression coefficients and ϕ_s is the dispersion parameter. If there are also fixed effects for the regression parameters specified they are denoted by

$\boldsymbol{\gamma}$. The dispersion can either be assumed to be varying, which is denoted by ϕ_s, or to be fixed denoted by ψ.

The density of a distribution from the exponential family is given by

$$f(y|\theta, \phi) = \exp\left\{\frac{y\theta - b(\theta)}{a(\phi) + c(y, \theta)}\right\}$$

(see McCullagh and Nelder 1989, p. 28). In this case it is assumed that the dependent variable is univariate.

Generalized linear models are extended to account for unobserved heterogeneity in the population by introducing varying effects following a finite mixture distribution. Assume there are S latent classes with component weights $\pi_s > 0$, $\sum_{s=1}^{S} \pi_s = 1$. If observation r belongs to component s the model with varying and fixed effects is given by

$$\begin{aligned} \eta &= g\left(\mathbb{E}[y_r|\boldsymbol{x}_r, \boldsymbol{z}_r]\right) \\ &= \boldsymbol{x}_r'\boldsymbol{\beta}^s + \boldsymbol{z}_r'\boldsymbol{\gamma} \end{aligned}$$

where η is the linear predictor and $g(\cdot)$ the link function. $\boldsymbol{x}_r$ are the covariates and $\boldsymbol{\beta}^s$ the coefficients of the varying effects, and $\boldsymbol{z}_r$ the covariates and $\boldsymbol{\gamma}$ the coefficients of the fixed effects.

The mixture distribution can now be written as

$$H(\boldsymbol{Y} \,|\, \boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{\Theta}) = \prod_{t \in T}\left[\sum_{s=1}^{S} \pi_s(\boldsymbol{w}_t, \boldsymbol{\alpha}) \prod_{r \in R_t} F(y_r|\theta_r^s, \Phi)\right],$$

where $F(\cdot|\theta, \Phi)$ is the Gaussian, Poisson or gamma with canonical parameter θ and dispersion parameter Φ. For the parameter vectors it holds that

$$g^{-1}\left(\theta_r^s\right) = \boldsymbol{x}_r'\boldsymbol{\beta}^s + \boldsymbol{z}_r'\boldsymbol{\gamma}$$

and for the dispersion parameter

$$\Phi = \begin{cases} \phi_s & \text{for varying effects} \\ \psi & \text{for fixed effects.} \end{cases}$$

The total parameter vector $\boldsymbol{\Theta}$ is then either equal to $((\pi_s, \boldsymbol{\beta}^s, \phi_s)_{s=1,\ldots,S}, \boldsymbol{\gamma})$ or $((\pi_s, \boldsymbol{\beta}^s)_{s=1,\ldots,S}, \boldsymbol{\gamma}, \psi)$.

2.1.3 Finite mixtures of GLMs: Multinomial logit

Assume a categorical dependent variable $Y \in \{1, \ldots, K\}$ to be given, and let $\mathbb{P}(Y = k|\boldsymbol{x})$ be the probability that the dependent variable Y equals k given the covariate values $\boldsymbol{x}$. Two popular regression models for these probabilities are the *multinomial logit* and *conditional logit* model, see e.g., Soofi (1992). The multinomial logit model uses a common set of predictors $\boldsymbol{x}$ for all levels of Y and choice-specific parameter vectors. The conditional logit model on the other hand allows for alternative-specific predictors $\boldsymbol{x}_k$ but uses the same parameter vector for all of them.

The combined multinomial and conditional logit model is given by

$$\mathbb{P}(Y = k|\boldsymbol{x}) \quad = \quad \frac{e^{\boldsymbol{x}'_{1,k}\gamma_1 + \boldsymbol{x}'_2\gamma_{2,k}}}{\sum_{u=1}^{K} e^{\boldsymbol{x}'_{1,u}\gamma_1 + \boldsymbol{x}'_2\gamma_{2,u}}}, \quad k = 1, \ldots, K$$

such that

$$\text{logit}[\mathbb{P}(Y = k|\boldsymbol{x})] \quad = \quad \boldsymbol{x}'_{1,k}\gamma_1 + \boldsymbol{x}'_2\gamma_{2,k}, \quad k = 1, \ldots, K,$$

where $\boldsymbol{x}_{1,k}$ are the covariates for the conditional logit part and $\boldsymbol{x}_2$ the covariates for the multinomial logit part. For identifiability different contrasts can be imposed, as e.g., taking category K as baseline and constraining $\boldsymbol{\gamma}_{2,K} = \mathbf{0}$ and $\boldsymbol{x}_{1,K} = \mathbf{0}$.

If there is a mixture distribution introduced to account for unobserved heterogeneity in the population and if it is assumed that observation number r belongs to component s the logit model with varying and fixed effects is given by

$$\text{logit}[\mathbb{P}(Y_r = k|\boldsymbol{x}_r, \boldsymbol{z}_r, s)] \quad = \quad \boldsymbol{x}'_{1,k,r}\boldsymbol{\beta}_1^s + \boldsymbol{x}'_{2,r}\boldsymbol{\beta}_{2,k}^s + \boldsymbol{z}'_{1,k,r}\gamma_1 + \boldsymbol{z}'_{2,r}\gamma_{2,k},$$

$\forall k = 1, \ldots, K$ where $\boldsymbol{x}_r$ and $\boldsymbol{\beta}$ are the covariates and the coefficients of the varying effects, and $\boldsymbol{z}_r$ and $\boldsymbol{\gamma}$ are the covariates and the coefficients of the fixed effects.

As will be shown in Section 3.3.3 repeated measurements for some individuals are an important information in guaranteeing identifiability of the model. Therefore, the following notation is convenient, where T denotes the set of all individuals in the population and R_t is the index set of observations belonging to individual $t \in T$. All observations for a single individual t with equal covariate values $\boldsymbol{x}_r$ and $\boldsymbol{z}_r$ can be combined using a new multinomial variable as dependent variable. Let R_t^* be the index set of *unique* covariate vectors $(\boldsymbol{x}_r', \boldsymbol{z}_r')$ and N_r the number of times this covariate vector occurs in R_t. The dependent variable $\boldsymbol{y}_r \in \mathbb{N}^K$ for these unique covariate points is the vector of counts for each category, with $\mathbf{1}_K' \boldsymbol{y}_r = N_r$ where $\mathbf{1}_K$ is a vector of K ones.

To simplify notation all unique covariate points and the dependent variables are row-wise combined into matrices. Let

$$\begin{aligned}
\boldsymbol{X}_{1,k} &:= (\boldsymbol{x}_{1,k,r}' : r \in R_t^*, t \in T) \\
\boldsymbol{X}_2 &:= (\boldsymbol{x}_{2,r}' : r \in R_t^*, t \in T) \\
\boldsymbol{X}_k &:= (\boldsymbol{X}_{1,k}, \boldsymbol{X}_2) \\
\boldsymbol{X} &:= (\boldsymbol{X}_k : k = 1, \ldots, K),
\end{aligned}$$

and let $\boldsymbol{Z}_{1,k}$, $\boldsymbol{Z}_2$, $\boldsymbol{Z}_k$, $\boldsymbol{Z}$ and $\boldsymbol{Y}$ be defined analogously.

The mixture distribution can now be written as

$$H(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{W}, \boldsymbol{\Theta}) = \prod_{t \in T} \left[\sum_{s=1}^{S} \pi_s(\boldsymbol{w}_t, \boldsymbol{\alpha}) \prod_{r \in R_t^*} F(\boldsymbol{y}_r | N_r, \boldsymbol{\theta}_r^s) \right],$$

where $F(\cdot | N, \boldsymbol{\theta})$ is the multinomial distribution with repetition parameter N and probability parameter vector $\boldsymbol{\theta} \in (0,1)^K$. For the probability parameter vectors it holds that

$$\operatorname{logit}[\theta_{k,r}^s] = \boldsymbol{x}_{1,k,r}' \boldsymbol{\beta}_1^s + \boldsymbol{x}_{2,r}' \boldsymbol{\beta}_{2,k}^s + \boldsymbol{z}_{1,k,r}' \gamma_1 + \boldsymbol{z}_{2,r}' \gamma_{2,k}.$$

The total parameter vector $\boldsymbol{\Theta}$ is equal to $((\pi_s, \boldsymbol{\beta}_s)_{s=1,\ldots,S}, \boldsymbol{\gamma})$ where $\boldsymbol{\beta}_s =$

$(\boldsymbol{\beta}^s_1, (\boldsymbol{\beta}^s_{2,k})_{k=1,\ldots,K})$ and $\boldsymbol{\gamma} = (\boldsymbol{\gamma}_1, (\boldsymbol{\gamma}_{2,k})_{k=1,\ldots,K})$.

2.2 Estimation

Finite mixture models can be either estimated within a frequentist framework by using maximum likelihood, within a Bayesian framework, using moment estimators (Lindsay 1989) or graphical tools (Titterington et al. 1985). An important characteristic of the estimation method is if the number of components has to be fixed a-priori or is simultaneously estimated.

In the following maximum likelihood estimation with the EM algorithm is described as this is the most popular estimation method in a frequentist setting. Bayesian estimation using MCMC samplers is only shortly described as it is not the main focus of the thesis. There exists, however, a relation between the problems which occur for this estimation method and those arising for model diagnostics using resampling techniques in a frequentist setting.

2.2.1 Frequentist maximum likelihood

There exist different methods for frequentist estimation of finite mixture models. The most popular is the EM algorithm (Dempster et al. 1977; McLachlan and Krishnan 1997) which aims at determining the ML estimator for a finite mixture model with a given number of components K. It has the advantage that it provides a general framework for estimating different kinds of mixture models as often only the M-step has to be modified if different component specific models or concomitant variable models are used. In addition, already available tools for weighted maximum likelihood estimation might be used. Nevertheless, there are also some disadvantages known such as slow convergence or that one might get stuck in local optima, i.e. it is in general difficult to ensure that the root corresponding to the maximum likelihood estimator was detected.

For determining the non-parametric ML estimator (NPMLE) different estimation methods have been proposed (Böhning et al. 1992; Lindsay 1995).

In this case the number of components is not fixed a-priori. The proposed methods focus on the special case without concomitant variables and independent variables, i.e. $\boldsymbol{w}, \boldsymbol{x} \equiv \mathbf{1}$ and are, for example, the Vertex Exchange Method (VEM) or the Vertex Directon Method (VDM). They exploit the characteristic that the directional derivatives of the log likelihood should be nonpositive for the NPMLE estimator.

EM algorithm

The EM algorithm uses a data augmentation scheme and is a general estimation method in the presence of missing data. In the case of finite mixture models the missing data is the latent variable $\boldsymbol{D}_t \in \{0,1\}^S$ for each individual t which indicates the component membership. This means that D_{ts} equals 1 if individual t is from component s and 0 otherwise. The data is therefore augmented by estimates of the component memberships, i.e. the estimated a-posteriori probabilities $\hat{p}_{ts}$.

For simplicity of notation it is assumed in the following that the component density function $f(\cdot|\cdot)$ takes all observations from each individual as arguments and that the independent variables $\boldsymbol{x}_t$ denote the variables of the varying and fixed effects for individual t. For a sample of T individuals $\{(\boldsymbol{y}_1, \boldsymbol{x}_1, \boldsymbol{w}_1), \ldots, (\boldsymbol{y}_T, \boldsymbol{x}_T, \boldsymbol{w}_T)\}$ the EM-algorithm is given by:

E-step: Given the current parameter estimates $\boldsymbol{\Theta}^{(j)}$ in the j-th iteration, replace the missing data D_{ts} by the estimated a-posteriori probabilities

$$\hat{p}_{ts} = \frac{\pi_s(\boldsymbol{w}_t, \boldsymbol{\alpha}^{(j)}) f(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\vartheta}_s^{(j)})}{\displaystyle\sum_{u=1}^{S} \pi_u(\boldsymbol{w}_t, \boldsymbol{\alpha}^{(j)}) f(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\vartheta}_u^{(j)})}.$$

M-step: Given the estimates for the a-posteriori probabilities $\hat{p}_{ts}$ (which are functions of $\boldsymbol{\Theta}^{(j)}$), obtain new estimates $\boldsymbol{\Theta}^{(j+1)}$ of the parameters by maximizing

$$Q(\boldsymbol{\Theta}^{(j+1)}|\boldsymbol{\Theta}^{(j)}) = Q_1(\boldsymbol{\vartheta}^{(j+1)}|\boldsymbol{\Theta}^{(j)}) + Q_2(\boldsymbol{\alpha}^{(j+1)}|\boldsymbol{\Theta}^{(j)}),$$

where

$$Q_1(\boldsymbol{\vartheta}^{(j+1)}|\boldsymbol{\Theta}^{(j)}) = \sum_{t=1}^{T}\sum_{s=1}^{S} \hat{p}_{ts} \log(f(\boldsymbol{y}_t|\boldsymbol{x}_t, \boldsymbol{\vartheta}_s^{(j+1)})) \tag{2.2}$$

and

$$Q_2(\boldsymbol{\alpha}^{(j+1)}|\boldsymbol{\Theta}^{(j)}) = \sum_{t=1}^{T}\sum_{s=1}^{S} \hat{p}_{ts} \log(\pi_s(\boldsymbol{w}_t, \boldsymbol{\alpha}^{(j+1)})). \tag{2.3}$$

Q_1 and Q_2 can be maximized separately. The maximization of Q_1 gives new estimates $\boldsymbol{\vartheta}^{(j+1)}$ and the maximization of Q_2 gives $\boldsymbol{\alpha}^{(j+1)}$. Q_1 is maximized using weighted ML estimation of GLMs and Q_2 using weighted ML estimation of multinomial logit models.

There exist variants of the EM algorithm such as the stochastic EM (SEM; Diebolt and Ip 1996) or the classification EM (CEM; Celeux and Govaert 1992). For both algorithms an additional step is made between estimation and maximization. This additional step is given for the SEM by:

S-step: Given the a-posteriori probabilities $\hat{\boldsymbol{p}}_t := (\hat{p}_{ts})_{s=1,\ldots,S}$ draw

$$\hat{\boldsymbol{D}}_t \sim \text{Mult}(\hat{\boldsymbol{p}}_t, 1)$$

where $\text{Mult}(\theta, N)$ denotes the multinomial distribution with success probabilities θ and number of repetitions N and $\hat{\boldsymbol{D}}_t := (\hat{D}_{ts})_{s=1,\ldots,S}$.

For the CEM this step is given by:

H-step: Given the a-posteriori probabilities define

$$\hat{D}_{ts} = \begin{cases} 1 & \text{if } s = \arg\max_{u=1,\ldots,S} \hat{p}_{tu} \\ 0 & \text{otherwise} \end{cases}$$

The $\hat{D}_{ts}$ are used instead of the $\hat{p}_{ts}$ in the E-step.

Both of these variants have been proposed to improve the performance of the EM algorithm, because the EM algorithm converges in general rather slowly and only to a local optimum. The convergence behavior is better for the CEM, while SEM can escape convergence to a local optimum. However, the CEM does not give ML estimates as it maximizes the complete likelihood. For SEM good approximations of the ML estimator are in general obtained if the parameters where the maximum likelihood was encountered are used. Another estimate for parameters could be from the SEM to use the mean after discarding a suitable number of burn-ins. An implementational advantage of both variations is that no weighted maximization is necessary in the M-step.

If there are only varying effects for the component distributions specified, the parameters can be separately determined for each component. However, if there are also fixed effects, the vector of observations $\boldsymbol{y} = (\boldsymbol{y}_t)_{t=1,\ldots,T}$ has to be replicated S times and the covariate matrix $(\boldsymbol{X}, \boldsymbol{Z})$ is given by

$$\begin{aligned}\boldsymbol{X} &= \mathbf{1}_S \otimes (\boldsymbol{x}_t')_{i=1,\ldots,T}\\ \boldsymbol{Z} &= \boldsymbol{I}_S \otimes (\boldsymbol{z}_t')_{i=1,\ldots,T},\end{aligned}$$

where $\mathbf{1}_S$ is a vector of 1s of length S, $\boldsymbol{I}_S$ is the identity matrix of dimension $S \times S$, and $\otimes$ denotes the Kronecker product.

Before each M-step the average component sizes (over the given data points) are checked and components which are smaller than a given (relatively) small size are omitted in order to avoid too small components where fitting problems might arise. This strategy has also been recommended for SEM (Celeux and Diebolt 1988) in order to determine the number of components. If the algorithm is started with too many components they will be omitted during the estimation process. The algorithm is stopped if the relative change in the likelihood is smaller than a pre-specified ϵ or the maximum number of iterations is reached.

It has been shown that the values of the likelihood are monotonically increased during the EM algorithm. This ensures the convergence of the EM algorithm if the likelihood is bounded. Unboundedness of the likelihood, however, might occur at the edge of the parameter space (Kiefer 1978), e.g., if the variance of one component tends to zero for mixtures of Gaussian distributions. As even in the case of boundedness only the detection of a local maximum can be guaranteed, it is in general recommended to repeat the EM algorithm with different initializations and to choose as final solution the one with the maximum likelihood. Different initialization strategies for the EM algorithm have been proposed, as its convergence to the optimal solution depends on the initialization (Biernacki et al. 2003; Karlis and Xekalaki 2003).

2.2.2 Bayesian MCMC sampling

Estimation within a Bayesian framework has become popular with the advent of MCMC methods. An overview on the different sampling approaches is given in Marin et al. (2005), Jasra et al. (2005) and Frühwirth-Schnatter (2006, chap. 3). Gibbs sampling is the most commonly used approach and it is done by augmenting the data with the unobservable variable of class membership similar to the EM algorithm (Diebolt and Robert 1994). A drawback of the Gibbs sampler is that it might fail to escape the attraction area of one mode and therefore does not explore the entire parameter space. It was therefore suggested to use Metropolis-Hastings sampling schemes (Celeux et al. 2000). Alternatively, the permutation sampler (Frühwirth-Schnatter 2001) may be used.

Additional approaches have been proposed for simultaneously estimating the number of components and the parameter values. This includes reversible jump MCMC (Richardson and Green 1997) and the inclusion of birth-and-death processes (Stephens 2000a).

Chapter 3

Identifiability

This chapter defines the term identifiability and discusses different issues such as label switching, overfitting and generic identifiability. Label switching or overfitting are in general a problem for estimating finite mixtures but might be resolved in a post-processing step. By contrast mixtures suffering from generic identifiability problems might be fitted without any problems as for example local identifiability is given and these problems are then only detected if the model is further investigated.

3.1 Definition

Statistical models are in general represented by parameter vectors. For finite mixture models the parameter vector Θ which consists of the component weights, the component specific parameters as well as the parameters of concomitant variable model determines a mixture distribution, i.e. there is a mapping from the parameter space to the model space. For identifiability this mapping has to be injective, i.e. for each model in the model space there is a unique parameter vector in the parameter space which is mapped to the model. Lack of identifiability can be a problem for model estimation or if parameters are interpreted.

This can be formally defined similar to Titterington et al. (1985) by:

Definition 1 (Identifiability of finite mixtures)**.** *Suppose* $\Theta, \hat{\Theta} \in \Omega$ *are two* *arbitrary parameter vectors which determine a mixture distribution and are given by*

$$\begin{aligned}\Theta &= ((\boldsymbol{\vartheta}_s)_{s=1,\ldots,S}, \boldsymbol{\alpha}) \\ \hat{\Theta} &= \left((\hat{\boldsymbol{\vartheta}}_t)_{t=1,\ldots,\hat{S}}, \hat{\boldsymbol{\alpha}}\right).\end{aligned}$$

Ω *is called* identifiable *if for arbitrary* Θ, $\hat{\Theta} \in \Omega$

$$H(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}, \Theta) \equiv H(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{w}, \hat{\Theta})$$

holds for all admissible $\boldsymbol{y}$, $\boldsymbol{x}$ *and* $\boldsymbol{w}$ *only if*

$$S = \hat{S}$$

and there exist a suitable ordering of the component indices such that

$$\boldsymbol{\alpha}_s = \hat{\boldsymbol{\alpha}}_s \qquad\qquad \boldsymbol{\vartheta}_s = \hat{\boldsymbol{\vartheta}}_s$$

$\forall s = 1, \ldots, S$.

In the following let Ω denote the space of admissible parameters for S-component mixtures where the following conditions are fulfilled

- $\pi_s > 0 \ \forall s = 1, \ldots, S$,
- $\forall s, t \in \{1, \ldots, S\}$: $s \neq t \rightarrow \boldsymbol{\vartheta}_s \neq \boldsymbol{\vartheta}_t$.

These two conditions prevent overfitting and identifiability problems which occur due to empty components where $\boldsymbol{\vartheta}_s$ cannot be uniquely determined and components with equal component parameter vectors and different values for $\boldsymbol{\alpha}$ are possible.

Let $\mathscr{A}_S = \mathscr{A}_S(F, \Omega)$ be the set of all finite mixture models with S components, component specific distribution function F and mixture distributions of form $H(\cdot|\cdot, \Theta)$, $\Theta \in \Omega$. Each parameter vector $\Theta \in \Omega$ corresponds to one model $a \in \mathscr{A}_S$, but each model a has at least $S!$ parameterizations Θ due to all possible permutations of the components, also known as *label switching* (Redner and Walker 1984).

$\mathscr{A}_S$ induces a system of equivalence classes Ξ on Ω where two elements of Ω are in the same equivalence class if they correspond to the same model a:

$$\Theta, \tilde{\Theta} \in \Xi \quad \Leftrightarrow \quad H(\cdot|\cdot, \Theta) \equiv H(\cdot|\cdot, \tilde{\Theta})$$

(see also Hennig 2000). The usual definition of model identifiability is that either all equivalence classes contain only one element (which is trivially not true for mixture models), or that at least a unique representative for each equivalence class can be selected.

Let ident$(\Omega) \subset \Omega$ be the subset of parameterizations which contain only one permutation of each possible set of component parameters. ident(Ω) can be obtained from Ω by imposing an ordering constraint on the components with respect to a certain parameter (or a combination of several parameters). We refer to any identifiability problems which are present for ident(Ω) as *generic* (Frühwirth-Schnatter 2006).

3.2 Label switching

As described in Section 2.2 finite mixture models can be either estimated within a frequentist or a Bayesian framework. For a fixed S, the ML solution is in general estimated in the unrestricted parameter space where the likelihood is multimodal and a unique solution is determined by relabelling the components with respect to an arbitrary ordering constraint, as e.g. on the prior probabilities. Label switching is of concern if bootstrapping with random initialization of the EM algorithm is used for model diagnostics (Grün and Leisch 2004) as the choice of ordering constraint influences the component specific analysis. Label switching does in general not occur if the parametric bootstrap is applied with initialization in the solution (McLachlan and Peel 2000, p. 70) and hence, the bootstrap solutions can be directly used without reordering them to fulfill a certain constraint.

For Bayesian estimation with MCMC methods label switching makes it impossible to make component specific inferences directly from the MCMC draws. Different approaches have been proposed to determine suitable estimates. An overview on the different approaches to solve the label switching problem in a Bayesian context is given in Marin et al. (2005), Jasra et al. (2005) and Frühwirth-Schnatter (2006, chap. 3). The methods include specification of an (artificial) ordering constraint, fixing the membership of some observations, applying label-invariant loss functions, cluster and relabelling algorithms and relabelling with respect to the MAP estimate. Stephens (2000b) outlines an approach where the possibility of multiple modes of the likelihood is taken into account and component specific estimates for each of the modes are determined.

Similar to the frequentist framework an ordering constraint can be imposed on one parameter (Diebolt and Robert 1994; Richardson and Green 1997). However, the choice of ordering constraint has a major impact on the marginal a-posteriori distributions of the component specific parameters (Richardson and Green 1997) and it is therefore necessary to find a suitable ordering constraint which takes the geometry of the likelihood into account and induces a unique labelling. This can be done in a post-processing step given the MCMC results as shown by Stephens (1997). Accordingly, Frühwirth-Schnatter (2001) proposes to use a permutation sampler and to relabel the MCMC draws in a post-processing step given an appropriate ordering constraint based on expert knowledge or exploratory analysis of the MCMC samples. The use of an ordering constraint has the drawback, that all components have to differ in one parameter. If the component specific parameter vectors are multivariate, an appropriate ordering constraint might either be difficult to choose or even impossible, because there is no variable where all components differ.

Chung et al. (2004) suggest to break the symmetry of the likelihood by fixing the class membership for one or more observations. This method is only effective if observations with high a-posteriori probabilities (close to one) for one component are selected. They present an example where the results of their approach are as good as if an appropriate ordering constraint is imposed. Hence, their method should be preferred if selecting observations for fixing the membership is easier than determining a suitable ordering constraint. Another advantage of their method is that it is computationally less demanding than clustering or relabelling algorithms. Fixing the membership in a Bayesian analysis signifies increasing the prior information. Hence, Chung et al. (2004) note that the degree of prior information implicit in preclassification should be explored. Obviously, not only label switching but also identifiability problems in a regression setting due to violation of the coverage condition (see Hennig 2000 and Section 3.3) can be eliminated by fixing the membership of sufficiently many observations.

A decision-theoretic approach using label invariant loss functions is given in Celeux et al. (2000) and Hurn et al. (2003). The loss function $L(\Theta, \hat{\Theta})$ describes the loss from estimating the parameter vector with $\hat{\Theta}$ if the true parameter vector is equal to Θ. The Bayes estimator is determined by minimizing the expected posterior loss and is given by

$$\hat{\Theta}^* \quad = \quad \arg\min_{\hat{\Theta}} \mathbb{E}_{\Theta|\boldsymbol{x},\boldsymbol{w},\boldsymbol{y}} L(\Theta, \hat{\Theta})$$

where $L(\Theta, \hat{\Theta})$ is the label invariant loss function and Θ the true parameter vector. The choice of loss function determines if the calculation of the estimator is analytically feasible or if it has to be approximated using e.g. a combination of MCMC sampling and simulated annealing. Different loss functions have been proposed depending on the inferential conclusions which shall be drawn. If inference for the parameters is the purpose Celeux et al. (2000) propose to use a measure of discrepancy based on the Baddeley metric. If the predictive distribution is of interest a symmetrized Kullback-Leibler distance has been proposed (Celeux et al. 2000; Hurn et al. 2003).

Relabelling algorithms based on cluster methods have been proposed in Stephens (1997) and Celeux (1998). Stephens (2000b) shows that the proposed relabelling algorithms are special cases of the decision-theoretic approach using label-invariant loss functions. However, in contrast to the loss functions proposed in Celeux et al. (2000) and Hurn et al. (2003) the relabelling algorithms do not only return an appropriate Bayes estimate $\hat{\Theta}^*$, but determine a suitable labelling of the components for each MCMC draw.

Another reordering scheme is proposed in Marin et al. (2005). The Maximum a Posteriori (MAP) estimator is approximated by determining the parameters of the draw with the maximum a-posteriori density. All other draws are then permuted in order to minimize the canonical scalar product between their parameters and those of the approximate MAP estimator.

3.3 Generic identifiability

In the following only finite mixtures with constant component weights, i.e. $\boldsymbol{w} \equiv 1$, are considered, as research with respect to generic identifiability has focused on this special case. A possible generalization to the more general model depends on assumptions for the concomitant variables, e.g. if it can be assumed that they are independent from the covariates $\boldsymbol{x}$ and $\boldsymbol{z}$ in the component specific models.

3.3.1 Model-based clustering

An overview on different theorems for the identifiability of mixtures of distributions is given in Titterington et al. (1985, pp. 35–42). A necessary and sufficient condition for identifiability is that the component distributions are a linearly independent set over the field of real numbers $\mathbb{R}$ (Yakowitz and Spragins 1968).

It has been shown that finite mixture distributions of several popular continuous parametric distributions are generically identifiable. This comprises the (multivariate) normal, gamma and exponential distribution (Teicher 1963; Yakowitz and Spragins 1968; Titterington et al. 1985). Identifiability has also been shown for location-scale families on the real line (Holzmann et al. 2004) which induces the identifiability of mixtures of von Mises distributions (see also Fraser et al. 1981; Kent 1983).

A discrete identifiable distribution is the Poisson distribution (Teicher 1960). By contrast the discrete and the continuous uniform distributions are not generically identifiable. The binomial and the multinomial distributions are identifiable if the number of components S is limited with respect to the number of repetitions N ($N \geq 2S - 1$; Teicher 1963; Blischke 1964; Titterington et al. 1985; Grün 2002; Elmore and Wang 2003). This constraint is necessary and sufficient for the model class of all mixtures with a maximum of S components.

3.3.2 Finite mixtures of GLMs: Gaussian, Poisson and gamma

The identifiability of mixtures of Gaussian regression models was analyzed by Hennig (2000). He shows that requiring a covariate matrix of full rank – as postulated previously (see for example Wang and Puterman 1998) – is not sufficient. Contrarily, it is necessary to check a coverage condition in order to ensure identifiability. With respect to generic identifiability of finite mixtures of regression models three influencing factors can therefore be distinguished:

- component distribution
- covariate matrix
- repeated observations/labelled observations

Repeated observations where the class membership is fixed are necessary for binomial and multinomial mixtures to be identifiable. In a regression setting repetitions over different covariate points can help in making a mixture identifiable as it changes the set of feasible hyperplanes for the coverage condition. Labels for some observations indicating that they belong to the same component have the same influence.

It can be concluded that the identifiability of a mixture regression model depends on the component distributions, the maximum number of components allowed, the available information per object and the regressor matrix.

In order to present the theorem on sufficient conditions for identifiability of the model presented in Section 2.1.2 we choose a slightly different representation of the observations. We assign different indices to the covariates for the fixed and varying effects:

$$(\boldsymbol{x}_r, \boldsymbol{z}_r) \quad \rightarrow \quad (\boldsymbol{x}_i, \boldsymbol{z}_j).$$

The observations are sorted such that all equal covariates for individual $t \in T$ of the varying effects are grouped together in the index set I_t and we have a set J_i with the indices of all covariates for the fixed effects where the varying effects are equal to $\boldsymbol{x}_i$.

Theorem 1. *The model defined by*

$$H(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\Theta}) = \prod_{t \in T} \Bigg[\sum_{s=1}^{S} \pi_s \prod_{i \in I_t} \prod_{j \in J_i} F(y_{ij} | \theta_{ij}^s, \phi_s) \Bigg]$$

and

$$g^{-1}\left(\theta_{ij}^s\right) = \boldsymbol{x}_i' \boldsymbol{\beta}^s + \boldsymbol{z}_j' \boldsymbol{\gamma}$$

is identifiable if the following conditions are fulfilled:

1. *$q^* > S$ with*

   $$q^* := \min \Bigg\{ q : \forall i^* \in I : \exists H_j \in \{H_1, \dots, H_q\} : \\ \{\boldsymbol{x}_i : i \in I_{t(i^*)}\} \subseteq H_j \wedge H_j \in \mathcal{H}_U \Bigg\}$$

   where $I := \bigcup_{t \in T} I_t$ and $\mathcal{H}_U$ is the set of $H(\alpha) := \{\boldsymbol{x} \in \mathbb{R}^U : \boldsymbol{\alpha}'\boldsymbol{x} = \boldsymbol{0}\}$ where $\boldsymbol{\alpha} \neq \boldsymbol{0}$.

2. *$rk(\boldsymbol{X}, \boldsymbol{Z}) = U + V$ where $rk(\cdot)$ determines the rank of a matrix.*

The proof is straight-forward given the results from Hennig (2000) and Section 3.3.3. The difference to the conditions presented in Hennig (2000) is that fixed effects are allowed in addition to the varying effects and that the possibility of repeated observations for the individuals is taken into account. Condition (1) indicates that the coverage condition necessary for the covariates has only to be fulfilled for the varying effects. In addition this condition is altered to take the repeated measurements into account and hence for each individual t there has to be one of the q hyperplanes through the origin H_j which covers all observations of this individual. For the fixed effects only a rank condition is necessary given in Condition (2), which ensures that each of the covariate matrices as well as the combined covariate matrix have full column rank.

These conditions inidicate that identifiability problems can especially occur if the covariate matrix contains categorical variables. To our knowledge this problem has only been noted by Hennig (2000) and no treatment is proposed in the literature. We refer to identifiability problems due to the violation of the coverage condition as

Intra-component label switching: If the labels are fixed in one covariate point according to some ordering constraint, then labels may switch in other covariate points for different parameterizations of the model.

For mixtures where the component distributions are identifiable this means that the component weights and possible dispersion parameters are unique, but the regression coefficients vary because they depend on the combination of the components between the covariate points. This identifiability problem is also of concern for prediction, because given the class membership the predicted value for new data depends on the chosen solution.

Unidentified mixture models with several isolated non-trivial (global) modes in the likelihood are to some extent more of a theoretical problem, because, e.g., minimal changes of the component weights π_k often make the model identified by breaking symmetry. However, models "close" to an unidentified model will have multiple local modes.

The following example presents a simple mixture of regression models with intra-component label switching. The model is unidentified (with two non-trivial modes) only if both components have exactly the same probability.

Example 1. *Assume we have a standard linear mixture regression with one measurement per object and a single categorical regressor with two levels. The usual design matrix for a model with intercept uses the two covariate points* $\mathbf{x}_1 = (1, 0)'$ *and* $\mathbf{x}_2 = (1, 1)'$*. Furthermore, let the mixture consist of two components with equal component weights. The mixture regression is given*

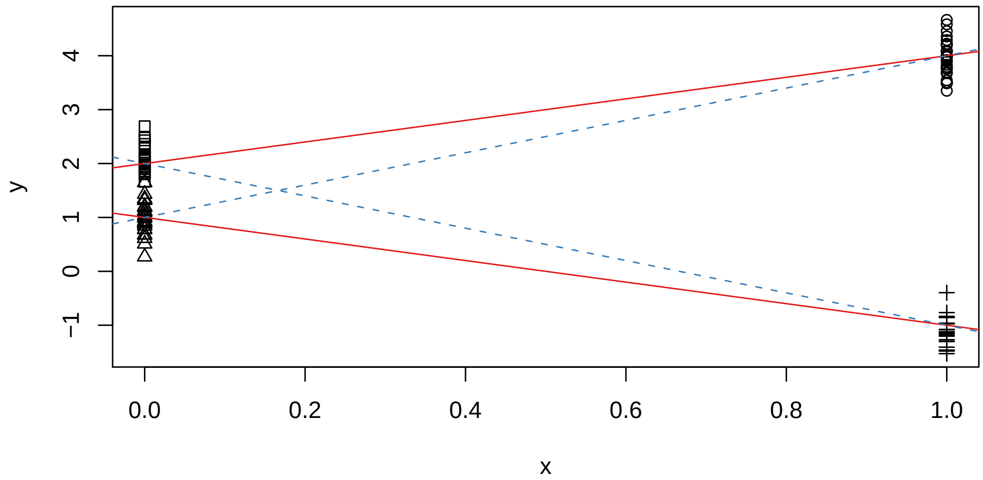


Figure 3.1: Balanced sample from the artificial example with the two theoretical solutions. The solid lines correspond to solution 1 and the dashed lines to solution 2.

by

$$H(y|\mathbf{x},\Theta) \quad = \quad \frac{1}{2}N(\mu_1, 0.1) + \frac{1}{2}N(\mu_2, 0.1)$$

where $\mu_k(\mathbf{x}) = \mathbf{x}'\boldsymbol{\alpha}_k$ *and* $N(\mu,\sigma^2)$ *is the normal distribution.*

Now let $\mu_1(\mathbf{x}_1) = 1$, $\mu_2(\mathbf{x}_1) = 2$, $\mu_1(\mathbf{x}_2) = -1$ *and* $\mu_2(\mathbf{x}_2) = 4$. *As Gaussian mixture distributions are generically identifiable the means, variances and component weights are uniquely determined in each covariate point given the mixture distribution. However, as the coverage condition is not fulfilled, the two possible solutions for* $\boldsymbol{\alpha}$ *are:*

Solution 1: $\boldsymbol{\alpha}_1^{(1)} = (2, \quad 2)'$, $\boldsymbol{\alpha}_2^{(1)} = (1, -2)'$

Solution 2: $\boldsymbol{\alpha}_1^{(2)} = (2, -3)'$, $\boldsymbol{\alpha}_2^{(2)} = (1, \quad 3)'$

In Figure 3.1 a balanced sample with 50 observations in each covariate point is plotted together with the two solutions for combining x_1 *and* x_2*, i.e., this illustrates intra-component label switching.*

This mixture model would be identifiable if

1. *three different covariate points were available, or*

2. *observations for both covariate points for the same object were available, or*

3. *the component weights were unequal, e.g.* $\pi_1 = 0.6$.

Condition 1 is not an option, for instance, for a single 2-level categorical regressor. Condition 2 is not possible if the categorical regressor cannot change for repeated observations of the same subject like, for instance, the gender of a person.

When developing a suitable measurement design, the possibility of these problems to occur should be considered and could therefore influence the proposed covariate matrix.

The coverage condition in Hennig (2000) and in Theorem 1 has the disadvantage that it is only a sufficient condition for a certain model class indicating if there is at least one model which is not identifiable and that it is hard to verify in practice as it is an NP hard problem. In general it will be of interest if a fitted model suffers from identifiability problems. This means that it has to be checked if there exist several modes of the likelihood in the parameter space ident(Ω) given data sets sampled from the fitted mixture model.

3.3.3 Finite mixtures of GLMs: Multinomial logit

The same notation as in the previous section for finite mixtures of Gaussian, Poisson and gamma regressionn models is used to distinguish between different values for the covariates of the fixed and the varying effects, i.e.

$$(\boldsymbol{x}_r, \boldsymbol{z}_r) \quad \rightarrow \quad (\boldsymbol{x}_i, \boldsymbol{z}_j).$$

The number of observations N_r is referred to by N_{ij}. This notation is illustrated by the following example where the data matrix for a given individual t is presented.

I_t	J_i	$\boldsymbol{x}_i$	z_{ij}	N_{ij}	$\boldsymbol{y}_{ij}$
1	1	1.1 0	4	7	3 2 2
1	2	1.1 0	0	3	1 2 0
1	3	1.1 0	1	1	0 0 1
1	4	1.1 0	2	5	2 1 2
2	1	2.7 1	4	1	0 0 1
2	2	2.7 1	0	1	0 1 0
2	3	2.7 1	2	6	4 0 2

Table 3.1: Illustration of notation for a given individual t

Example 2. *Let the dependent categorical variable have three different categories. The covariates $\boldsymbol{x}$ of the random effects consist of a numeric variable and a binary variable, whereas the covariate $\boldsymbol{z}$ of the fixed effects is a categorical variable with 4 categories. For simplicity of presentation these variables are all for a multinomial logit model.*

Assume that for individual t 24 trinomial outcomes are observed at 7 different covariate values. For example, when $\boldsymbol{x} = (1.1, 0)$ and $z = 0$ there are 3 trinomial outcomes observed, a "1" and two "2"s. The varying covariate $\boldsymbol{x}$ assumes two values $(1.1, 0)$, and $(2.7, 1)$. These have, respectively, 16, and 8 replicates where we allow different z values. The corresponding data matrix is given in Table 3.1.

The multinomial and conditional logit models can be combined and formulated within the same framework, as the multinomial logit part can be transformed to a conditional logit model (Agresti 1990, pp. 316–317). The covariates and coefficients are given by

$$\boldsymbol{x}_{k,i} := \begin{pmatrix} \boldsymbol{x}_{1,k,i} \\ \boldsymbol{e}_k \otimes \boldsymbol{x}_{2,i} \end{pmatrix} \in \mathbb{R}^U \qquad \boldsymbol{\beta}^s := \begin{pmatrix} \boldsymbol{\beta}_1^s \\ (\boldsymbol{\beta}_{2,k}^s)_{k=1,\ldots,K-1} \end{pmatrix} \in \mathbb{R}^U$$

$$\boldsymbol{z}_{k,j} := \begin{pmatrix} \boldsymbol{z}_{1,k,j} \\ \boldsymbol{e}_k \otimes \boldsymbol{z}_{2,j} \end{pmatrix} \in \mathbb{R}^V \qquad \boldsymbol{\gamma} := \begin{pmatrix} \boldsymbol{\gamma}_1 \\ (\boldsymbol{\gamma}_{2,k})_{k=1,\ldots,K-1} \end{pmatrix} \in \mathbb{R}^V$$

with $U = U_1 + (K-1)U_2$, $V = V_1 + (K-1)V_2$, $\boldsymbol{e}_k \in \{0,1\}^{K-1}$ is a unit vector with 1 at position k, and $\otimes$ is the Kronecker product.

Sufficient identifiability conditions

Conditional logit We first present sufficient conditions for identifiability of the conditional logit model with varying and fixed effects, results for the combined model are then derived as a corollary.

Theorem 2. *The model defined by*

$$H(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\Theta}) \;=\; \prod_{t\in T}\Big[\sum_{s=1}^{S}\pi_s \prod_{i\in I_t}\prod_{j\in J_i} F(\boldsymbol{y}_{ij}|N_{ij}, \boldsymbol{\theta}_{ij}^{s})\Big]$$

and

$$\ln\left[\frac{\theta^s_{k,ij}}{\theta^s_{K,ij}}\right] \;=\; \boldsymbol{x}'_{k,i}\boldsymbol{\beta}^s + \boldsymbol{z}'_{k,j}\boldsymbol{\gamma}$$

is identifiable if the following conditions are fulfilled:

1. *(a)* $\forall k \in \{1, \ldots, K-1\}$*:* $\exists \tilde{I}_k \neq \emptyset$*:* $\tilde{I}_k \subseteq \bigcup_{t\in T} I_t$ *with:*

$$\sum_{i\in E_{k,i^*}} \sum_{j\in J_i} N_{ij} \geq 2S-1 \qquad \forall i^* \in \tilde{I}_k$$

where $E_{k,i^*} := \{i \in I_{t(i^*)} : \boldsymbol{x}_{k,i} = \boldsymbol{x}_{k,i^*}\}$. $I_{t(i^*)}$ *is defined as the index set of all observations for the individual* t *with covariate vector* $\boldsymbol{x}_{k,i^*}$.

(b) $q^* > S$ *with*

$$q^* := \min\Big\{q : \forall i^* \in \bigcup_{k=1}^{K-1} \tilde{I}_k : \exists H_j \in \{H_1, \ldots, H_q\} : \\ \{\boldsymbol{x}_{k,i} : i \in I_{t(i^*)} \cap \tilde{I}_k, k = 1, \ldots, K-1\} \subseteq H_j \wedge H_j \in \mathcal{H}_U\Big\}$$

where $\mathcal{H}_U$ *is the set of* $H(\alpha) := \{\boldsymbol{x} \in \mathbb{R}^U : \boldsymbol{\alpha}'\boldsymbol{x} = \boldsymbol{0}\}$ *where* $\boldsymbol{\alpha} \neq \boldsymbol{0}$.

2. *rk*$(\boldsymbol{X}, \boldsymbol{Z}) = U + V$ *where rk*$(\cdot)$ *determines the rank of a matrix.*

3. $\boldsymbol{x}_{K,i} = \boldsymbol{0}$ *and* $\boldsymbol{z}_{K,j} = \boldsymbol{0}$ $\forall j \in J_i$, $\forall i \in I_t$, $\forall t \in T$.

The proof is given in Appendix A. Condition (1) guarantees that no intra-component label switching is possible as the number of feasible hyperplanes which are necessary to cover the set of covariate points where marginal identifiability can be guaranteed is larger than the number of components. As the component membership is fixed for each individual only those hyperplanes are feasible where the covariate points from the same individual lie on the same hyperplane. Condition (1a) implies that there exists a $t \in T$ with at least $2S-1$ observations. For these observations the covariates for the varying effects have to be constant, but they can vary for the fixed effects. The inclusion of the set E_{k,i^*} is possible, because the covariates are allowed to change in the other categories of the multinomial distribution. Condition (1b) corresponds to the coverage condition in Hennig (2000) for mixtures of Gaussian regressions which ensures that no intra-component label switching is possible. While in Hennig (2000) only the case of one repetition per individual is considered we generalize the condition for the case where there are repeated observations per individual available.

Condition (2) and (3) correspond to conditions which are necessary for a model without varying effects in order to uniquely determine the coefficients. Condition (2) also ensures that the partition between fixed and varying effects is unique. Other equivalent conditions are possible at this point, as, for instance, to require that the mean value is captured by the fixed effects and therefore is zero for the varying effects. Condition (3) represents the choice of contrasts between the K alternatives. We choose K as baseline category, but again other contrasts are possible as for example to constrain that the sum of the coefficients over the categories is zero. In the future identifiability conditions could be established for these contrasts.

Multinomial and conditional logit The following theorem gives sufficient identifiability conditions for the combined model (without concomitant variables) presented in Section 2.1.3 .

Theorem 3. *The model defined by*

$$H(\boldsymbol{Y} \mid \boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\Theta}) \quad = \quad \prod_{t \in T} \left[\sum_{s=1}^{S} \pi_s \prod_{i \in I_t} \prod_{j \in J_i} F(\boldsymbol{y}_{ij} | N_{ij}, \boldsymbol{\theta}_{ij}^{s}) \right]$$

and

$$\ln \left[\frac{\theta_{k,ij}^{s}}{\theta_{K,ij}^{s}} \right] \quad = \quad \boldsymbol{x}'_{1,k,i} \boldsymbol{\beta}_1^s + \boldsymbol{x}'_{2,i} \boldsymbol{\beta}_{2,k}^s + \boldsymbol{z}'_{1,k,j} \boldsymbol{\gamma}_1 + \boldsymbol{z}'_{2,j} \boldsymbol{\gamma}_{2,k}$$

is identifiable if the following conditions are fulfilled:

1. *(a)* $\forall k \in \{1, \ldots, K-1\}$*:* $\exists \tilde{I}_k \neq \emptyset$*:* $\tilde{I}_k \subseteq \bigcup_{t \in T} I_t$ *with:*

   $$\sum_{i \in E_{k,i^*}} \sum_{j \in J_i} N_{ij} \geq 2S - 1 \qquad \forall i^* \in \tilde{I}_k$$

   where $E_{k,i^*} := \{i \in I_{t(i^*)} : \boldsymbol{x}_{1,k,i} = \boldsymbol{x}_{1,k,i^*} \wedge \boldsymbol{x}_{2,i} = \boldsymbol{x}_{2,i^*}\}$.

   (b) $q^* > S$ *with*

   $$q^* := \min \Bigg\{ q : \forall i^* \in \bigcup_{k=1}^{K-1} \tilde{I}_k : \exists H_j \in \{H_1, \ldots, H_q\} :$$
   $$\{(\boldsymbol{x}'_{1,k,i}, \boldsymbol{x}'_{2,i}) : i \in I_{t(i^*)} \cap \tilde{I}_k, k = 1, \ldots, K-1\} \subseteq H_j \wedge H_j \in \mathcal{H}_U \Bigg\}$$

   where $\mathcal{H}_U$ *is the set of* $H(\alpha) := \{\boldsymbol{x} \in \mathbb{R}^{U_1+U_2} : \boldsymbol{\alpha}'\boldsymbol{x} = \boldsymbol{0}\}$ *where* $\boldsymbol{\alpha} \neq \boldsymbol{0}$.

2. *rk*$(\boldsymbol{X}, \boldsymbol{Z}) = U + V$

3. $\boldsymbol{x}_{1,K,i} = \boldsymbol{0}$ *and* $\boldsymbol{z}_{1,K,j} = \boldsymbol{0}$ $\forall j \in J_i$, $\forall i \in I_t$, $\forall t \in T$, $\boldsymbol{\beta}_{2,K} = \boldsymbol{0}$ *and* $\boldsymbol{\gamma}_{2,K} = \boldsymbol{0}$.

The proof is straight-forward given Theorem 2.

Application to special cases

In the following sections we illustrate which sufficient identifiability constraints can be derived from Theorem 3 for important special cases.

Mixtures of multinomial distributions The simplest case are mixtures of multinomial distributions without a regression part where only a component specific intercept needs to be estimated such that $x_{2,i} \equiv 1 \; \forall i \in I$ and $U_1 = V = 0$:

$$\ln\left[\frac{\mathbb{P}(Y_r = k)}{\mathbb{P}(Y_r = K)}\right] \quad = \quad \boldsymbol{\beta}^s_{2,k} \qquad \forall k = 1, \ldots, K.$$

Condition (1a) ensures that there is at least one individual where the number of observations N is larger or equal to $2S - 1$. Hence we have the same results as in Grün (2002) and Elmore and Wang (2003): The class of mixtures of multinomial distributions with a maximum of S components is identifiable if $N \geq 2S - 1$.

Model in Follmann and Lambert (1991) Theorem 3 generalizes the first set of sufficient conditions in Follmann and Lambert (1991). They considered mixtures of binomial logit distributions where only the intercept followed a finite mixture distribution and all other coefficients were constant. Hence for our model this signifies $K = 2$, $x_{1,1,i} \equiv 1 \; \forall i \in I$, $U_2 = 0$ and V arbitrary.

As there is no difference between multinomial and conditional logit models in the binomial case the model is given by

$$\ln\left[\frac{\mathbb{P}(Y_r = 1|\boldsymbol{z}_r)}{\mathbb{P}(Y_r = 0|\boldsymbol{z}_r)}\right] \quad = \quad \boldsymbol{\beta}^s_1 + \boldsymbol{z}_r'\boldsymbol{\gamma}.$$

The conditions in Follmann and Lambert (1991) are:

- $\exists i \in I$: $\exists j \in J_i$: $N_{ij} \geq 2S - 1$
- $\mathrm{rk}(\mathbf{1}, \boldsymbol{Z}) = 1 + V$

For condition (1a) we need at least one individual with $N \geq 2S - 1$. In contrast to Follmann and Lambert (1991) the covariates for the fixed effects are allowed to vary, i.e. we require only

$$\exists i \in I : \sum_{j \in J_i} N_{ij} \geq 2S - 1.$$

The other condition which has to be checked is condition (2) which corresponds to the rank condition of Follmann and Lambert (1991). The difference to their conditions is due to the fact that we allow for a grouping variable where the component membership is fixed over different covariate points. Furthermore, we have established sufficient identifiability constraints for multinomial distributions while their constraints did only apply to binomial.

Choice models Conditional logit models are often applied as choice models in marketing research based on random utility theory (McFadden 1974). Using marketing mix variables like price and promotion as explanatory variables the probability of choosing a certain product is determined. In order to account for heterogeneity among the customers and hence capture the differences in their tastes finite mixture models can be used.

Kamakura and Russell (1989) estimated a finite mixture model of MNLs including only conditional logits based on the assumption that the price elasticity of the consumers varies over the consumer population but is fixed for each consumer over the different brands. This model can be specified within our framework by assuming that $U_2 = 0$ and $V = 0$. The conditional logit model with only varying effects is given by

$$\ln\left[\frac{\mathbb{P}(Y_r = k|\boldsymbol{x}_{1,k,i})}{\mathbb{P}(Y_r = K|\boldsymbol{x}_{1,k,i})}\right] = \boldsymbol{x}'_{1,k,i}\boldsymbol{\beta}^s_1 \qquad \forall k = 1, \dots, K.$$

Illustration on an artificial example

The identifiability of finite mixtures of multinomial logit models depends on the covariate matrix and the available repetitions per individual. This relationship is illustrated using an artificial example, where the number of different covariate points and the number of repetitions are varied such that the corresponding data generating process is either identifiable or not.

For simplicity of presentation a binomial variable with categories 0 and 1 is used as dependent variable and the regressors are the intercept and a univariate variable x. For both regressors varying effects are used. In the binomial logit model the probability of 1 is modelled which is also referred to as choice probability. The component weights are assumed to be constant over the individuals. The parameters of the mixture with two components are given by

$$\begin{aligned} \pi_1 &= 0.5 & \beta_1 &= (-2, \quad 4)' \\ \pi_2 &= 0.5 & \beta_2 &= (\quad 2, -4)' \end{aligned}$$

We use 2, 3 or 5 different covariate points x which are equidistantly spread across the interval $[0, 1]$. The mixture is not identifiable if there are only 2 covariate points available. In this case intra-component label switching is possible and the second solution is given by

$$\begin{aligned} \pi_1^{(2)} &= 0.5 & \beta_1^{(2)} &= (-2, 0)' \\ \pi_2^{(2)} &= 0.5 & \beta_2^{(2)} &= (\quad 2, 0)' \end{aligned}$$

The number of repetitions is fixed over all individuals and repetitions are only available for the same covariate point. The values for parameter N are 1, 2, 3, 5 and 10. The condition $N \geq 2S - 1$ implies that the mixture is not identifiable for $N \in \{1, 2\}$.

In Figure 3.2 the observed relative frequencies of choices of 1 for a random sample with 100 observations are given where the number of different x values is either 2 or 5 (i.e. $\#x \in \{2, 5\}$) and the number of repetitions for each observation is 1 or 10. The symmetry of the specified model is not entirely reflected in the observed values as the sample sizes are rather small.

The solid curves are the choice probabilities for each component of the true underlying model. For $N = 1$ the mixture is observational equivalent to a degenerate mixture with only one component. The probabilities of the degenerate model are given by the dashed line. Following the principle of parsimonity in the model fitting process, the degenerate mixture would be selected as solution. In addition to the degenerate mixture all mixtures with two components where the marginal choice probabilities over the two components are equal to those of the true model for each covariate point and

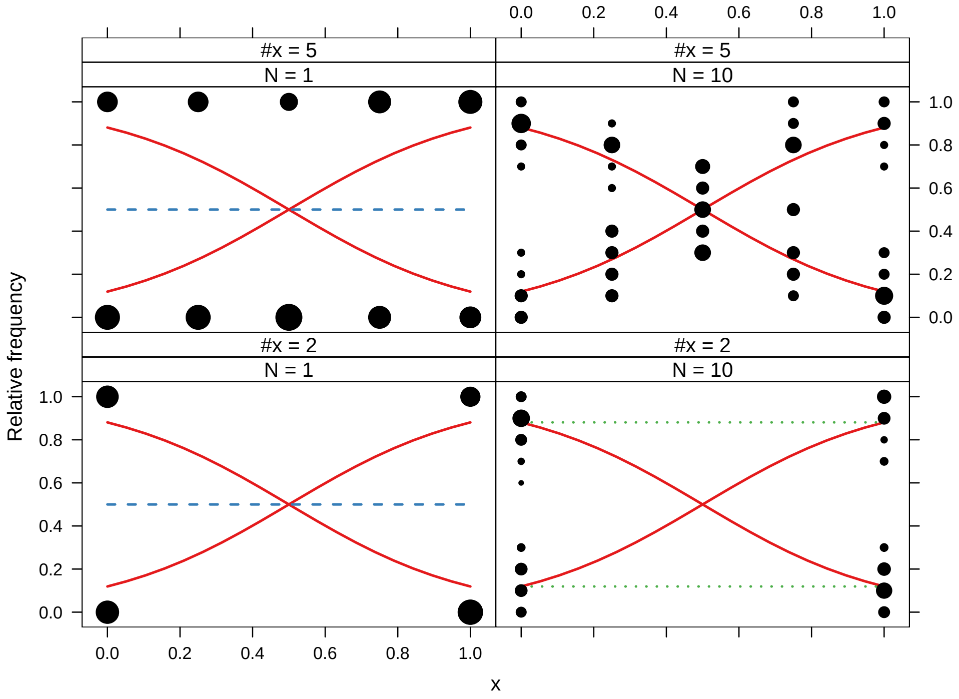


Figure 3.2: Observed values for the artificial example with $N = \{1, 10\}$ and where the number of different x values is 2 or 5.

where the relationship between the logit of the choice probabilities and x is linear are possible parameterizations of the same mixture.

For $N = 10$ it can be seen that intra-component label switching is possible if there are only two different covariate points available, whereas the mixture is identifiable for five different covariate points. The choice probabilities of each component of the observational equivalent mixture for two covariate points are given by the dotted lines.

100 samples with 100 observations each are drawn from the artificial model for all possible combinations of N and $\#x$. For the covariates we use a balanced design with an equal number of observations in each covariate point. To each sample we fit a mixture with two components using the EM algorithm. As stopping criterion the difference in log likelihood is used, i.e. the EM algorithm is stopped if the absolute relative difference between

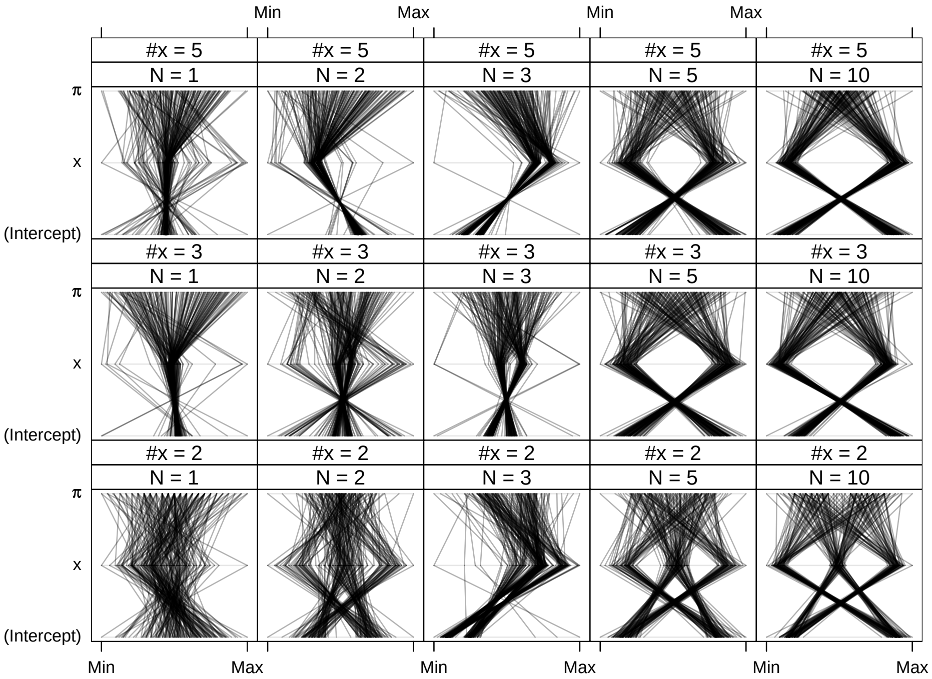


Figure 3.3: Parallel coordinate plots of the estimated parameters for 100 samples from the artificial example with different repetition parameters N and number of covariate points x. (Each coordinate has been independently rescaled to [0,1].)

the log likelihoods is smaller than $\epsilon = 10^{-8}$.

In order to avoid local maxima we repeat the EM algorithm 5 times with different random initializations and report only the best solution with respect to the log likelihood for each sample. The choice of 5 repetitions seems to be reasonable as only for 3% of the fitted models the best model is detected in the fifth repetition, while the best model is found in the first and second repetition in 86%.

In Figure 3.3 parallel coordinate plots are used to investigate the estimated parameters for all combinations of N and $\#x$. For an identifiable mixture with two components the parallel coordinate plot should contain two "bundles" corresponding to the parameters of each of the components.

For $N = 1$ there is only one large bundle observed because given a certain initialization the EM algorithm converges to one of the possible parameterizations which maximize the likelihood. For higher N it depends on $\#x$ if two or four bundles can be seen. Intra-component label switching occurs for $\#x = 2$ and four bundles can be seen. For $N > 2$ and $\#x > 2$ the two bundles can be distinguished which correspond to the parameter vectors for each component of the true model. If N increases the variability in the estimates decreases and the width of the bundles (visualizing variability of estimates) gets smaller.

The parallel coordinate plot has the advantage that label switching is of no concern as no unique labelling of the components is necessary. If we want to look at the estimated parameters separately for each component, a suitable relabelling of the components is necessary. As in our case the true model is known, suitable ordering constraints can be easily determined. Furthermore, an inspection of the parallel coordinate plot in Figure 3.3 suggests that if $N \geq 3$ an ordering constraint on the intercept can be imposed to induce a unique labelling. If the number of different covariate points is at least three we might also use an ordering constraint on the coefficient of x. An ordering constraint on the component weights is not suitable, as the components have equal component weights.

The effect of the different repetition values N and number of covariate points on the estimation of the coefficients of x is investigated using violin plots (Hintze and Nelson 1998) after imposing an ordering constraint on the intercept to induce an unique assignment of the parameters to one of the two components. This visualization method has the advantage that it is more robust to outliers than the parallel coordinate plot. If the coefficients for the intercept and x differ for the two components, which is true for our artificial model, the violin plot should indicate that the coefficients cluster around one mode for each component and that these modes are different for the two components. Overlapping modes indicate that there is no difference between the two components for the coefficients of x such that a fixed effect for x is more appropriate. If the coefficients scatter around several modes for each component, this signifies either that the ordering constraint did not induce a unique labelling or that intra-component label switching is present.

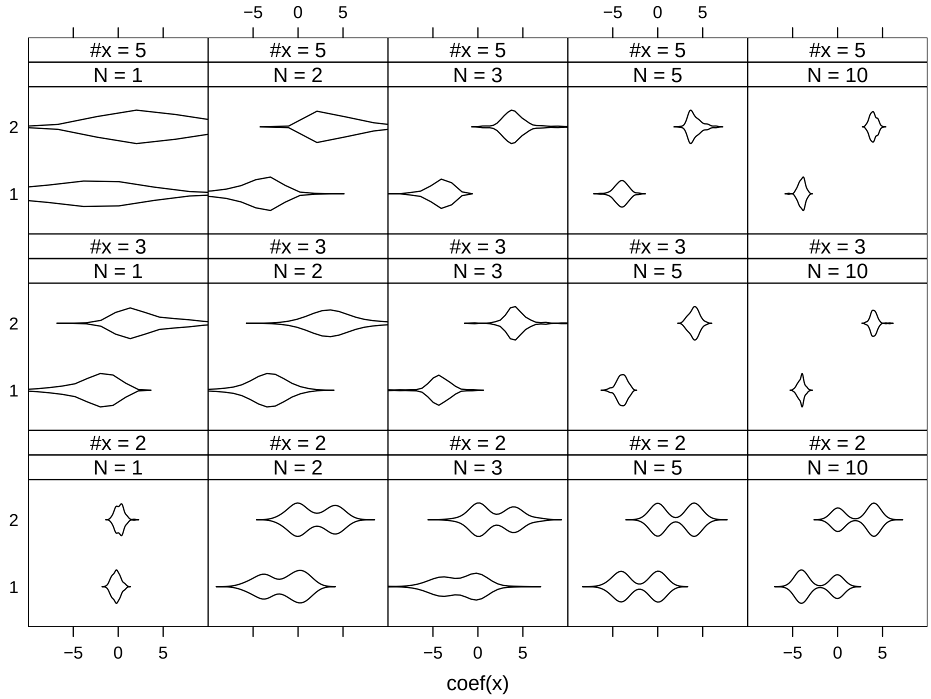


Figure 3.4: Estimated coefficients for x after imposing an ordering constraint on the intercept for 100 samples from the artificial example with different repetitions parameters N and number of covariate points x.

Figure 3.4 shows that the coefficients of x for the two components are similar for $N = 1$ independently from $\#x$. Especially for $\#x = 2$ it can be observed that the fitted mixture is degenerate which is caused by the random initialization and the fact that to assume a linear relationship is no restriction in the case of two covariate points.

For $N \geq 2$ and $\#x = 2$ the coefficients scatter around two different modes for each component due to intra-component label switching. For $N = 2$ and $\#x \geq 3$ a slight separation of the estimates for the coefficients of x for the two components can be observed, even if there is still quite a large overlap and the sufficient identifiability constraints do not apply. For $N \geq 3$ and $\#x \geq 3$ there are no longer identifiability problems, the coefficients of x are well separated by the constraint on the intercept and cluster only around their true values. An increase in the number of repetitions reduces the variability of the estimates.

In this artificial example no finite mixture can be estimated for $N = 1$. Furthermore, it can be seen that the quality of the parameter estimates improves if the number of repetitions increases. However, the identifiability of the parameter estimates is only guaranteed if the coverage condition is fulfilled in order to eliminate intra-component label switching.

3.4 Summary

Trivial identifiability problems such as label switching or overfitting are a problem for estimating finite mixtures or validating these models. These problems are a nuisance if they occur and have to be dealt with in this case. Generic identifiability problems can be present without being easily observable as they are, for example, only due to different separate parameterizations of the same model. The existence of these competing parameterizations is in general not easily detected during model estimation nor by model diagnostics using standard asymptotic theory which rely on local characteristics and can not be used to draw global conclusions.

Chapter 4

Model diagnostics using resampling techniques

Model diagnostics are an important tool for model identification. They can be used to check if the data set is likely to be a sample from the fitted model, i.e. the true data generating process (DGP) is well approximated by the model (Gelman 2004). In addition, model assumptions can be validated and the confidence which can be reasonably given to the parameter estimates can be determined by deriving standard deviations or posterior distributions.

In Section 4.1 a general framework for using resampling methods for model diagnostics of finite mixtures is formulated. This includes several existing techniques as special cases, especially determining the number of components (McLachlan 1987; Feng and McCulloch 1996), estimating standard errors of the parameters (Basford et al. 1997) and simple identifiability diagnostics (Grün and Leisch 2004), and provides a common basis for integration and software implementation of these tools. In addition, we propose several new techniques for analyzing the distribution of estimated mixture models. We demonstrate the advantages of using parallel coordinate plots where the estimated parameters of each component and each bootstrap replicate are used separately as data vectors. With this plot we can decide on possible model restrictions, a suitable ordering constraint, investigate the difference of the parameter values between the components, which can be used for imposing model restrictions, and analyze the identifiability of the fitted model. If the fitted mixture model is used for clustering the data, the stability of the clusters is of concern. This can be investigated by partitioning the data according to the posterior probabilities given the fitted models to each of the bootstrap samples. The congruence of the partitions between pairs of bootstrap samples is determined using agreement measures like the Rand index. All methods proposed have the advantage that they can be applied independently from the dimension of the parameter space.

Methods for detecting different (genuine) modes of a mixture model given a certain data set are presented in Section 4.2. This includes on the one hand methods to determine if genuine modes exist and on the other hand tools for determining the different modes and their attraction area. These tools are based on a suggestion in Stephens (2000b). In addition constrained clustering as outlined in Leisch and Grün (2006) can be used to simultaneously assign the estimated models to different modes and determine a suitable ordering of the components for each model. The imposed constraints only have to ensure that the component specific parameter estimates are assigned to different clusters.

4.1 Analyzing the model fit

When analyzing a fitted mixture model, we are interested in the distribution of the model $a \in \mathscr{A}_S$ which explains our data "best". Let $\mathcal{X}_n$ denote a sample with n independently identically distributed (iid) observations from the DGP. For a given data set the "best" model $a(\mathcal{X}_n)$ is usually defined as the one maximizing the likelihood, or in general the maximizer of an arbitrary performance measure $R(a, \mathcal{X}_n)$. More specifically, R could encode robust estimation or, if the model is used for prediction, R could minimize the maximum error instead of average error, see also Hothorn et al. (2005).

For a given data set and performance measure the optimal model $a(\mathcal{X}_n)$

is

$$a(\mathcal{X}_n) = \arg\sup_{a \in \mathscr{A}_S} R(a, \mathcal{X}_n)$$

Even for simple mixtures and maximum likelihood estimation no closed form solution for the optimal model exists and iterative parameter estimation algorithms like EM have to be used. Let $a(\mathcal{X}_n, \Theta_0)$ denote the local maximizer of R to which the EM algorithm converges if it is started in Θ_0. $a(\mathcal{X}_n)$ is usually approximated by the maximizer of the likelihood from several runs of the EM algorithm with different starting values Θ_0.

The fitted model $\hat{a}(\mathcal{X}_n)$ has a distribution $\mathcal{A}_S$ (depending on the DGP of $\mathcal{X}_n$) which can be explored for the analysis of model fit and identifiability by sampling either from the empirical distribution function $\hat{F}$ of $\mathcal{X}_n$ (non-parametric bootstrap) or from the estimated model $\hat{a}(\mathcal{X}_n)$ (parametric bootstrap).

A general framework for bootstrapping mixture models is

1. Estimate $\hat{a}(\mathcal{X}_n) \in \mathscr{A}_S$ and determine a corresponding parameterization $\hat{\Theta} \in \Omega$.

2. Sample B bootstrap samples $\mathcal{X}_n^b$ $(b = 1, \ldots, B)$ independently using either

   (a) empirical bootstrap: $\mathcal{X}_n^b \sim \hat{F}(\mathcal{X}_n)$, or

   (b) parametric bootstrap: $\mathcal{X}_n^b \sim \hat{a}(\mathcal{X}_n)$.

3. Fit models to the bootstrap samples with either

   (a) random initialization: $\hat{a}^b(\mathcal{X}_n^b) \in \mathscr{A}_{S_0}$ with possibly $S_0 \neq S$, or

   (b) initialization in $\hat{\Theta}$: $\hat{a}^b(\mathcal{X}_n^b, \hat{\Theta}) \in \mathscr{A}_{S_0}$ with $S_0 = S$.

4. Analyze the distribution $\mathcal{A}_{S_0}$ using sample $\hat{a}^b$, $b = 1, \ldots B$.

The set of models $\{\hat{a}^1, \ldots, \hat{a}^B\}$ can be used for exploring possible model restrictions, checking for identifiability, assessing the reliability of the estimated parameters, and analyzing the stability of the induced partitions.

Sequence 1-2(b)-3(b)-4, for instance, is used in Basford et al. (1997) for determining standard errors. McLachlan (1987) and Feng and McCulloch (1996) use 1-2(b)-3(a)-4 with $S_0 \in \{S, S+1\}$ and a likelihood ratio test statistic for determining the number of components. Below we will use 1-2(b)-3(a)-4 with $S_0 = S$ for assessment of identifiability problems, especially intra-component label switching.

Label switching is usually not of concern for $\hat{a}^b(\mathcal{X}_n, \hat{\Theta})$ in step 3(b) (McLachlan and Peel 2000) because most fitting algorithms will converge to an optimum that has the same permutation of components. On the other hand, label switching is a problem for $\hat{a}^b(\mathcal{X}_n)$ in step 3(a) if component specific analyses are to be made. A suitable relabelling of the components has to be determined if the solutions which are equivalent up to a permutation of the labels shall be matched before analyzing the component-specific parameters.

$\{\hat{a}^1, \ldots, \hat{a}^B\}$ is a set of iid observations which can be analyzed using standard techniques from exploratory and inferential statistics (Hothorn et al. 2005). When clustering data, two different criteria should usually be fulfilled if the suggested model fits the data:

- the components should be as different as possible from each other, and
- each component itself should be as homogeneous as possible.

The equality of parameters aggregated over all components can be tested with location tests or tests for unimodality (e.g. the dip test; Hartigan and Hartigan 1985) or for the number of modes (e.g. with kernel density estimates; Silverman 1981). This can be used to check, for example, if a restricted model, where a certain parameter is fixed over the components, is suitable. Estimating the mixture indirectly maximizes the differences between the components, so it is an even stronger indicator if the null hypothesis is not rejected. If the fitted model is identifiable the coefficients for each component follow an unimodal distribution. If intra-component label switching is present, they follow a mixture distribution (which can be multimodal).

Diagnostic plots can be used as an exploratory tool for analyzing identifiability and also to gain further insight into the estimated parameters and the difference between the components. Parallel coordinate plots (Wegman

1990) where the estimated parameters for each component are used separately as data vector are a suitable graphical representation, see Figure 4.1 for an example. Intra-component label switching between two solutions is indicated if the number of different "bundles" is twice the number of mixture components. "Bundles" can be discerned because the data vectors scatter around each of the true solutions. For the component weights and for the dispersion parameters (if present) the estimates scatter at maximum around S different points in this case. An advantage of this plot is that no relabelling of the components is necessary in advance. However, using different line types or colors for components after choosing a suitable relabelling might provide additional insights and help with interpretation.

A partition of the data can be determined with the models $\hat{a}^b(\mathcal{X}_n^b)$ by assigning the original data points to the component with the maximum a-posteriori probability. In order to determine the agreement between the partitions the Rand index corrected for agreement by chance (Hubert and Arabie 1985) is calculated for pairs of bootstrap samples. The use of the Rand index has the advantage that its value is independent of the specific labelling of the components. The Rand index values assess the stability of the induced partitions. Low stability is either an indicator for the presence of observations which are difficult to classify hard or intra-component label switching. If intra-component label switching is present the density estimate of the Rand index values has several modes. If the estimated parameters correspond to the same solution the Rand index will be close to 1, while it will be much smaller if they correspond to different solutions. The exact value depends on how many observations are combined differently.

4.1.1 Illustration using an artificial example

In the following we illustrate the proposed methods using the artificial example from Section 3.3.2 which is not identified due to intra-component label switching. We approximate $\mathcal{A}_K$ with the parametric bootstrap and $B = 200$ using a data set with 50 observations in each covariate point. The component weights used are $\pi_1 \in \{0.5, 0.6, 0.7\}$. For estimation of $a(\mathcal{X}_n)$ we let the

EM algorithm run 5 times with random starting values and choose the best solution with respect to the log likelihood.

In Figure 4.1 the parameters of $\hat{a}^b(\mathcal{X}_n)$, $b = 1, \ldots, 200$, are plotted. For equal component weights the components are similar for π and σ, whereas this is only the case for σ for unequal component weights. For the intercept two different, clearly separated groups can be distinguished. There are 4 separate groups for the coefficient of x for $\pi_1 \in \{0.5, 0.6\}$, but the coefficients of x scatter almost only around two points for $\pi_1 = 0.7$.

As we have only estimated mixtures with two components, the presence of 4 distinct bundles indicates identifiability problems for $\pi_1 \in \{0.5, 0.6\}$. Identifiability is still a problem for $\pi_1 = 0.6$ because the components have similar values for $x = 0$ and observations in this point are falsely assigned to the larger component if intra-component label switching is present. In Figure 4.1 it can be seen that the smaller coefficients of x are assigned to the smaller component for $\pi_1 = 0.6$. An ordering condition on the coefficient of x also gradually separates the component weight estimates.

Our set $\hat{a}^b(\mathcal{X}_n)$ of models is iid, hence we can use standard techniques from inferential statistics to test hypotheses. Table 4.1 shows the results of applying the dip test for unimodality for each of the parameters aggregated over the components. The null hypothesis of unimodality can not be rejected at a significance level of 0.05 for π and σ for equal component weights, whereas this is only the case for σ for unequal component weights. These variables are not suitable for imposing an ordering constraint. Furthermore, a homoscedastic model with equal variances in all components may be more suitable.

We now check for potential identifiability problems using an ordering constraint on the coefficients of x. The null hypothesis of unimodality is rejected for the intercept and x in both components for $\alpha = 0.05$ for equal and unequal component weights with $\pi_1 = 0.6$, while it is not rejected for the component weights with $\pi_1 = 0.7$. This indicates that intra-component label switching is present for the component weights $\pi_1 \in \{0.5, 0.6\}$, while the mixture distribution is identifiable for $\pi_1 = 0.7$. All conclusions drawn from applying the dip test agree well with the true DGP.

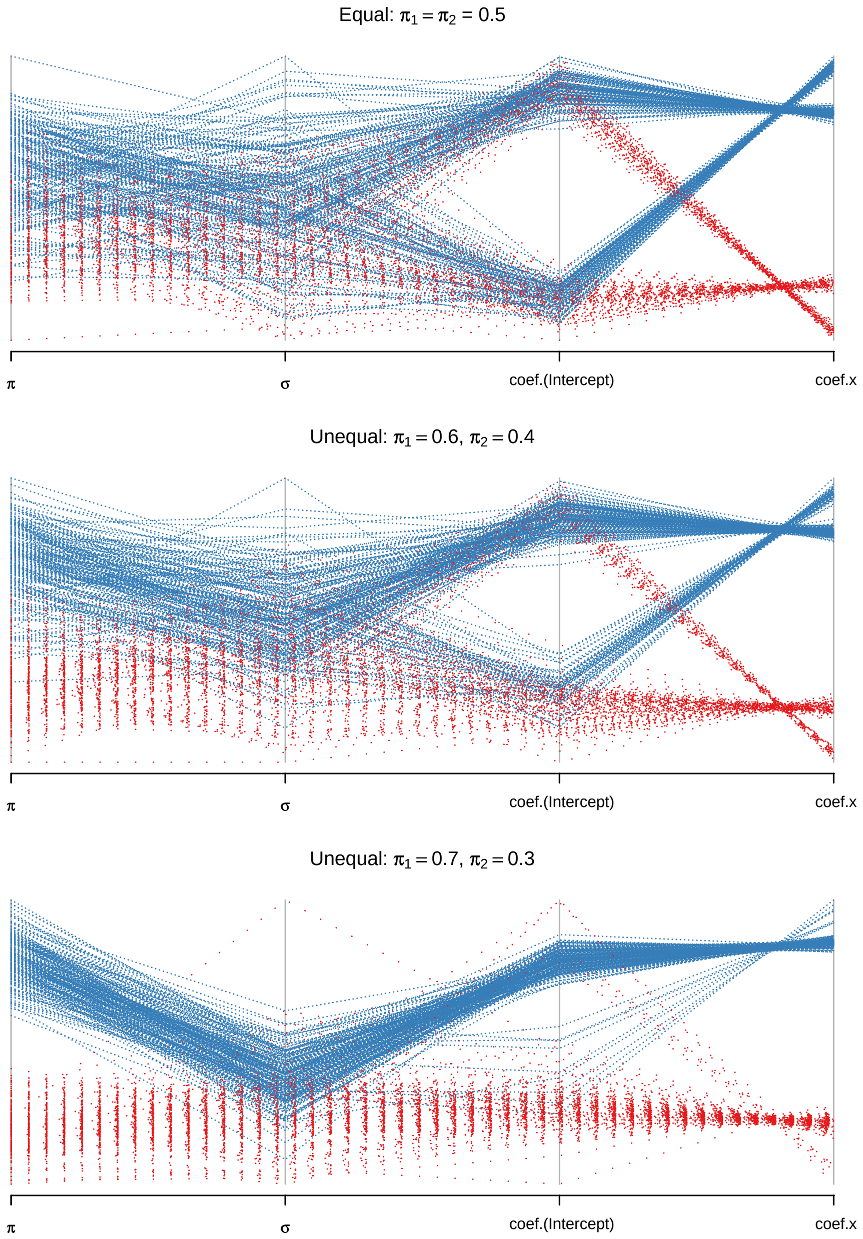


Figure 4.1: Parallel coordinate plot of the parameters of $\hat{a}^b$, $b = 1, \ldots, 200$ fitted with the parametric bootstrap for the artificial example with different component weights and an ordering constraint on x.

		π	Intercept	x	σ
0.5/0.5	Overall	0.01	**0.17**	**0.17**	0.01
	C_1	0.02	**0.14**	**0.13**	0.01
	C_2	0.02	**0.15**	**0.13**	0.02
0.6/0.4	Overall	**0.05**	**0.15**	**0.17**	0.01
	C_1	0.02	**0.09**	**0.08**	0.02
	C_2	0.02	**0.07**	**0.07**	0.02
0.7/0.3	Overall	**0.12**	**0.13**	**0.20**	0.02
	C_1	0.02	0.02	0.02	0.02
	C_2	0.02	0.02	0.02	0.02

Table 4.1: Test statistics of the dip test of unimodality for the artificial data set in the overall parameter distribution and within a component (C_i, $i = 1, 2$) after imposing an ordering constraint on the coefficients of x. Those test statistics which are significant with respect to $\alpha = 0.05$ are printed **bold**.

In addition to the parameter estimates we may also be interested in the partitions induced by the different models, i.e., how the data points are grouped. We compare each subsequent pair of partitions with the Rand index corrected for chance, resulting in $B/2$ Rand index values. A Rand index of 1 marks identical partitions, a Rand index of 0 marks agreement only by chance (given cluster sizes). Kernel density estimates of the Rand indices for the artificial example are shown in Figure 4.2. Especially for equal component weights most Rand indices are either 0 or 1. Intra-component label switching affects half of the observations, the Rand index of 0 corresponds to partition pairs which are induced by the two different modes of the likelihood.

4.1.2 Application

In the following the procedure for analyzing the model fit proposed in Section 4.1 is applied to two real world examples. Data from a clinical trial is used for fitting finite mixtures of Poisson regressions and this example illustrates how the proposed procedure helps in detecting identifiability problems. The second application uses higher-dimensional market basket data for latent class analysis. In this case the application of the resampling methods are

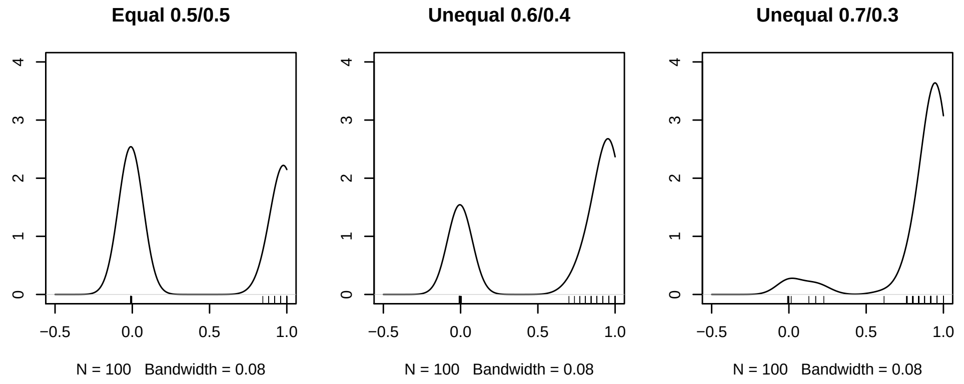


Figure 4.2: Kernel density estimates for the Rand indices corrected for chance of pairs of bootstrap samples for the artificial example with different component weights.

useful to validate the model and gain further insights into its characteristics.

Seizure data

In Wang et al. (1996) a Poisson mixture regression is fitted to data from a clinical trial where the effect of intravenous gammaglobulin on suppression of epileptic seizures is investigated. The data used were 140 observations from one treated patient, where treatment started on the 28$^{\text{th}}$ day. In the regression model three independent variables were included: treatment, trend and interaction treatment-trend. Treatment is a dummy variable indicating if the treatment period has already started. Furthermore, the number of parental observation hours per day were available and it is assumed that the number of epileptic seizures per observation hour follows a Poisson mixture distribution. The number of epileptic seizures per parental observation hour for each day are plotted in Figure 4.3. The fitted mixture distribution consisted of two components which can be interpreted as representing 'good' and 'bad' days of the patients.

The mixture model can be formulated by

$$H(y|\mathbf{x}, \Theta) = \pi_1 P(\lambda_1) + \pi_2 P(\lambda_2)$$

where $\lambda_k = e^{\mathbf{x}'\boldsymbol{\alpha}_k}$ for $k = 1, 2$ and $P(\lambda)$ is the Poisson distribution.

Identifiability Even though mixtures of Poisson distributions are generically identifiable (Teicher 1960), this holds not true for the model class specified in this example. The covariate points lie on two lines and therefore, the coverage condition is not fulfilled for a mixture with two components. Theoretic non-identifiability certainly only arises if there are equal component weights. In this case the components can be combined differently for the base and the treatment period and the following two solutions are valid:

Solution 1: $\boldsymbol{\alpha}_k^{(1)} = (\alpha_{kp}^{(1)})_{p=1,\ldots,4}$ for $k = 1, 2$

Solution 2: $\boldsymbol{\alpha}_k^{(2)} = (\alpha_{k1}^{(1)}, \alpha_{l2}^{(1)} + \alpha_{l1}^{(1)} - \alpha_{k1}^{(1)}, \alpha_{k3}^{(1)}, \alpha_{l4}^{(1)} + \alpha_{l3}^{(1)} - \alpha_{k3}^{(1)})$ for $(k, l) \in \{(1, 2); (2, 1)\}$

If the model is restricted by imposing a zero coefficient for treatment generic identifiability is still not given. However, this restriction imposes a continuity constraint between base and treatment period and therefore, non-identifiability can only arise if the mixture components cross at exactly this point.

The repeated observations for one single patient do not help in this case as it is assumed that the component memberships vary from day to day. Identifiability could be established if there were class labels for one observation from the base and one from the treatment period available.

Estimation When reestimating the model the results are the same (up to numerical differences due to different estimation software and control parameters) as in Wang et al. (1996). Our solution for $\boldsymbol{\alpha}_k$, $k = 1, 2$, with the corresponding standard deviations is given in Table 4.2. The component weight of the first component representing 'bad' days is 0.28.

We also use an equivalent model with different contrasts so that we can estimate the coefficient of the jump between the base and treatment period directly. If we use standard asymptotic theory (without adjusting the degrees of freedom for the estimation of the component weights) to determine the significance of the estimated coefficients the coefficients for the jump are

	Full model		Modified model		Rest. model	
Covariate	C_1	C_2	C_1	C_2	C_1	C_2
(Intercept)	2.84	2.07	1.49	1.17	1.43	1.10
	(0.23)	(0.09)	(0.08)	(0.06)	(0.06)	(0.05)
Treatment	1.30	7.43	-0.13	-0.15		
	(0.47)	(0.52)	(0.12)	(0.10)		
log(Day)	-0.41	-0.27	-0.41	-0.27	-0.46	-0.30
	(0.09)	(0.04)	(0.09)	(0.04)	(0.07)	(0.03)
Interaction	-0.43	-2.28	-0.43	-2.28	-0.46	-2.34
	(0.13)	(0.14)	(0.13)	(0.14)	(0.13)	(0.13)
π	0.28	0.72	0.28	0.72	0.28	0.72
log likelihood	-376.18		-376.18		-377.36	
AIC	770.35		770.35		768.73	
BIC	796.83		796.83		789.32	

Table 4.2: Parameter estimates (standard errors) for the model proposed in Wang et al. (1996) with the same and different contrasts and for the restricted model.

insignificant for both components. The restricted model with no jump between base and treatment period has a better model fit with respect to the AIC and BIC criteria and a likelihood ratio test also favors the more parsimonious model ($\chi^2 = 2.38$, p-value $= 0.30$). In contrast to the test for the number of components the likelihood ratio test can be used for the nested models as the asymptotic theory is applicable given that the null hypothesis is not at the margin of the parameter space. When imposing the continuity constraint on the model the remaining coefficients change only slightly and the component weights remain the same. All numerical results are given in Table 4.2, the fitted values for the models together with the original data are plotted in Figure 4.3.

Analyzing the model fit 200 parameteric bootstrap samples are drawn from both fitted models and a mixture model is fitted to each of them using the EM algorithm with random initialization and choosing the best solution of 5 repetitions. In Figure 4.4 the fitted values for the 200 parametric bootstrap samples are plotted for component 1, which is the component with the

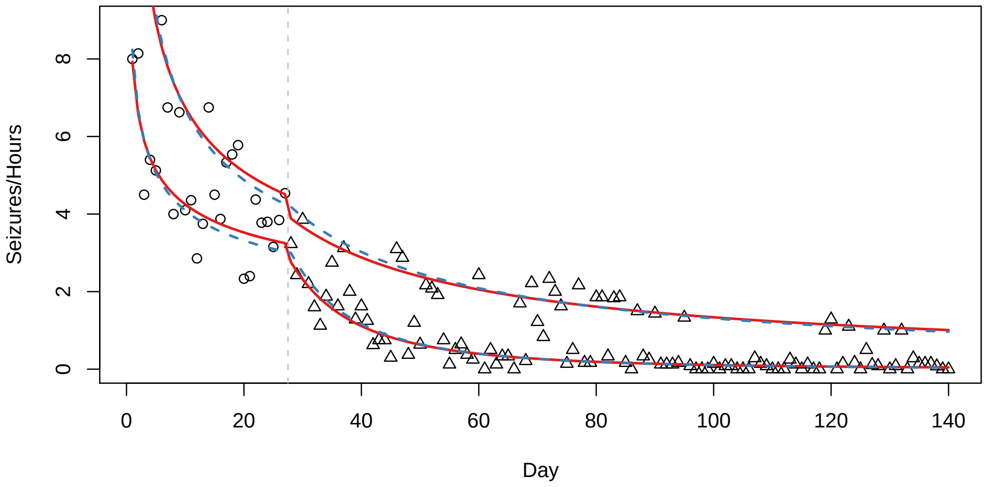


Figure 4.3: Seizure data with the fitted values for the Wang et al. model (solid line) and for the restricted model (dashed line). The plotting character for the observed values in the base period is a circle and for those in the treatment period a triangle.

smaller component weight. On the left the results for the model proposed in Wang et al. (1996) are shown and on the right for the restricted model. For the full model it can be seen that the fitted values for the bootstrap samples spread rather close around those from the original data in the treatment period. For the base period the values fitted to the bootstrap samples scatter around the fitted values of both components for the original data set, which signifies that intra-component label switching is present. Theoretically intra-component label switching is only possible in mixtures where the coverage condition is violated and where the component weights are equal. However, it seems that intra-component label switching occurs even though the estimated group sizes differ because there are only few observations in the base period and it is not possible to estimate the component weights well for the mixture distribution in the base period. In the restricted model (Figure 4.4 right) the intra-component label switching effect is almost not present any more, as in this model there is no jump between base and treatment pe-

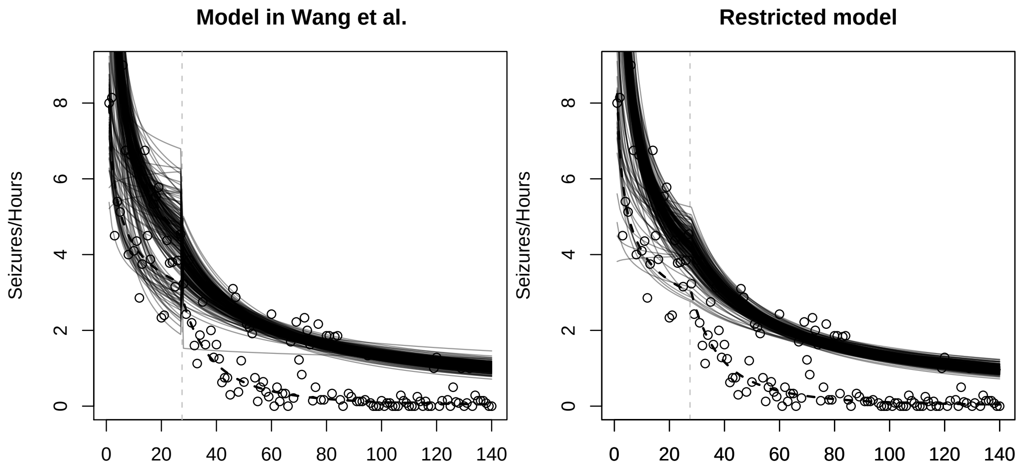


Figure 4.4: Fitted values for component 1 under the ordering constraint $\pi_1 < \pi_2$ for the 200 parametric bootstrap samples from the Wang et al. model (left) and the restricted model (right).

riod possible. The results for the second component are similar and given in Figure 4.5.

In Figure 4.6 the parallel coordinate plots of the parameters of $\hat{a}^b(\mathcal{X}_N^b)$ $(b = 1, \ldots, B)$ for the model in Wang et al. (1996) and the restricted model are given. The coefficients assigned to component 1 according to an ascending ordering constraint on π are drawn dashed and the others dotted. In both cases it can clearly be seen that the component weights and the coefficients where `Treatment` is involved differ strongly for the two components. They are rather close for the main effect of `Day` for the full model. While only few occurrences of intra-component label switching can be observed for the restricted model, a considerable amount is present for the original model proposed in Wang et al. (1996). In contrast to the artificial example there are only two instead of four clusters present for the coefficients involving `Treatment` as the values differ only slightly but are switched for the two components of the two parametric representations.

The results of the application of the dip test are given in Table 4.3. For the model proposed in Wang et al. (1996) the dip test indicates that the

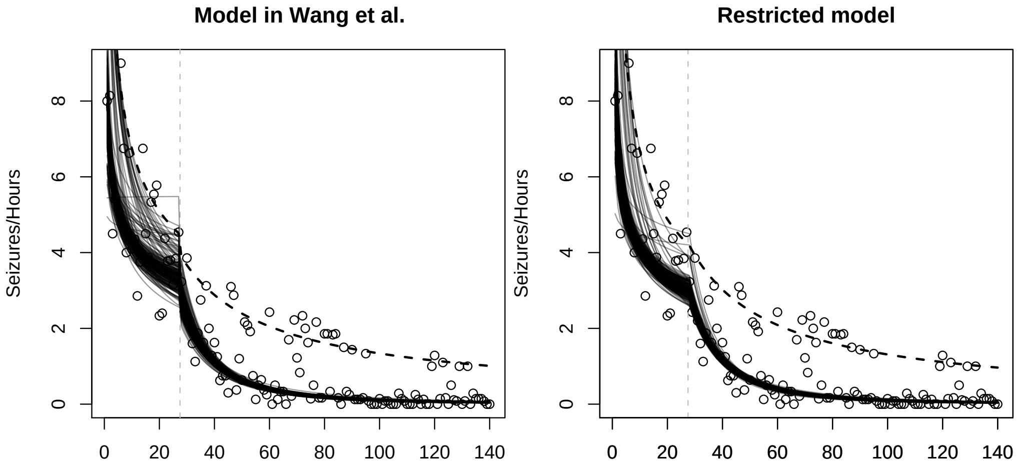
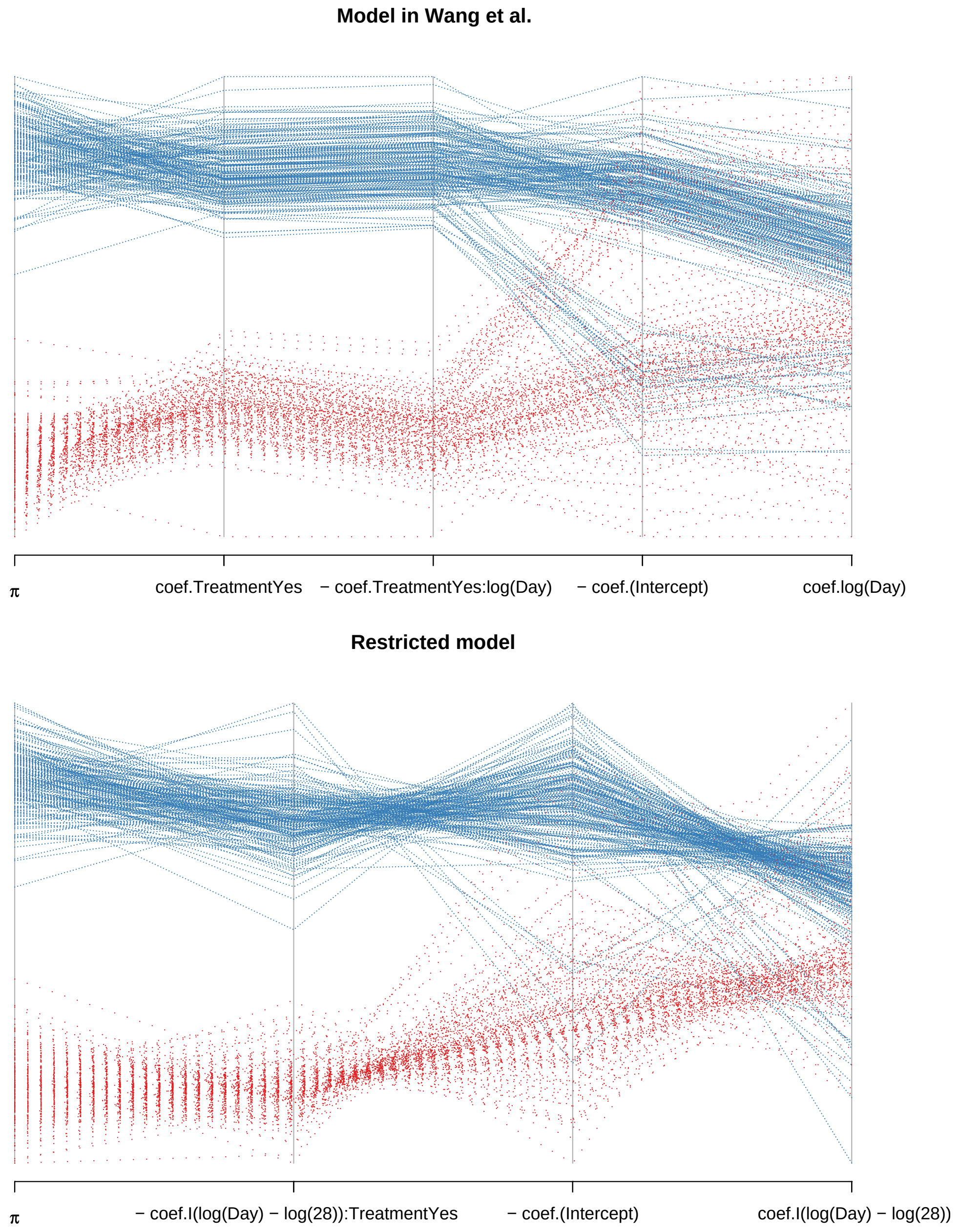


Figure 4.5: Fitted values for component 2 under the ordering constraint $\pi_1 < \pi_2$ for the 200 parametric bootstrap samples from the Wang et al. model (left) and the restricted model (right).

hypothesis of unimodality can not be rejected for $\alpha = 0.05$ for the combined values of the coefficient of `Day`. This suggests that a model where the parameter for `Day` is fixed over the two components might be appropriate. For the separate analysis of the components we use an ascending ordering constraint on the component weights. The dip test suggests that multimodality is present (with $\alpha = 0.1$) for the coefficient of the intercept of component 1. For the restricted model the hypothesis of unimodality is rejected for each parameter and no indication of multimodality is found in the separate analysis of the components.

In Figure 4.7 the density estimates of the Rand indices corrected for chance for the 200 bootstrap samples are given. It can be seen that the density for the model proposed by Wang et al. is bimodal. This is caused by intra-component label switching. For the restricted model it can be seen that some solutions give slightly different partitions but there can be no second real mode identified.

The methods and tools proposed indicate that intra-component label switching is present for the model from Wang et al. (1996) and it can be


Model in Wang et al.

Restricted model

Figure 4.6: Diagnostic plot of the EM solutions for 200 parametric bootstrap samples with random starting values for the model proposed in Wang et al. (1996) and the restricted model using an ordering constraint on the component weights.

	Model in Wang et al.			Rest. model		
Variable	Overall	C_1	C_2	Overall	C_1	C_2
π	**0.15**	0.02	0.02	**0.15**	0.02	0.02
(Intercept)	**0.08**	*0.04*	0.02	**0.07**	0.02	0.02
Treatment	**0.14**	0.02	0.02	-	-	-
log(Day)	0.01	0.02	0.02	**0.03**	0.02	0.02
Interaction term	**0.15**	0.02	0.02	**0.15**	0.02	0.02

Table 4.3: Test statistics of the dip test of unimodality for the seizure data set and the two different models in the overall parameter distribution and within a component (C_i, $i = 1, 2$) after imposing an ordering constraint on π. Those test statistics which are significant with respect to $\alpha = 0.05$ are printed **bold** and with respect to $\alpha = 0.1$ *italic*.

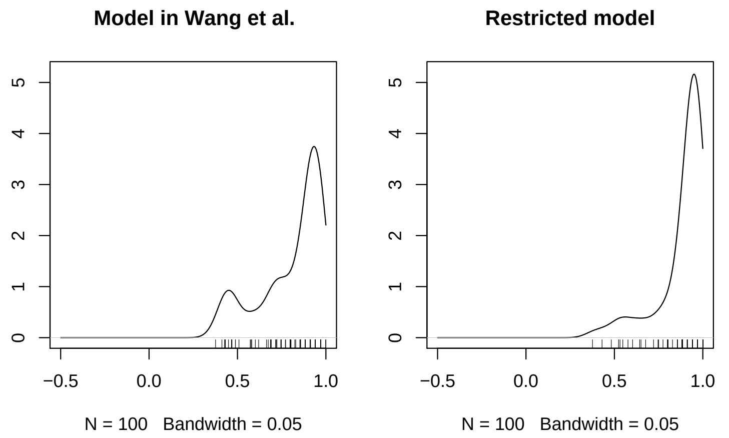


Figure 4.7: Diagnostic plot using Rand indices corrected for chance for pairs of bootstrap samples for the full model and the restricted model.

concluded that this mixture model is close to (empirical) non-identifiability. This problem can be alleviated by imposing a continuity constraint between base and treatment period. These findings confirm the conclusions drawn from the more intuitive plot of the fitted values. However, this plot is only possible because of the 2-dimensional relationship between dependent and independent variables, while the other methods are less dependent on the dimension of the covariate space.

Market basket data

The data used in this section is the ZUMA subsample of the 1995 Consumer-Scan Household Panel (GfK, Nürnberg) data set. The data set includes 4424 households for which consumption data were collected throughout 1995. For more specific information about the consumer panel data see Papastefanou (2001). For clarity of presentation we restrict the following analysis to one shopping basket for each household on 10 popular product groups. The data are binary indicators if the respective product group has been bought or not.

The task here is market segmentation, i.e., we want to find households with similar shopping behaviour. For model-based clustering we fit a finite mixture of binomial distributions under the assumption of local independence: The probabilities of buying each of the 10 product groups is independent given cluster membership. Hence, the parameter vector for each component is a vector of 10 probabilities. First we fit models with 1 to 8 components using again the best out of 5 EM replications for each number of components.

Model selection with the BIC recommends the solution with 3 components, the purchase probabilities for each of the components and the respective component weights are given in Table 4.4. The standard deviations in Table 4.4 are derived with standard asymptotic theory using the inverse of the expected information matrix (de Menezes 1999). Full rank of the expected information matrix which can be checked by the estimated standard deviations signifies that the model is locally identifiable (Goodman 1974).

The first component consists of larger baskets with an average size of 2.95

Covariate	C_1	C_2	C_3
pet food/cat litter	0.13 (0.02)	0.07 (0.01)	0.07 (0.01)
milk	0.59 (0.04)	0.17 (0.04)	0.34 (0.02)
mineral water	0.37 (0.04)	0.34 (0.06)	0.01 (0.04)
toilet paper	0.22 (0.03)	0.06 (0.01)	0.06 (0.01)
toothpaste	0.13 (0.02)	0.04 (0.01)	0.07 (0.01)
beer	0.20 (0.03)	0.22 (0.05)	0.00 (0.02)
yogurt	0.48 (0.04)	0.09 (0.03)	0.19 (0.01)
hard cheese	0.33 (0.04)	0.05 (0.02)	0.14 (0.01)
coffee	0.30 (0.03)	0.11 (0.02)	0.13 (0.01)
tea	0.19 (0.02)	0.05 (0.01)	0.07 (0.01)
π	0.15 (0.03)	0.27 (0.09)	0.58 (0.10)

Table 4.4: Parameter estimates (standard errors) for the 3-component (C_i, $i = 1, 2, 3$) finite mixture fitted to the ZUMA data set.

product groups, and all product groups have a comparatively high purchase probability. The other two components contain small baskets with an average size of 1.21 and 1.07 product groups, respectively. The second component has high probabilities for mineral water and beer only, the third component has low probabilities for all product groups.

For the analysis of model fit we use 200 parametric bootstrap samples and fit models to them using the best of 5 repetitions of the EM algorithm with random initialization. The models fitted on bootstrap data are used to confirm model validity, as standard asymptotic theory might not work well on sparse data with low expected frequencies (von Davier 1997). The estimated p-values for the likelihood ratio test is 0.62 and for the Pearson χ^2-test is 0.26. Both goodness-of-fit tests indicate that model validity is no problem for our fitted model.

Figure 4.8 shows a parallel coordinate plot of the parameter estimates of the bootstrap samples. It can clearly be seen that at least two components differ for each parameter. Furthermore, component 1 containing the large baskets is nearly always on top. Component 2 has always smaller purchase probabilities than component 3 except for mineral water and beer. Formal significance tests for multimodality of the parameter estimates using the dip

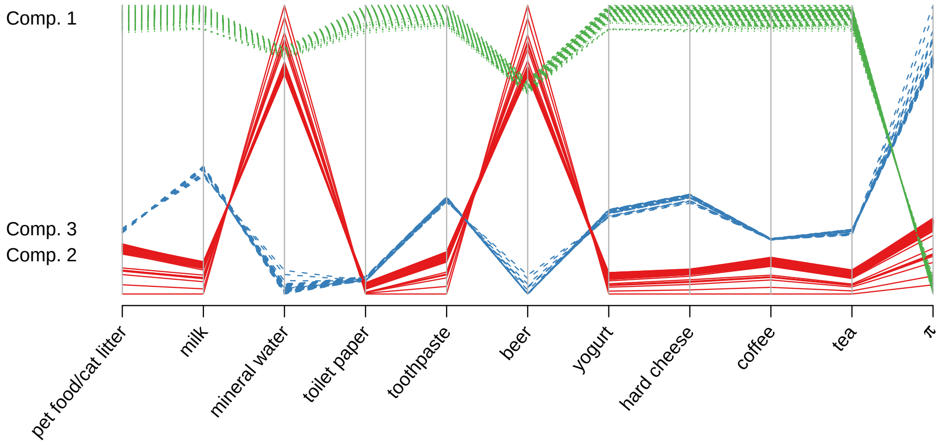


Figure 4.8: Diagnostic plot of the EM solutions for 200 parametric bootstrap samples using an ordering constraint on "milk".

test confirm that the components are well separated (as expected given Figure 4.8) with a minimal dip test statistic of 0.15 which is highly significant given the sample size. Finally, Rand indices corrected for chance comparing pairs of partitions shown in Figure 4.9 confirm that basically the same solution is found in every bootstrap replicate.

Fitting a mixture with too many components All model diagnostics for the 3-component mixture model indicate stability of parameter estimates and an identifiable model, hence from a statistician's point of view the model fits the data well. However, from a marketing researcher's point of view the fitted model is not ideal, because two of the three components do not differentiate between the product groups at all: both component 1 and 3 buy "everything", only with different probability (all high vs. all low). Only component 2 is a real niche market (mineral water, beer) that could be used effectively in an advertising campaign.

Hence, to identify more potential market segments we need a model with more components. We do no longer assume the existence of natural segments, and use clustering procedures as a construction task rather than a search mission for natural phenomena, i.e. we partition the data into "arbi-

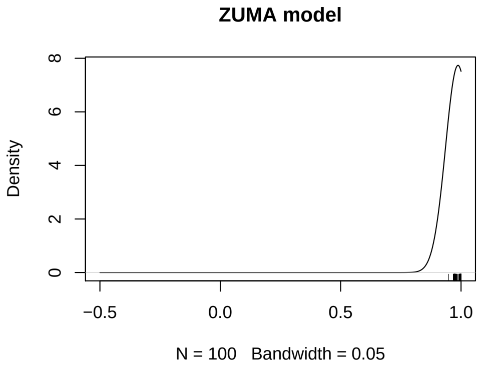


Figure 4.9: Diagnostic plot using Rand indices corrected for chance for pairs of bootstrap samples for the ZUMA data set.

trary clusters" instead of "natural clusters" (Kruskal 1977). This strategy is often necessary in market segmentation as distinct segments rarely exist in empirical data sets (Mazanec et al. 1997).

In the following we fit a mixture with 8 components. In order to analyse the stability of this solution, 200 empirical bootstrap samples are drawn and a mixture with 8 components is fitted to each of them where the best solution of 5 repetitions is chosen. The EM algorithm did only converge for 62% of the samples to a mixture of 8 components and in the other cases the solution had only 5 (0.5% of the samples), 6 (5.5% of the samples) or 7 (32% of the samples) components, as components with a component weight smaller than 5% are omitted during the EM algorithm.

Figure 4.10 shows the parameter estimates. As the estimated parameters overlap for each of the variables for different components, any ordering constraint fails to return a suitable labelling. Therefore, we use constrained k-means clustering on the estimated parameters to determine a suitable labelling (see Section 4.2.3 or Leisch and Grün 2006). This allowed us to also include those models where less than 8 components were fitted. Components 4 and 7 seem to be stable clusters, while the other components correspond to arbitrary clusters, which artificially split the observations into groups. Component 7 is again the mineral water & beer segment which we already know

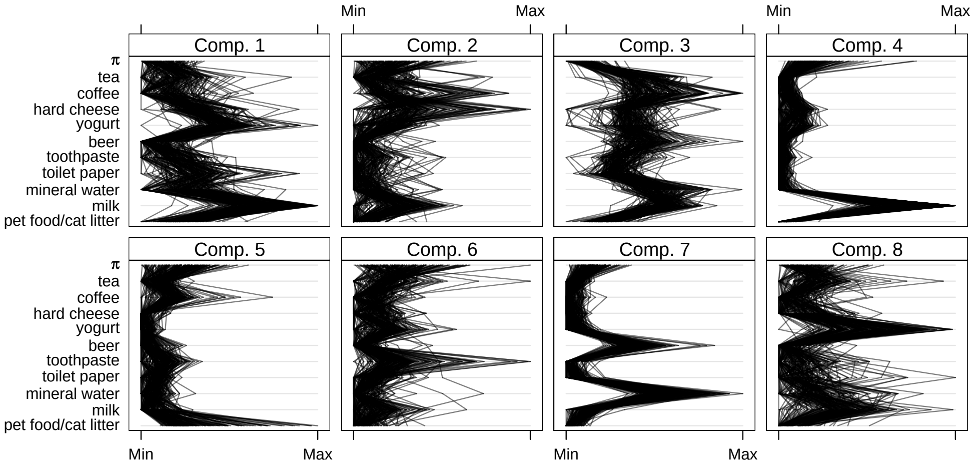


Figure 4.10: Parallel coordinate plot of the parameter estimates for the empirical bootstrap samples. The components are labelled using constrained k-means clustering.

from the 3-component model.

For further analysis the log likelihoods of the different models are compared as well as the induced clusterings. As can be seen in Figure 4.11 the likelihoods are comparable for all models. The class agreement levels of the induced clusterings shows Rand indices corrected for chance scattering around 0.5, thus agreement is larger than merely by chance, but considerably lower than for the 3-component model.

The empirical bootstrap can be efficiently used to determine which of the components of a fitted mixture model correspond to natural clusters and which are artificial clusters. This is a valuable information for a market researcher when developing a suitable marketing strategy: one should be aware that only components 4 and 7 are stable components, while the other components correspond to artificially constructed market segments.

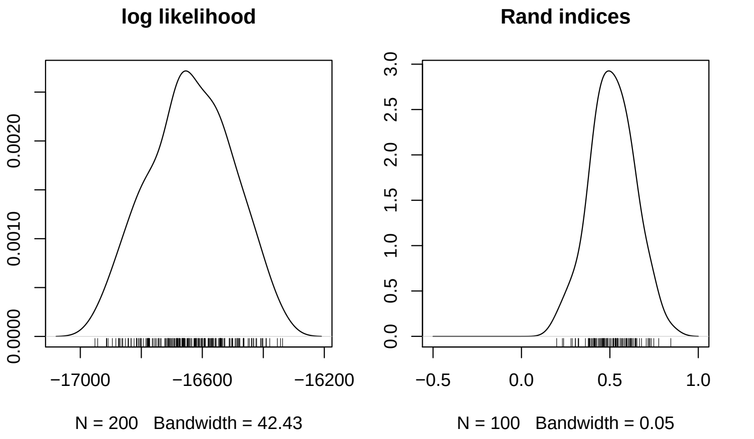


Figure 4.11: Density estimates of log likelihoods and Rand indices corrected for chance for pairs of bootstrap samples.

4.2 Genuine multimodality

4.2.1 Definition

In the following we are interested in competing parameterizations for the same model or to describe the underlying DGP which are not equivalent in the parameter space $\tilde{\Omega}$ of the equivalence classes induced by label permutation. The presence of these genuine competing parameterizations is referred to as *genuine multimodality*.

Definition 2. *The distribution $\mathcal{O}$ of the parameters $\Theta \in \Omega$ is called* genuinely multimodal *if it holds for the set of modes $\mathcal{M}$ of $\mathcal{O}$ that*

$$\exists \Theta_1, \Theta_2 \in \mathcal{M} : \Theta_1 \neq \nu(\Theta_2) \qquad \forall \nu \in Perm(S)$$

A mode of a probability distribution is defined as a local maximum in the associated probability density function (Minnotte 1997). The distribution $\mathcal{O}$ is called *genuinely unimodal* if the opposite holds for the set of modes $\mathcal{M}$.

An equivalent definition where the admissible parameter space Ω has been suitably restricted to $\tilde{\Omega}$ is given by:

Definition 3. *A distribution $\mathcal{O}$ of the parameters $\Theta \in \tilde{\Omega}$ is called* genuinely unimodal *if it holds that the set of modes $\mathcal{M}$ of $\mathcal{O}$ is a singleton.*

With this definition multimodality is a characteristic of the underlying DGP and the specified model, while in the case where multimodality is defined in dependency of the likelihood this is in fact only a characteristic of the given data set $\mathcal{X}_n$ and the specified model. Because in general a generalization from the given data set $\mathcal{X}_n$ on the DGP is desired focusing on the distribution of the fitted parameters depending on the DGP seems to be appropriate. Given a single data set $\mathcal{X}_n$ the distribution $\mathcal{O}$ can be approximated given the fitted model $\hat{a}(\mathcal{X}_n)$ together with the application of resampling methods. This is done by:

1. Determine $\hat{a}(\mathcal{X}_n) \in \mathscr{A}_S$ and a corresponding parameterization $\hat{\Theta} \in \Omega$, e.g. with the EM algorithm using the best solution of several random initializations.

2. Sample B bootstrap samples $\mathcal{X}_n^b$ independently for $b = 1, \ldots, B$ with the parametric bootstrap, i.e. $\mathcal{X}_n^b \sim \hat{a}(\mathcal{X}_n)$.

3. Fit models to the bootstrap samples, i.e. determine $\hat{a}(\mathcal{X}_n^b) \in \mathscr{A}_S$ using the EM algorithm with several random initializations.

4. Analyze the parameterizations $\hat{\Theta}_b$ of the bootstrap models $\hat{a}(\mathcal{X}_n^b)$ which imply an approximation of the distribution $\mathcal{O}$.

The use of the parametric bootstrap is based on the assumption that an appropriate model has been fitted and that the characteristics of the fitted model are analyzed, i.e. we are interested in model diagnostics of the fitted model. The empirical bootstrap should be used instead if there can not be much confidence assigned to the model choice.

4.2.2 Checking for genuine multimodality

Testing procedure

Under the null hypothesis of genuine unimodality the distribution of the component specific parameter vectors ϑ_s^b should be unimodal after suitable relabelling such that $\forall b = 1, \dots, B$: $\hat{\Theta}_b \in \tilde{\Omega}$. The relabelled components for the parameters space $\tilde{\Omega}$ may be determined using the methods proposed in a Bayesian context, as e.g. imposing an ordering constraint or the relabelling algorithms suggested by Stephens (2000b) or Frühwirth-Schnatter (2001). The testing procedure is given by:

1. Determine a suitable parameter space $\tilde{\Omega}$ and relabel the component specific bootstrap samples ϑ_s^b accordingly.

2. Test each component specific parameter for unimodality, e.g. using the dip test (Hartigan and Hartigan 1985).

3. Adjust the p-values for multiple testing, e.g. using the method proposed in Holm (1979), and determine for a given significance level α if the null hypothesis of genuine unimodality is rejected.

The null hypothesis of unimodality is also rejected, if the relabelling algorithm did not succeed in determining a unique labelling or if the EM algorithm did not converge to a suitable root for the bootstrap samples. The convergence of the EM algorithm to a suitable root can be ensured by repeating the fitting with (different) random initializations and choosing the best root with respect to the likelihood. Different strategies to initialize the EM algorithm and ensure the detection of a suitable root have been proposed, e.g. by Biernacki et al. (2003). Another possibility is to compare the solution found to the one derived using initialization in the fitted model and choose the better one. However, this increases the chance that the model fitted to a bootstrap sample corresponds to the maximum of the likelihood close to the original model.

The dip test for unimodality uses the maximal difference between the empirical distribution function and the unimodal distribution function that minimizes this maximal difference as test statistic. In order to evaluate the dip test the distribution of the test statistic under the null hypothesis has to be determined. Hartigan and Hartigan (1985) sample under the null hypothesis of a uniform distribution as this is the asymptotically least favorable unimodal distribution. However, this makes the test rather conservative and therefore, Cheng and Hall (1998) choose a calibration distribution depending on the form of the unique mode. Another possibility is suggested by Tantrum et al. (2003) who sample from the unimodal distribution which is closest to the empirical cumulative distribution function.

In the following we use the approach proposed by Cheng and Hall (1998) but instead of estimating the form of the mode from the available data we assume that prior knowledge is available. We sample from a Gaussian distribution, because the bootstrap samples are expected under the null hypothesis to follow this distribution. However, to account for possible convergence problems of the EM algorithm it could also be considered to use a distribution with heavier tails, as e.g. a t-distribution.

Exploratory analysis

In addition to the testing procedure which can be made in an automatic way, also an exploratory analysis of the bootstrap samples can be applied in order to investigate the distribution $\mathcal{O}$. Genuine multimodality leads to differences between the reliability results of standard asymptotic theory, which are based on local characteristics, and those using resampling methods where the complete parameter space is explored. For each component specific parameter vector ϑ_s the confidence band CB_s^α with significance level α is derived using the standard deviations of the parameters determined with standard asymptotic theory and normal approximation. In order to determine a $\alpha\%$ confidence band Bonferroni correction is used to determine the confidence intervals for each component specific parameter. For each bootstrap sample $b = 1, \dots, B$ it is checked for each component $s_0 \in \{1, \dots, S\}$ if $\exists s \in \{1, \dots, S\}$: $\vartheta_{s_0}^b \in \mathrm{CB}_s^\alpha$ and they are accordingly assigned to S classes together with a class containing the remaining parameters. This procedure implicitly assumes that the pairwise confidence bands do not overlap for at least one parameter, which is in fact not a restriction as an overlap for all parameters indicates that a mixture with less components is suitable.

Under the assumption of genuine unimodality αB component specific parameters of the bootstrap samples should be assigned to each of the confidence bands. Therefore, a large discrepancy between the observed and expected values is an indication for the presence of genuine multimodality. Other possible reasons for this difference are a bad approximation of the true confidence bands or a failure of the EM algorithm to converge to a suitable root for the bootstrap samples. The validity of the asymptotic results can be verified by using the bootstrap approach with initialization in the solution. This in general eliminates convergence problems of the EM algorithm and the detection of other global maxima. In addition the same methods as proposed for the testing procedure can be used to check if the EM algorithm converged to a suitable root.

The bootstrap results can be visualized using parallel coordinate plots of the component specific parameters ϑ_s^b together with the confidence bands CB_s^α, where the coloring indicates the class assignment. This allows to check if there exist bundles of bootstrap parameters ϑ_s^b for those parameters which have not been assigned to a confidence band. The presence of bundles indicates identifiability problems, as the bootstrap results cluster around another parameter value which might be given by a different mode. Figure 4.13 shows this diagnostic plot for a finite mixture where identifiability problems are present.

In addition the testing procedure can be complemented with an exploratory analysis of the results by visualizing the component specific distributions using kernel density estimates and inspecting them for multimodality.

The suggested test and the exploratory tools should be used to check the fitted model and be part of the usually applied model diagnostics. If the null hypothesis is not rejected, this increases the confidence in the fitted model. However, as there are several possible reasons for rejecting the null hypothesis, further investigations of the fitted model are necessary in this case.

4.2.3 Determining the modes

If genuine multimodality is detected, the next step is to determine the parameters for each of the modes, because this gives us the competing parameterizations of the underlying DGP. Given some expert knowledge it might be possible to eliminate one of the solutions or to choose the most appropriate one. Furthermore, a restricted model can be considered in order to resolve the identifiability problems. A comparison of the different parameterizations reveals, if there are components of the mixture which are uniquely determined as they are present in all modes (cp. component 3 in the example in Section 4.2.4).

A straight-forward approach to solve this problem is to cluster the complete parameter vectors $\nu_b(\Theta_b)$ after suitable relabelling in order to determine the number of modes and assign each parameter vector to one of the modes. This procedure relies on the assumption that a suitable parameter space $\tilde{\Omega}$ has been determined which induced a unique labelling of the components. However this is hard to verify as multimodality of the component specific estimates occurs in any case due to the genuine multimodality.

Another approach is to use relabelling algorithms while allowing for multimodality (Stephens 2000b). This approach has the advantage that in one step the assignment to the different modes and a suitable labelling given the assignment to a mode are determined. A disadvantage is that the proposed optimization methods do only converge to a local optimum. Therefore, it might be necessary to repeat the algorithm several times with different initializations in order to be able to detect the global optimum. An heuristic to determine good initial values might improve the performance. In addition, a combination of relabelling algorithms and clustering methods can be used as exploratory data analysis tools to determine the number of different modes.

Exploratory analysis using clustering methods

Standard clustering methods, as e.g. k-means or hierarchical clustering, are useful tools to investigate the number of different modes of $\mathcal{O}$. The choice of input data depends on the specific model fitting aim as either the component specific parameter estimates ϑ_s^b or the component specific a-posteriori probabilities $p_{is}(\Theta_b)$ can be used. As the parameters ϑ_s^b are in general on different scales, we standardize them before clustering. Furthermore, we include only those parameters, where the aggregated component specific estimates follow a multimodal distribution, to improve the performance of the clustering algorithms. Choosing the appropriate number of clusters is a problem which has not been completely resolved in cluster analysis yet but several methods and heuristics have been proposed and can be applied to facilitate a decision. We use a graphical tool: the scree plots of the sum of within cluster distances.

By clustering the component specific estimates the parameters of each bootstrap sample should be assigned to S different clusters. In addition, each mode consists in general of S clusters. By combining the cluster assignments with the information which estimates belong to the same bootstrap sample, it is checked if the estimates from the same bootstrap sample belong to different clusters. An assignment to the same cluster indicates that the difference between the components is not sufficient to ensure assignment to different clusters, i.e. this indicates that a mixture with less components might be suitable. Furthermore, an estimate for the number of modes is determined by investigating how many different combinations of clusters occur for the bootstrap samples.

As an assignment to different modes can only be made if the estimates from the same bootstrap sample are assigned to different clusters, this can be enforced by using constrained k-means clustering (Wagstaff et al. 2001; Leisch and Grün 2006). The input data $X_{B,S}$ is given by $\{x_{b,s} : b = 1, \ldots, B; s = 1, \ldots, S\}$, where $x_{b,s}$ is either ϑ_s^b after standardization or $p_{is}(\Theta_b)$.

We implement the constrained k-means clustering algorithm by:

Algorithm 1. *Starting with a random set of initial centroids* $C_K = \{c_1, \ldots, c_K\}$*, e.g., by randomly choosing* K *vectors, iterate the following steps until a fixed point is reached:*

Step 1: *Assign each vector of component specific estimates* $x_{b,s}$ *to the cluster of the closest centroid:*

$$c(x_{b,s}) := \arg\min_{c \in C_K} d(x_{b,s}, c)$$

where $d(x_{b,s}, c)$ *denotes the Euclidean distance between observation* $x_{b,s}$

and centroid c.

Step 2: *If the constraint is violated for the estimates of one draw, i.e.*

$$\tilde{B} := \big\{b \in \{1, \ldots, B\} | \exists s, t \in \{1, \ldots, S\} : s \neq t \wedge c(x_{b,s}) = c(x_{b,t})\big\} \neq \emptyset,$$

then find the best assignment to the clusters under the constraint. This can be done by solving a linear sum assignment problem (LSAP) which consists of finding a minimum cost assignment of S objects to K persons given a cost matrix of dimension S times K under the constraint that not more than one object is assigned to each person. This problem can be solved using a primal-dual algorithm such as the so-called Hungarian method as outlined in Papadimitriou and Steiglitz (1982) which finds the optimum in time $\mathcal{O}(K^3)$. In this application the LSAP is solved $\forall b \in \tilde{B}$ with the cost matrix given by the distances between the $x_{b,s}$ $s = 1, \ldots, S$ and the current centroids C_K.

Step 3: *Update the set of centroids by averaging over the points which were assigned to each cluster:*

$$c_k := |A_k|^{-1} \sum_{x_{b,s} \in A_k} x_{b,s}$$

where A_k is the set of points in cluster k, i.e. $A_k := \{x_{b,s} \in X_{B,S} | c(x_{b,s}) = c_k\}$.

Relabelling algorithms

Stephens (2000b) proposed to take a decision theoretic approach by selecting an action which corresponds to a mixture distribution where the number of components is equal to the number of modes M. The prior class probabilities of this mixture are given by ξ^m and the mode specific actions by a^m. The loss between the action $(\boldsymbol{\xi}, \boldsymbol{a})$ and the true parameter vector Θ is determined by minimizing over the different modes:

$$\mathcal{L}\big((\boldsymbol{\xi}, \boldsymbol{a}); \Theta\big) = \min_m \{-\log \xi^m + \mathcal{L}_0(a^m; \Theta)\}$$

where $\boldsymbol{a} = (a^m)_{m=1,\ldots,M}$ and $\boldsymbol{\xi} = (\xi^m)_{m=1,\ldots,M}$ with $\xi^m > 0 \; \forall m$ and $\sum_{m=1}^M \xi^m = 1$.

The loss function for each mode can be taken as the Kullback-Leibler divergence ('KLdiv') between the a-posteriori probabilities $p_{is}(\Theta)$ of the observations and the action a^m where we drop the label-invariant term. This is given by:

$$\mathcal{L}_0(Q^m; \Theta) = \min_\nu \left\{ -\sum_{i=1}^n \sum_{s=1}^S p_{is}(\nu(\Theta)) \log(q_{is}^m) \right\}$$

The action a^m is given by $Q^m = (q_{is}^m)_{i,s}$, where q_{is}^m represents the probability that observation i is assigned to group s for mode m.

The algorithm is given by:

Algorithm 2. *Starting with some initial values for $\nu_{b,m}$ (setting them all to the identity permutation for example) and m_b, $b = 1, \ldots, B$ (using a random partition of the draws for example), iterate the following steps until a fixed point is reached holding all other parameters fixed in each step:*

Step 1: *Choose $\forall m = 1, \ldots, M$ the $\hat{Q}^m = (\hat{q}_{is}^m)_{i,s}$ that minimizes*

$$-\sum_{b=1}^B \sum_{i=1}^n \sum_{s=1}^S \mathbb{I}_{\{m_b=m\}} p_{is}(\nu_{b,m}(\Theta_b)) \log(\hat{q}_{is}^m)$$

where $\mathbb{I}$ is the indicator function.

Step 2: *For $b = 1, \ldots, B$ and $m = 1, \ldots, M$ choose $\nu_{b,m}$ to minimize*

$$-\sum_{i=1}^n \sum_{s=1}^S p_{is}(\nu_{b,m}(\Theta_b)) \log(\hat{q}_{is}^m)$$

Step 3: *For $b = 1, \dots, B$ choose m_b to minimize*

$$-\log(\hat{\xi}^m) - \sum_{i=1}^{n} \sum_{s=1}^{S} p_{is}(\nu_{b,m}(\Theta_b)) \log(\hat{q}_{is}^m)$$

Step 4: *Determine $\hat{\boldsymbol{\xi}}$ by*

$$\hat{\xi}^m = \frac{1}{B} \sum_{b=1}^{B} \mathbb{I}_{\{m_b = m\}}$$

The proposed loss function is especially appropriate for clustering inference. The advantage of this loss function is that it can be used for different component distribution functions and without modifications also in the regression case. Other loss functions have been proposed in the case of unimodality and can certainly also be applied in this context as the mode specific loss function. If the components follow a Gaussian distribution Stephens (1997) suggested to use a Kullback-Leibler divergence measure between the density functions. This loss function is given by

$$\begin{aligned}
&\mathcal{L}_0(\boldsymbol{\pi}^m, \boldsymbol{\mu}^m, \boldsymbol{\Sigma}^m; \Theta) = \\
&\quad \min_{\nu} \Bigg\{ -\sum_{s=1}^{S} \pi_s(\nu(\Theta)) \log(\pi_s^m) + (1 - \pi_s(\nu(\Theta))) \log(1 - \pi_s^m) + \\
&\qquad \pi_s(\nu(\Theta)) \int \phi(x | \boldsymbol{\mu}_s(\nu(\Theta)), \Sigma_s(\nu(\Theta))) \log(\phi(x | \boldsymbol{\mu}_s^m, \Sigma_s^m)) \mathrm{d}x \Bigg\}
\end{aligned}$$

where $\phi(\cdot|\cdot)$ is the Gaussian density function, $\boldsymbol{\pi}^m = (\pi_s^m)_s$, $\boldsymbol{\mu}^m = (\boldsymbol{\mu}_s^m)_s$ and $\boldsymbol{\Sigma}^m = (\Sigma_s^m)_s$. The action a^m is given by $(\boldsymbol{\pi}^m, \boldsymbol{\mu}^m, \boldsymbol{\Sigma}^m)$. In the following we refer to this loss function as ‘Densdiv’.

Another possibility mentioned in Stephens (2000b) similar to Celeux (1998) is given by

$$\mathcal{L}_0(\boldsymbol{\mu}^m, \boldsymbol{\Sigma}^m; \Theta) = \min_{\nu} \left\{ -\sum_{s=1}^{S} \log \phi(\mu_s(\nu(\Theta)); \boldsymbol{\mu}_s^m, \Sigma_s^m) \right\}$$

where the action a^m is given by $(\boldsymbol{\mu}^m, \boldsymbol{\Sigma}^m)$. This loss function will be referred to as ‘Celeuxdiv’.

In addition to random initialization the clustering results of constrained k-means can be used as heuristic to provide good initial values. For the Kullback-Leibler divergence it makes sense to use the a-posteriori probabilities $p_{is}(\Theta)$ as input data, while for the other two loss functions the component specific parameter values should be clustered after appropriate variable selection and standardization.

4.2.4 Illustration using an artificial example

In this section we apply the methods proposed in the previous sections on an artificial example in order to check for multimodality, make an exploratory cluster analysis and to relabel the fitted bootstrap models given a specified number of modes. We use a finite mixture of Gaussian regression models which is not identifiable due to intra-component label switching. In this case the number of global modes and the true parameters of the different modes are known and we can therefore evaluate the performance of the methods. The mixture regression of the example is given by

$$H(y|\mathbf{x}, \Theta) \;=\; \sum_{s=1}^{3} \frac{1}{3} N(\mu_s(\boldsymbol{x}), 0.01)$$

where $\mu_s(\mathbf{x}) = \mathbf{x}'\boldsymbol{\beta}_s$ and $N(\mu, \sigma^2)$ is the Gaussian distribution.

We assume that the regressors consist of an intercept, a continuous variable $x_1 \in [0, 1]$ and an interaction term between a binary variable x_2 and x_1. For simplicity of presentation we assume that there is no main effect of the binary variable x_2, i.e. the coefficient is equal to 0 for all components. The interaction term between x_1 and x_2 is in the following denoted by x_1:x_2.

As Gaussian mixture distributions are generically identifiable the means, variances and component weights are uniquely determined in each covariate point. Due to the specific structure of the covariate matrix, only the following three covariate points are necessary to uniquely determine the marginal distributions in each possible covariate point. Let the component specific means in these covariate points be given by $\boldsymbol{\mu}(x_1 = 0, x_2 = 0) = (4, 4, 2)$,

$\boldsymbol{\mu}(x_1 = 1, x_2 = 0) = (4, 2, 2)$ and $\boldsymbol{\mu}(x_1 = 1, x_2 = 1) = (2, 0, 2)$.

As the ordering of the components in each point is not unique due to the violation of the coverage condition (Hennig 2000), the two possible solutions for $\boldsymbol{\beta} := (\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \boldsymbol{\beta}_3)$ are:

Solution 1: $\boldsymbol{\beta}_1^{(1)} = (4, -2, \quad 0)'$, $\boldsymbol{\beta}_2^{(1)} = (4, 0, -4)'$, $\boldsymbol{\beta}_3^{(1)} = (2, 0, 0)'$

Solution 2: $\boldsymbol{\beta}_1^{(2)} = (4, -2, -2)'$, $\boldsymbol{\beta}_2^{(2)} = (4, 0, -2)'$, $\boldsymbol{\beta}_3^{(2)} = (2, 0, 0)'$

The omission of x_2 in the regression clearly simplifies the example, because the mixture with the same marginal distributions where the binary variable x_2 is also included in the regression and allowed to vary between the components, has 6 different parameterizations.

In the following we use a sample with 100 observations from this mixture distribution, where the x_1 values are equidistantly given in the interval $[0, 1]$ and we observe both x_2 values for each x_1 value. We fit a finite mixture model with 3 components to the sample using the EM algorithm. As the EM algorithm may converge to a local maximum only, we report the best solution with respect to the likelihood from 10 random initializations. Random initialization means that an a-posteriori probability of 0.9 is assigned to one of the three components and an a-posteriori probability of 0.05 to the other two components for each observation where the component with a-posteriori probability 0.9 is determined randomly with equal probability. Further implementational details are given in Leisch (2004).

The sample together with the fitted values for all three components is given in Figure 4.12. The estimated parameters with the standard deviations derived using standard asymptotic theory as in Turner (2000) are given in Table 4.5.

Checking for genuine multimodality

In order to check for genuine multimodality we proceed as outlined in Section 4.2.2. We sample 200 parametric bootstrap samples from the fitted model and fit a mixture model with three components using the EM algorithm with 10 random initializations to each of them. In Figure 4.13 the

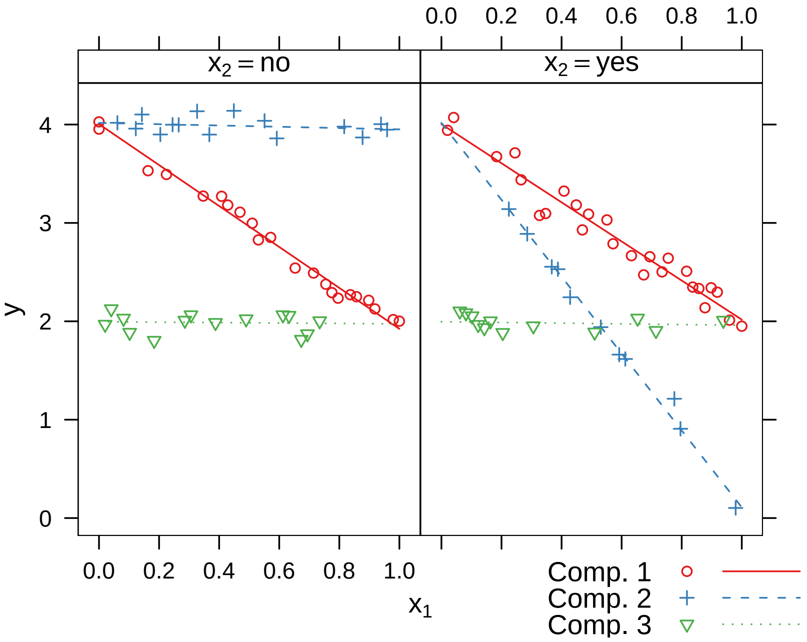


Figure 4.12: Sample with 100 observations from the artificial example together with the fitted values for each component. The symbols of the data points are correspond to the component with the maximum a-posteriori probability.

	(Intercept)	x_1	x_1:x_2	σ	π
Comp. 1	4.00	-2.08	0.09	0.11	0.45
	(0.04)	(0.07)	(0.05)	(0.05)	(0.05)
Comp. 2	4.02	-0.07	-3.84	0.08	0.27
	(0.03)	(0.06)	(0.06)	(0.05)	(0.05)
Comp. 3	2.00	-0.02	-0.01	0.08	0.28
	(0.03)	(0.07)	(0.07)	(0.05)	(0.05)

Table 4.5: Estimated parameters (standard deviations) of the finite mixture fitted to a sample with 100 observations.

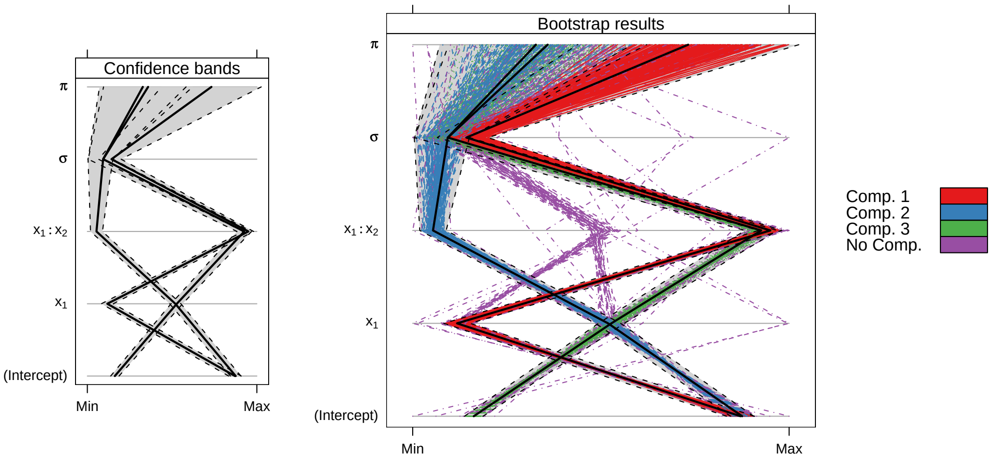


Figure 4.13: Parallel coordinate plot of the parameter estimates for 200 bootstrap samples. The black lines give the fitted values and the dashed black lines the 95% confidence bands. The coloring indicates the assigned membership to one component or to no component.

results are visualised using parallel coordinate plots for the component specific parameter estimates of the bootstrap samples together with the 95% confidence bands for the parameter estimates for each component.

We check if the confidence bands and the bootstrap samples correspond. There are 83% of the component specific bootstrap parameters completely within one of the bands while we would expect 95%. The percentage within one band given that 200 should belong to each component is for the first component 81.5%, for the second 79.5% and for the third 87%. While there are nearly as many bootstrap samples as expected within the band of component 3, there are a lot less than expected in the other two components indicating that there is a problem for these two components. The difference between the observed and expected values is mainly caused by intra-component label switching, as two bundles corresponding to the second mode for these two components can be clearly distinguished in Figure 4.13, due to the difference of the coefficients for the interaction x1:x2. In addition, there can be also a few spurious results seen which are due to convergence problems of the EM algorithm.

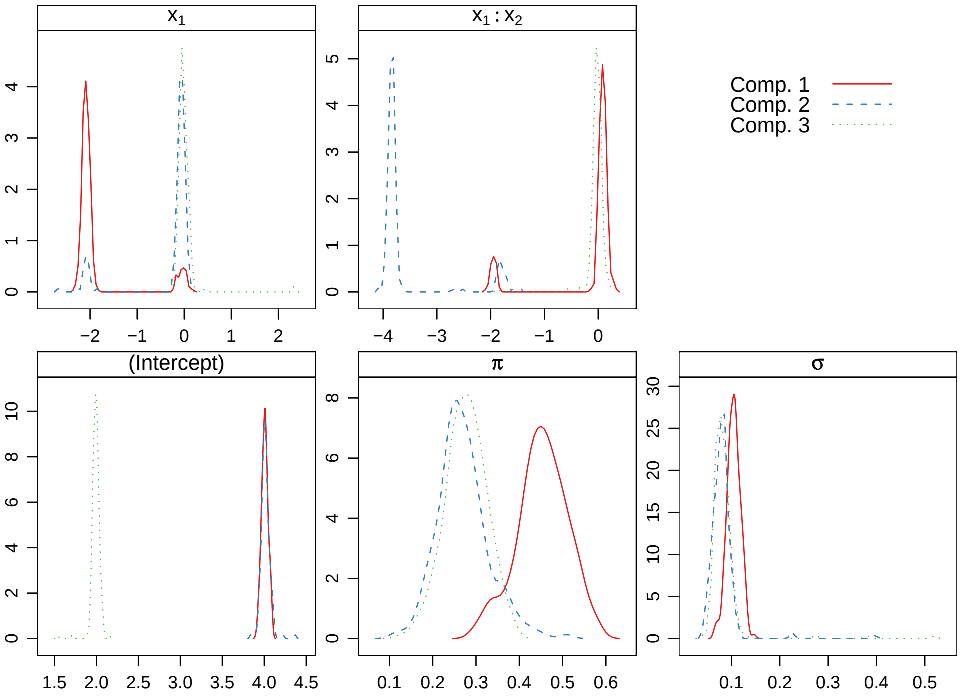


Figure 4.14: Kernel density plot of the bootstrap parameter estimates after relabelling with $M = 1$ and loss 'KLdiv'.

For the second approach we use the relabelling algorithm with loss 'KLdiv' to determine a unique labelling of the bootstrap samples. We check if the relabelling algorithm returned a unique labelling by using density plots and by testing for unimodality of the component specific parameters. In Figure 4.14 we see that the component specific estimates are unimodal for the intercept, the component weights and σ. Furthermore, they are also unimodal for the other parameters for component 3. For components 1 and 2 there can be two different, widely separated modes for the coefficients of x_1 and $x_1 : x_2$ distinguished. This indicates that the relabelling algorithm did not succeed in determining a unique labelling. As one reason for this is multimodality of the likelihood, a further analysis is necessary.

The p-values of the dip test for the parameters aggregated over all components and separately for each component are given in Table 4.6. The p-values

	(Intercept)	x_1	x_1:x_2	σ	π
Aggregated	0.06	0.06	0.06	0.97	0.57
Comp. 1	1.00	0.00	0.00	1.00	1.00
Comp. 2	1.00	0.00	0.00	1.00	1.00
Comp. 3	1.00	1.00	1.00	1.00	1.00

Table 4.6: P-values of the dip test for the bootstrap parameter estimates adjusted for multiple testing using the method given in Holm (1979).

are determined under the null hypothesis of a uniform distribution for the aggregated values and of a Gaussian distribution for the separate components and are adjusted for multiple testing using the method given in Holm (1979). The dip test indicates that the components clearly do not differ for σ and the component weights for the aggregated parameter estimates. This signifies that these variables are not suitable for imposing an ordering constraint. Furthermore, it might be possible to estimate a constrained mixture model with the variance fixed over all components. Similar to the density plot unimodality is rejected for the component specific estimates of x_1 and $x_1 : x_2$ for components 1 and 2.

In order to investigate the size and power of the proposed testing procedure 100 samples are drawn from an identifiable and an unidentifiable mixture and the procedure is applied to each of them. For the power analysis the samples are drawn from the given example, whereas for the size analysis the example is modified such that there are 25 observations where the component membership is fixed for both values of x_2 in order to eliminate intra-component label switching. In Table 4.7 the relative numbers of samples where the null hypothesis is rejected for a given nominal significance level are given. The p-values for the dip test are determined for each parameter and component under a uniform and a Gaussian null distribution and are then adjusted for multiple testing using the method given in Holm (1979).

Even if the number of replications is rather small due to the computational complexity involved, it can clearly be seen that the dip test under the uniform null hypothesis is conservative while the Gaussian distribution gives

	Size		Power	
Nominal level	Uniform	Gaussian	Uniform	Gaussian
0.01	0.00	0.01	0.93	0.98
0.02	0.00	0.01	0.93	0.98
0.04	0.00	0.05	0.93	0.99
0.06	0.00	0.06	0.94	0.99
0.08	0.00	0.08	0.96	0.99
0.10	0.00	0.09	0.97	0.99
0.20	0.00	0.17	0.97	0.99

Table 4.7: Estimated true levels and power for given nominal levels using 100 random samples from an identifiable and an unidentifiable mixture.

a good level accuracy. The power performance is very good for both null hypothesis distributions for the given unidentifiable mixture distribution. The null hypothesis is rejected for nearly all samples for any nominal significance level even though the performance under the Gaussian null distribution is slightly better.

This small simulation study justifies the choice of the Gaussian distribution as null distribution and validates the testing procedure as the nominal significance levels under the null distribution are met and the null hypothesis is rejected for 99% of the samples for a significance level of 0.05 if the mixture is not identifiable.

Imposing an ordering constraint

As using an ordering constraint on a single variable to determine a unique labelling is one of the most popular approaches to solve the label switching problem we also apply this method. However, this is not a suitable way to solve the label switching problem in this example, as at least two components overlap for each of the variables. The dip test results in Table 4.6 clearly indicate that σ and the component weights are no suitable variables for imposing an ordering constraint. The results for the other three variables are given in Figure 4.15 and it can be seen that the label switching problem has not been solved by any of them.

If the ordering constraint fails to determine a unique labelling for the

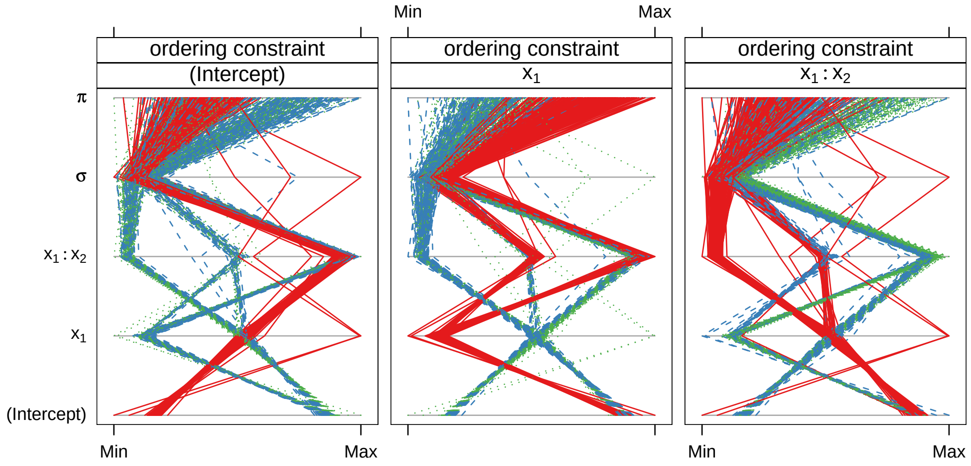


Figure 4.15: Component assignment of the bootstrap results with an ordering constraint for each of the variables where overall unimodality was rejected.

components for each of the modes, it is also not easily possible to separate the two modes by clustering the relabelled complete parameter vectors. This is shown using k-means as cluster algorithm and the complete bootstrap parameter vectors after relabelling with respect to the intercept as input data. The results suggest in fact the presence of at least 4 modes. The 4-cluster solution together with the sum of within cluster distances for the different number of clusters (modes) is given in Figure 4.16. Even if the labelling is unique for the components and there are also only samples which correspond to the same mode for each cluster in the 4-cluster solution, it can be noticed that cluster 1 and 3 and cluster 2 and 4 correspond to the same modes after suitable relabelling.

Exploratory analysis of the modes

k-means In the following we use only the component specific parameter values to show the application of clustering methods using k-means and constrained k-means. The results for clustering of the component specific a-posteriori probabilities are omitted as they gave similar results. For the constrained k-means algorithm at least 3 clusters have to be specified be-

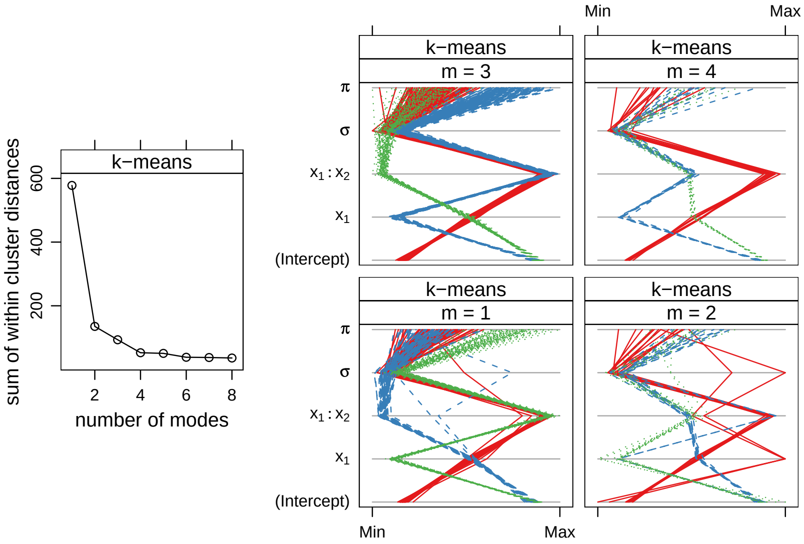


Figure 4.16: Number of modes suggested by k-means and the solution for each mode using the parameter vectors after imposing an ordering constraint on the intercept.

cause there is no feasible solution otherwise. The maximum number of clusters estimated is 12 which corresponds to at least four modes. As input we use only the parameters where the dip test rejected the null hypothesis of unimodality for the aggregated values, i.e. we exclude σ and the component weights. Furthermore, we standardise the input variables.

All unconstrained k-means solutions for $K = 4, \ldots, 12$ meet the constraints and only two bootstrap samples violate the constraints for $K = 3$. This signifies that the constraints are in fact not necessary, because the components are sufficiently different from each other. This result can be seen as an indicator that the choice of number of components is adequate. The sum of within cluster distances indicate that there are 5 clusters present in the data.

In Figure 4.17 the suggested relabelling for each of the modes implied by the constrained k-means solution with 5 clusters is given. 4 modes are induced by the 5 clusters and the information which parameters belong to the same bootstrap sample. The size of the modes are 24, 2, 2 and 172.

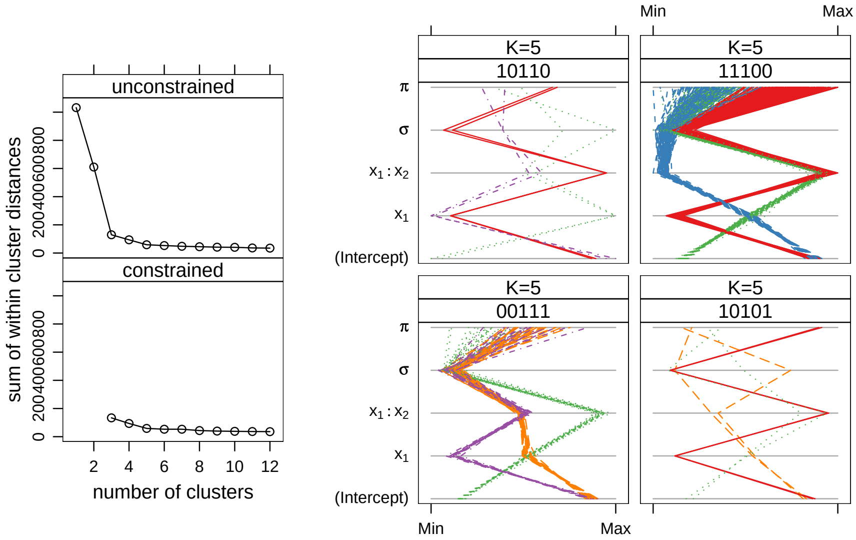


Figure 4.17: Number of clusters suggested by unconstrained and constrained k-means clustering and the parallel coordinate plot of the 200 bootstrap samples where the coloring indicates the cluster assignments for 5 clusters.

The two small modes occur because the EM algorithm failed to detect the global optimum and converged to a local optimum in these cases. The binary numbers in the lower title strip of each panel indicate to which of the 5 clusters the components of the bootstrap samples have been assigned. This shows that the third cluster is present in each mode.

Hierarchical clustering Similar to unconstrained k-means clustering hierarchical clustering methods can also be used for an exploratory analysis in order to determine the number of component specific estimates and modes. In the following we use an agglomerative hierarchical clustering algorithm with Euclidean distance measure after parameter specific standardisation and complete linkage. We use again only the parameters where the dip test rejected the null hypothesis of unimodality for the aggregated values.

The resulting dendrogram is given in Figure 4.18 (left). We choose a 7-cluster solution and cut the dendrogram in the respective height. The parallel coordinate plot in Figure 4.18 (right) gives the estimates separately

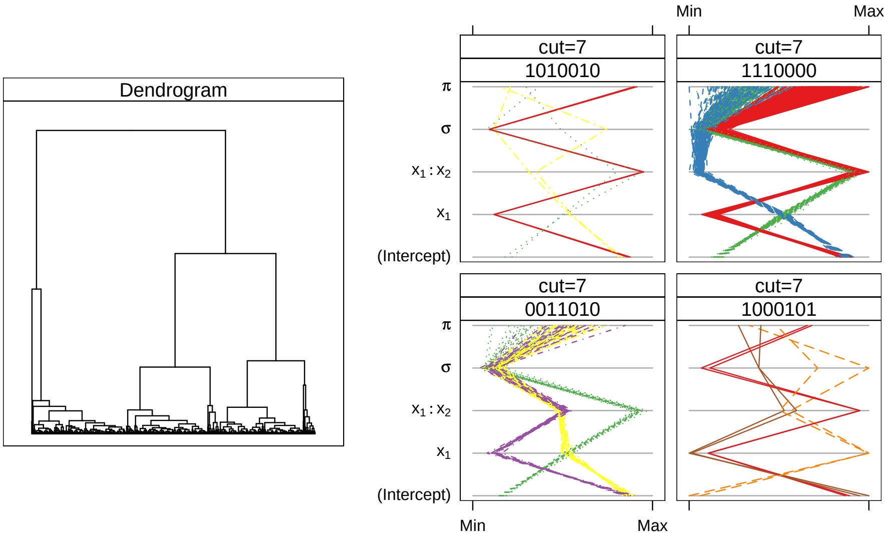


Figure 4.18: Dendrogram of the hierarchical clustering results and the parallel coordinate plot of the 200 bootstrap samples where the coloring indicates the cluster assignments for 7 clusters.

for each mode induced by the cluster assignments and the information which estimates belong to the same bootstrap sample. The coloring is according to the 7-cluster solution. The number of bootstrap samples in each mode are again 24, 2, 2 and 172. The two small components contain the bootstrap samples where the EM algorithm converged to a local maximum, while the other two components correspond to one of the global modes.

Relabelling under genuine multimodality

Random initialization The relabelling algorithm given in Section 4.2.3 is applied with $M = 1, \ldots, 4$. As the algorithm might only converge to a local minimum, the best solution of 10 runs with random initialization is reported. This is in fact still not sufficient to determine the global maximum, as a local maximum with less number of modes as specified is found in certain cases. The number of modes returned together with the corresponding objective values are given in Figure 4.19 (left). On the right the solution for 'KLdiv' with 3 modes is given.

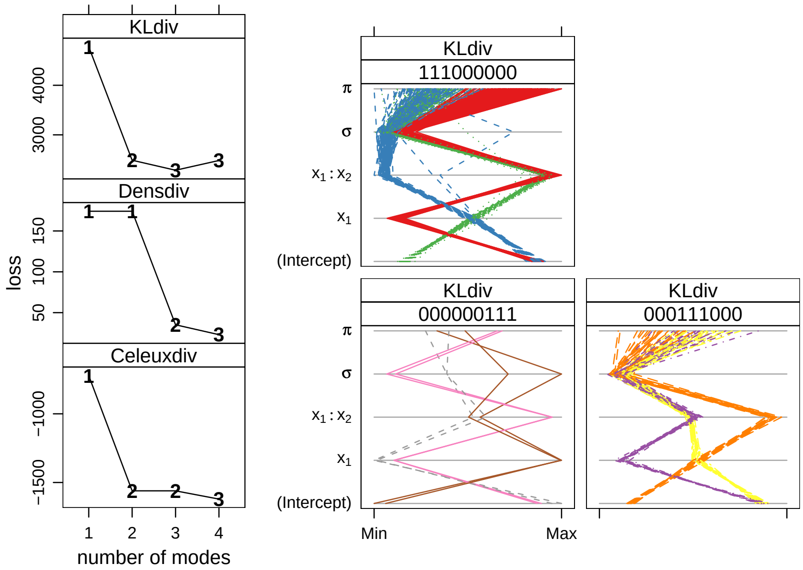


Figure 4.19: Values of the objective function together with the number of modes found and the parallel coordinate plot of the parameter estimates for the 200 bootstrap samples. The coloring indicates the labels of each component and mode using loss function 'KLdiv'.

In Table 4.8 the correspondence between the cluster solutions using the different loss functions is given using the Rand index corrected for agreement by chance (Hubert and Arabie 1985). Most of the Rand indices are close to one if the number of modes is larger or equal to two, as in these cases the two big modes can be separated while samples of the two small modes containing the bootstrap samples where the EM algorithm did not converge to a suitable root are differently assigned.

It can be therefore seen that the results of the relabelling algorithms are

	Number of Modes			
	1	2	3	4
KLdiv vs. Densdiv	0.79	0.83	1.00	0.98
KLdiv vs. Celeuxdiv	0.79	0.97	0.98	0.96
Densdiv vs. Celeuxdiv	1.00	0.80	0.98	0.98

Table 4.8: Class agreement between the cluster solutions of the different loss functions using the Rand index corrected for agreement by chance.

similar for the different loss functions even in the case of multiple modes. A similar conclusion was already drawn in the unimodal case (cf. Stephens 2000b).

Initialization in constrained k-means solution As 10 repetitions with random initialization were not sufficient to detect the global optimum, we try to initialize in the constrained k-means solution where the mode assignment is determined using the cluster assignment together with the information which components belong to the same sample.

The relabelled results for loss 'KLdiv' and initialization with 5 clusters are given in Table 4.9 together with the results for the relabelling with only one mode specified. The initialization in the k-means solution led to 4 modes, which we were not able to detect with random initialization. There are two small modes which contain the bootstrap results where the EM algorithm converged to a local maximum and two larger modes, which correspond to the two global modes. This method was not only able to separate the global modes, but also the spurious results and it can be seen that the estimates of the standard deviations for the largest mode correspond to those derived with standard asymptotic theory (cp. Table 4.5).

4.3 Summary

In this chapter it has been shown that resampling methods are a useful tool for finite mixture model diagnostics. Given the increase in computing power they are not prohibitive computational demanding any more and could be applied by default in a standard application. The purpose of the application might be to detect genuine multimodality as different important modes are suspected to be present for the fitted model given the available data. In addition cluster stability can be assessed or the standard errors estimated using standard asymptotic theory can be validated.

The label switching problem which occurs for random initialization of the EM algorithm was addressed. The suggestion for using a relabelling algorithm under the assumption of multimodality given in Stephens (2000b)

Modes	ξ	Comp.	(Intercept)	x_1	x_1:x_2	σ	π
1	1.00	1	4.01	-1.84	-0.16	0.10	0.45
			(0.04)	(0.67)	(0.67)	(0.01)	(0.06)
		2	4.01	-0.33	-3.56	0.08	0.27
			(0.05)	(0.70)	(0.70)	(0.03)	(0.06)
		3	1.99	0.01	-0.03	0.09	0.28
			(0.05)	(0.25)	(0.19)	(0.04)	(0.05)
4	0.86	1	4.01	-2.08	0.09	0.11	0.46
			(0.04)	(0.08)	(0.07)	(0.01)	(0.05)
		2	4.01	-0.06	-3.83	0.08	0.26
			(0.05)	(0.07)	(0.12)	(0.03)	(0.05)
		3	1.99	-0.02	-0.01	0.08	0.28
			(0.03)	(0.08)	(0.07)	(0.01)	(0.05)
	0.12	1	4.00	-2.08	-1.80	0.09	0.37
			(0.04)	(0.08)	(0.07)	(0.01)	(0.06)
		2	2.00	-0.02	-0.01	0.08	0.27
			(0.03)	(0.09)	(0.08)	(0.02)	(0.05)
		3	4.01	-0.04	-1.95	0.10	0.37
			(0.06)	(0.09)	(0.05)	(0.02)	(0.05)
	0.01	1	4.06	-2.20	0.13	0.11	0.56
			(0.05)	(0.18)	(0.08)	(0.03)	(0.03)
		2	2.06	0.12	-0.46	0.18	0.22
			(0.14)	(0.39)	(0.15)	(0.14)	(0.05)
		3	3.93	0.02	-3.31	0.07	0.22
			(0.06)	(0.02)	(0.86)	(0.04)	(0.02)
	0.01	1	1.62	2.34	-1.73	0.45	0.30
			(0.12)	(0.01)	(0.19)	(0.10)	(0.05)
		2	4.03	-2.14	0.09	0.08	0.43
			(0.04)	(0.00)	(0.00)	(0.02)	(0.01)
		3	4.30	-2.64	-1.60	0.23	0.27
			(0.10)	(0.05)	(0.21)	(0.00)	(0.04)

Table 4.9: Mean component specific parameter estimates (standard deviations) using 200 bootstrap samples and the relabelling algorithm with loss 'KLdiv' with initialization suggested by constrained k-means.

was explored. In addition constrained clustering (Leisch and Grün 2006) was considered for reordering the components under the assumption of uni- or multimodality. This method can be applied even if there exists no ordering constraint on a single variable which induces a unique labelling. It can also be used for mixtures with a different number of components. If there are components omitted during the estimation of the models to the bootstrap samples the remaining components can nevertheless be assigned to different clusters. If the presence of different modes is suspected the number of clusters can be increased and it is possible to simultaneously assign the estimated models to different modes and induce a suitable ordering of the components.

Chapter 5

Implementation in R

In this chapter the design principles of the R package **flexmix** are discussed and the main functions presented. The implementational details which are necessary for being able to extend the package and write new model drivers are explained. The application of the package is demonstrated on two data sets and examples for extending the package by writing a new concomitant variable model and a new component specific model are given. The extension of the package to allow for concomitant variable models and for varying and fixed effects in the component specific models is also described in Grün and Leisch (2006a,b).

5.1 Design principles

flexmix implements flexible finite mixture modelling. It provides maximium likelihood estimation with the EM algorithm and some of its variants. An overview on the package is given in Leisch (2004). The main design principles are easy extensibility and fast prototyping for new types of mixture models. It uses S4 classes and methods (Chambers 1998) as implemented in the R package **methods** and exploits certain features of R such as lexical scoping (Gentleman and Ihaka 2000). It provides the E-step and all data handling, while the M-step can be supplied by the user to easily define new models. The main focus is on finite mixtures of regression models and it allows for multiple independent responses, repeated measurements, to specify some control arguments for the EM algorithm and provides tools for automated model search.

Functions and model formulae are first class objects in the S language, which allows in combination with the lexical scoping rules of R for very modular software design. Rather than using text mode arguments used as switches within function bodies, **flexmix** uses driver functions to specify all aspects of the mixture model. Users can either use the growing collection of drivers distributed as part of **flexmix**, or write and use their own drivers.

In a first step the (unfitted) component specific model $F(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{\vartheta}_s)$ and the concomitant variable model $\pi(\boldsymbol{w}, \boldsymbol{\alpha})$ have to be specified. For this no data are needed, only the names of the independent and dependent variables and their respective interaction structure are defined. The component specific model is specified with `FLXglm()` or `FLXglmFix()` and the multinomial logit concomitant variable model with `FLXmultinom()`. If no concomitant variable model is fitted the function `FLXconstant()` is used by default.

`FLXglm()` only allows varying effects for the coefficients and the dispersion parameters. In this case the likelihood can be maximized separately for each component in the M-step of the EM algorithm. If there are also fixed and nested varying effects for the regression coefficients and dispersion parameters, `FLXglmFix()` has to be used and the likelihood is maximized simultaneously for all components. The design matrix is constructed by replicating the observations K times with suitable columns of zeros added. Model formulae for the varying, nested varying and fixed effects have to be provided. These are evaluated by successively updating the formula of the random effects with the formula for the fixed and then the nested varying effects.

The concomitant variable model is specified in a similar fashion. The default dummy driver `FLXconstant()` uses no concomitant variables and acts only as a placeholder. For multinomial logistic regression `FLXmultinom()` can be used.

By default, EM is initialized using random assignment of observations to mixture components, function `stepFlexmix()` can be used to automatically determine the best solution out of several random initializations. The possibility to start EM with user-specified posteriors or (more common) posteriors from a previous run is also provided. To select a model with a suitable number of components information criteria such as the Akaike information criterion (AIC), the Bayesian information criterion (BIC) and the integrated completed likelihood information criterion (ICL; Biernacki et al. 2000) can be used.

The EM algorithm is controlled by the `control` argument of `flemix()`, where the maximum number of iterations and the tolerance of (relative) change of log-likelihood for stopping can be given. In addition it can be specified if the a-posteriori probabilities ("weighted"), the assignment to the maximum a-posteriori probability ("hard") or a random assignment to one component by sampling from a multinomial distribution with probabilities equal to the a-posteriori probabilities ("random") is used in the M-step. The variant with hard assignment is also referred to as Classification EM (CEM; Celeux and Govaert 1992) and with random assignment as Stochastic EM (SEM; Diebolt and Ip 1996).

A minimum component weight or prior probability of the components can be required such that components where the prior probability drops under a certain threshold are omitted during the EM algorithm. If concomitant variable models are fitted the average prior probabilities over the given data points are used. This strategy can be either used for model selection if the model builder is only interested in mixtures where the components are of a given minimum size or to prevent convergence of the EM algorithm to a solution with an unbounded likelihood, as e.g. Gaussian mixtures have an unbounded likelihood if components with zero variance are present.

`flexmix()` returns an object of class `flexmix` and methods defined for this class include `show()`, `summary()` and `plot()`. `show()` gives the call, the table of cluster assignments and the number of iterations until convergence. Further details are given by `summary()` which provides the prior probabilities together with the table of cluster assignments, the number of observations with a-posteriori probability larger than `eps` and the ratio of these numbers, which indicates how well separated the components are. In addition the likelihood (with degrees of freedom used), the AIC and the BIC are printed. The default plot is a rootogram of the a-posteriori probabilities for each component. In addition there are accessor functions for the component specific parameters (`parameters()`), for the a-posteriori probabilities (`posterior()`), the maximum a-posteriori class assignments (`cluster()`) and the fitted values for each component (`fitted()`). More information on the estimated paramters of the component specific and concomitant variable models can be obtained using `refit()` (see Section 5.3.1).

5.2 Implementational details

With respect to **flexmix** version 1.0-0 described in Leisch (2004) the implementation had to be modified to include concomitant variable models and mixtures of regression with varying and fixed effects. This made the definition of a better class structure for the component specific models and the modification of the fit functions `flexmix()` and `FLXfit()` necessary.

For the component specific model we now have a virtual class `"FLXM"` which has (currently) two subclasses: `"FLXMC"` for model-based clustering and `"FLXMR"` for clusterwise regression, i.e. there are independent variables given. Additional slots have been introduced to allow data preprocessing and the construction of the components was separated from the fit and is now captured as an expression (to allow for lexical scoping) in the slot `defineComponent`. `"FLXMC"` has an additional slot `dist` to specify the name of the distribution of the variable.

For `flexmix()` and `FLXfit()` code blocks which are model dependent have been identified in these functions and different methods implemented. Finite mixtures of regression with varying, nested and fixed effects were a suitable model class for this identification task as they are different from models previously implemented. The main differences are:

- The number of components is related to the component specific model and the omission of small components during the EM algorithm impacts on the model.

- The parameters of the component specific models can not be determined separately in the M-step and a joint model matrix is needed.

This makes it also necessary to have different model dependent methods for `fitted()` and `refit()`.

The default plot methods now use lattice graphics. Users familiar with the syntax of lattice graphics and with the plotting and printing arguments will find the application more intuitive as a lot of plotting arguments are passed to lattice functions. In fact only new panel, pre-panel and group-panel functions were implemented. The returned object is of class `"trellis"` and the print method can be used for plotting.

5.2.1 Component models with varying and fixed effects

A new M-step driver is provided which fits finite mixtures of GLMs with fixed and nested varying effects for the coefficients and the dispersion parameters. The class `"FLXMRglmfix"` returned by the driver `FLXglmFix` has the additional slots with respect to `"FLXMRglm"`:

design: An incidence matrix indicating which columns of the model matrix are used for which component.

nestedformula: An object of class `"FLXnested"` containing the formula for the nested effects of the regression coefficients and the number of components in each K_c, $c \in C$.

fixed: The formula for the fixed effects of the regression coefficients.

variance: A logical indicating if varying effects should be estimated or a vector specifying the grouping of the nested effects for the variance of the Gaussian distribution.

The difference between estimating finite mixtures including only varying effects using models specified with `FLXglm()` and those with varying and fixed effects using function `FLXglmFix()` is hidden from the user, as the user interface for function `flexmix()` is the same. The fitted model is of class `"flexmix"` and can be analyzed using the same functions as for any model fitted using package **flexmix**. The methods used are the same except if the slot containing the model is accessed and method dispatching is made via the model class. New methods are provided for models of class `"FLXMRglmfix"` for the following functions: `refit()`, `fitted()` and `predict()` which can be used for analyzing the fitted model.

The implementation allows repeated measurements by specifying a grouping variable in the formula argument of `flexmix()`. Furthermore, it has to be noticed that the formulas of the different effects are evaluated by updating the formula of the random effects successively with the formula of the fixed and then of the nested varying effects. This ensures that if a random effect is fitted to the intercept, the model matrix of a categorical variable includes only the remaining columns for the fixed effects to have full column rank. However, this updating scheme makes it impossible to estimate fixed effects for the intercept while fitting random effects to a categorical variable.

5.2.2 Concomitant variable models

For representing concomitant variable models the class `"FLXP"` is defined. It specifies how the concomitant model is fitted using the concomitant variable model matrix as independent variables and the current a-posteriori probability estimates as dependent variables. The object has the following slots:

fit: A function (x, y, ...) returning the fitted values for the component weights during the EM algorithm.

refit: A function (x, y, ...) used for refitting the model.

df: A function (x, k, ...) returning the degrees of freedom used for estimating the concomitant model given the model matrix `x` and the number of components `k`.

x: A matrix containing the model matrix of the concomitant variables.

formula: Formula for determining the model matrix `x`.

name: A character string describing the model, which is only used for print output.

Two constructor functions for concomitant variable models are provided. `FLXconstant()` is for constant component weights without concomitant variables and `FLXmultinom` for multinomial logit models. `FLXmultinom()` has its own class `"FLXPmultinom"` which extends `"FLXP"` and has additional slots for the fitted coefficients. The multinomial logit models in Equation (2.3) are fitted using package **nnet** (Venables and Ripley 2002).

5.3 Illustration

In this section the application of the package is demonstrated on two data sets as well as the way how it can be extended for a concomitant variable model with only categorical variables and to fit zero-inflated Poisson or binomial regression models.

5.3.1 Application

We now illustrate model fitting and model selection in R on simple artificial data from a mixture of binomial regression models and on the `patent` data set taken from Wang et al. (1998) for a mixture of Poisson regression models. More examples for members of the GLM family are provided as part of the software package through a collection of artificial and real world data sets, most of which have been previously used in the literature (see references in the online help pages). Each data set can be loaded to R with `data(name)` and the fitting of the proposed models can be replayed using `example(name)`. Further details on these examples are given in the Appendix B. The data sets are: `betablocker` and `Mehta` (Aitkin 1999b), `fabricfault` (Aitkin 1996), `salmonellaTA98` and `seizure` (Wang et al. 1996), `tribolium` (Wang and Puterman 1998) and `trypanosome` (Follmann and Lambert 1989).

Logistic Regression Example

The artificial data considered here are sampled from a mixture distribution with three components and with varying effects for the intercept and nested varying effects for covariate x. The mixture distribution is given by:

$$H(y|x, w, \boldsymbol{\Theta}) = \sum_{s=1}^{3} \pi_s(w, \boldsymbol{\alpha}) \mathrm{Bi}(y|N, \theta_s)$$

where $\mathrm{Bi}(\cdot|N, \theta)$ denotes the binomial distribution with success probability θ and number of repetitions N. The success probabilities are given by

$$\begin{aligned}
\mathrm{logit}(\theta_1) &= x\beta_{2,1} + \beta_{3,1} \\
\mathrm{logit}(\theta_2) &= x\beta_{2,1} + \beta_{3,2} \\
\mathrm{logit}(\theta_3) &= x\beta_{2,2} + \beta_{3,3}
\end{aligned}$$

where $\beta_{2,.} = (2, 0)$ and $\beta_{3,.} = (-4, 1, 3)$. The component weights depend on the variable w and are determined by

$$\begin{aligned}
\text{Class 2:}\, \mathrm{logit}[\pi_2(w, \boldsymbol{\alpha})] &= 1 - w \\
\text{Class 3:}\, \mathrm{logit}[\pi_3(w, \boldsymbol{\alpha})] &= w.
\end{aligned}$$

A random sample with 200 observations is drawn from this mixture distribution for $N = 20$, x standard Gaussian and w from the set $\{0, 1\}$ with equal probability (and independent of x). The observations are plotted separately for the two levels of w in Figure 5.1, the plotting symbol corresponds to the true component membership. It can be clearly seen that most observations are from Class 2 for $w = 0$ and from Class 3 for $w = 1$.

In practice the true structure of the data is unknown, so we start by fitting a full model with different parameters for each component, the corresponding R code is shown in Figure 5.2. After loading package and data and setting a random seed for repoducability of the results, we define the concomitant variable model and store it in object `Conc`. Then we define the full model using function `FLXglm()` and store it in `Model.1`, note that the actual data

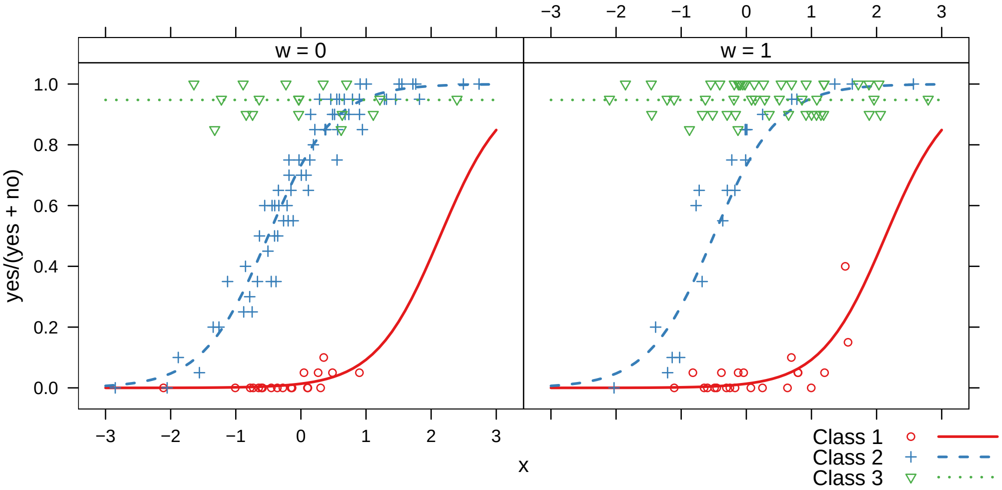


Figure 5.1: Sample with 200 observations from a mixture of binomial regression models. The plotting symbols correspond to the true component memberships and the lines are the fitted values.

have not been used so far. Finally, we fit a mixture model with 2 to 4 components using `nrep=5` replications of the EM algorithm for each model and store the best of each models in `Fitted.1`. The number of components can be selected using AIC or BIC by comparing the values of the information criteria for these models, e.g. with `sapply(Fitted.1, BIC)`. This suggests three components. In the next step, we determine the correct structure for the fixed and varying effects.

Figure 5.3 depicts the values of the intercept and coefficients for covariate x together with 95% confidence intervals. The intercepts in the three components are all different and the confidence intervals do not overlap. For x we get a completely different picture: The coefficient for Component 3 is almost zero (and hence greyed out), and the confidence intervals for the other two components overlap. Note that the confidence intervals are not taking into account that the components have been estimated simultaneously and are not independent, hence overlaps with zero or other components should only be interpreted as hints for model selection, not as formal significance tests.

We now use function `FLXglmFix()` to specify a more parsimonious model: We have a varying effect for the intercept (formula `~1`) and restrict the first

```
> library(flexmix)
> set.seed(8)
> data(BregFix)
> Conc <- FLXmultinom(~w)
> Model.1 <- FLXglm(~x, family = "binomial")
> Fitted.1 <- stepFlexmix(cbind(yes, no) ~ 1, data = BregFix,
+     model = Model.1, K = 2:4, concomitant = Conc,
+     nrep = 5)
> Model.2 <- FLXglmFix(~1, nested = list(formula = c(~x,
+     ~0), k = c(2, 1)), family = "binomial")
> Fitted.2 <- flexmix(cbind(yes, no) ~ 1, data = BregFix,
+     cluster = posterior(Fitted.1[["3"]]), model = Model.2,
+     concomitant = Conc)
```

Figure 5.2: Fitting mixtures of binomial regression models without constraints (`Model.1`) and with grouped varying effects (`Model.2`).

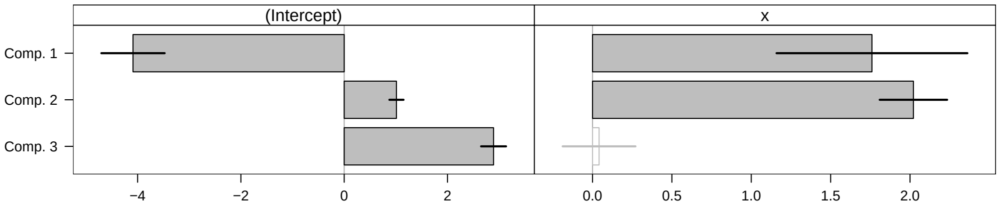


Figure 5.3: Coefficients of the larger `Model.1`.

two components to have the same coefficient for x (nested formula `~x`) and the third component to have only the intercept (nested formula `~0`). The concomitant variable model remains unchanged. To get the same ordering of the components and speed up computations we initialize EM with the posteriors of the first model. `Fitted.1[["3"]]` has a BIC of 903.57, while the BIC of `Fitted.2` is 893.58, so the smaller model is prefered (AIC and ICL lead to the same result). Details of the smaller model are shown in Figure 5.4, all coefficients differ from zero and between components. Thus the correct model would have been obtained even without knowledge of the true data generating process. Figure 5.1 shows the corresponding predicted values as lines.

Patent data: Poisson regression models

The patent data given in Wang et al. (1998) consist of 70 observations on patent applications, R&D spending and sales in millions of dollar from pharamaceutical and biomedical companies in 1976 taken from the National Bureau of Economic Research R&D Masterfile. The observations are displayed in Figure 5.6. The model which is chosen as the best in Wang et al. (1998) is given by:

$$H(\text{Patents}|\text{lgRD}, \text{RDS}, \boldsymbol{\Theta}) \quad = \textstyle\sum_{s=1}^{S} \pi_s(\text{RDS}, \boldsymbol{\alpha})\text{Poi}(\text{Patents}|\lambda_s),$$

where $\text{Poi}(\cdot|\lambda)$ denotes the Poisson distribution and

$$\log(\lambda_s) = \beta^1_{3,s} + \beta^2_{3,s}\text{lgRD}.$$

The R code for fitting this model is given in Figure 5.5. First, the data set is loaded. The component specific model and the concomitant variable model are specified and assigned to `Model.Patent.1` and `Conc`. This model is fitted with `stepFlexmix()` which returns the best of `nrep`=5 runs of the EM algorithm with random initialization.

The fitted values for the component specific models and the concomitant variable model are given in Figure 5.6 and the estimated parameters in Fig-

```
> refit(Fitted.2)

Call:
refit(Fitted.2)

Number of components: 3

$Comp.1
            Estimate Std. Error z value  Pr(>|z|)
x            2.00418    0.10094  19.856 < 2.2e-16
(Intercept) -4.28639    0.22399 -19.136 < 2.2e-16

$Comp.2
            Estimate Std. Error z value  Pr(>|z|)
x           2.004180   0.100937  19.856 < 2.2e-16
(Intercept) 1.005756   0.068438  14.696 < 2.2e-16

$Comp.3
            Estimate Std. Error z value  Pr(>|z|)
(Intercept)  2.89577    0.12146  23.841 < 2.2e-16

> refit(Fitted.2, which = "concomitant")

Call:
refit(Fitted.2, which = "concomitant")

Number of components: 3

$Comp.2
            Estimate Std. Error z value  Pr(>|z|)
(Intercept)  1.19610    0.25460  4.6979 2.629e-06
w1          -1.23522    0.39169 -3.1536  0.001613

$Comp.3
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.031706   0.318041 -0.0997  0.92059
w1           0.783599   0.406276  1.9287  0.05376
```

Figure 5.4: Parameters of the model with nested varying effects.

```
> data(patent)
> Model.Patent.1 <- FLXglm(family = "poisson")
> Conc <- FLXmultinom(~RDS)
> Fitted.Patent.1 <- stepFlexmix(Patents ~ lgRD, k = 3,
+     nrep = 5, model = Model.Patent.1, data = patent,
+     concomitant = Conc)
> Model.Patent.2 <- FLXglmFix(family = "poisson", fixed = ~lgRD)
> Posteriors <- posterior(Fitted.Patent.1)
> Fitted.Patent.2 <- flexmix(Patents ~ 1, model = Model.Patent.2,
+     cluster = Posteriors, data = patent, concomitant = Conc)
```

Figure 5.5: Fitting mixtures of Poisson regression models without constraints (`Model.Patent.1`) and with fixed effects (`Model.Patent.2`).

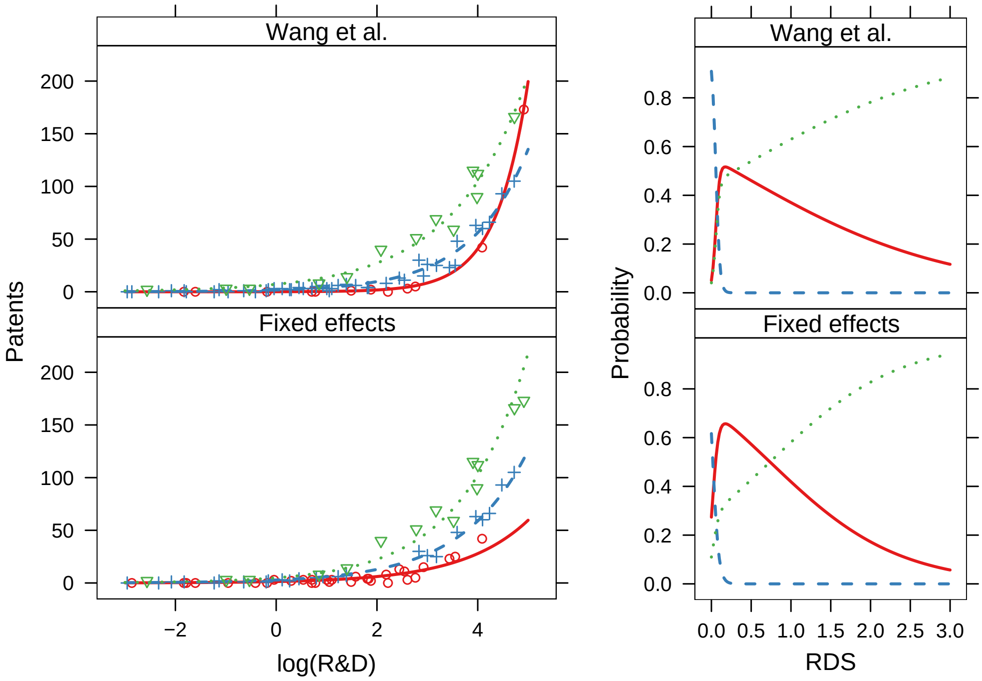


Figure 5.6: Patent data with the fitted values of the component specific models (left) and the concomitant variable model (right) for the model in Wang et al. and with fixed effects for $\log(R\&D)$. The plotting symbol for each observation is determined by the component with the maximum a-posteriori probability.

```
> refit(Fitted.Patent.1)

Call:
refit(Fitted.Patent.1)

Number of components: 3

$Comp.1
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -2.638876   0.404726 -6.5202 7.023e-11
lgRD         1.587006   0.089986 17.6362 < 2.2e-16

$Comp.2
            Estimate Std. Error z value  Pr(>|z|)
(Intercept) 0.507796   0.123663  4.1063 4.021e-05
lgRD        0.879831   0.033284 26.4343 < 2.2e-16

$Comp.3
            Estimate Std. Error z value  Pr(>|z|)
(Intercept) 1.962118   0.139667  14.049 < 2.2e-16
lgRD        0.671907   0.035712  18.815 < 2.2e-16

> refit(Fitted.Patent.1, which = "concomitant")

Call:
refit(Fitted.Patent.1, which = "concomitant")

Number of components: 3

$Comp.2
             Estimate Std. Error z value  Pr(>|z|)
(Intercept)   2.89107    0.62629  4.6161 3.909e-06
RDS         -40.22081   11.76771 -3.4179 0.0006311

$Comp.3
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.21417    0.41111 -0.5209   0.6024
RDS          0.74544    1.00438  0.7422   0.4580
```

Figure 5.7: Parameters of the model `Model.Patent.1` fitted to the `patent` data set with only varying effects.

ure 5.7. The plotting symbol of the observations corresponds to the induced clustering given by `cluster(Fitted.Patent.1)`.

We modify this model to have fixed effects for the logarithmized R&D spendings:

$$\log(\lambda_s) = \beta_{3,s} + \beta_1 \mathrm{lgRD}.$$

We use the already fitted model for initialization, i.e. we start the EM algorithm with an M-step given the a-posteriori probabilities. The R code for fitting this model (`Model.Patent.2`) is given in Figure 5.5.

With respect to the BIC the full model is better than the model with the fixed effects. However, fixed effects have the advantage that the different components differ only in their baseline and the relation between the components in return of investment for each additional unit of R&D spending is constant. Due to a-priori domain knowledge this model might seem more plausible. The fitted values for the constrained model are also given in Figure 5.6. The fitted parameters are given in Figure 5.8.

5.3.2 Writing your own drivers

New concomitant variable models can be defined by writing a constructor function for a `"FLXP"` object and new component specific models by writing a constructor function for an object which extends the class `"FLXM"`. Two examples are given in the following: concomitant variable models with only categorical variables and component-specific models where one component has a zero mean, i.e. this model class defines a zero-inflated Poisson or binomial regression model.

Concomitant variable models

If the concomitant variable is a categorical variable, the multinomial logit model is equivalent to a model where the component weights for each level of the concomitant variable are determined by the mean values of the a-posteriori probabilities. The driver which implements this `"FLXP"` model is

```
> refit(Fitted.Patent.2)

Call:
refit(Fitted.Patent.2)

Number of components: 3

$Comp.1
            Estimate Std. Error z value  Pr(>|z|)
lgRD        0.758960   0.023327  32.535 < 2.2e-16
(Intercept) 0.292339   0.097187   3.008  0.002630

$Comp.2
            Estimate Std. Error z value  Pr(>|z|)
lgRD        0.758960   0.023327  32.535 < 2.2e-16
(Intercept) 1.039521   0.096507  10.771 < 2.2e-16

$Comp.3
            Estimate Std. Error z value  Pr(>|z|)
lgRD        0.758960   0.023327  32.535 < 2.2e-16
(Intercept) 1.590995   0.100014  15.908 < 2.2e-16

> refit(Fitted.Patent.2, which = "concomitant")

Call:
refit(Fitted.Patent.2, which = "concomitant")

Number of components: 3

$Comp.2
             Estimate Std. Error z value Pr(>|z|)
(Intercept)   0.81202    0.47947  1.6936  0.09035
RDS         -26.00943   11.05348 -2.3531  0.01862

$Comp.3
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.91353    0.34713 -2.6317 0.008496
RDS          1.24043    1.40271  0.8843 0.376527
```

Figure 5.8: Parameters of the model `Model.Patent.2` fitted to the `patent` data set with fixed effects for `lgRD`.

```
setClass("FLXPaverage", contains = "FLXP")
myConcomitant <- function(formula = ~1) {
    z <- new("FLXPaverage", name = "myConcomitant",
        formula = formula)
    z@fit <- function(x, y, ...) {
        f <- as.integer(factor(apply(x, 1, paste,
            collapse = "")))
        AVG <- apply(y, 2, tapply, f, mean)
        (AVG/rowSums(AVG))[f, , drop = FALSE]
    }
    z
}
```

Figure 5.9: Driver for a concomitant variable model where the component weights are determined by averaging over the a-posteriori probabilities for each level of the concomitant variable.

given in Figure 5.9. A name for the driver has to be specified and a `fit()` function.

Example: Using the driver If the concomitant variable model returned by `myConcomitant()` is used for the artificial example in Section 5.3.1 the same fitted model is returned as if a multinomial logit model is specified. An advantage is that in this case there are no problems if the fitted probabilities are close to zero or one.

The fitting of the model with the new concomitant variable model is given in Figure 5.10. The estimated component weights are compared for each level of w using the function `prior()` with the fitted mixture where a multinomial logit model is used for the concomitant variable model. Obviously the fitted values of the two models correspond to each other.

Component models: zero-inflated models

In Poisson or binomial regression models it can be often encountered that the observed number of zeros is higher than expected. This can be modelled by a mixture with two components where one has mean 0 (see for example Böhning et al. 1999). This model can be even generalized to a mixture with more than two components where one component has a mean fixed at zero. In the following this component will be the first. This model can be defined for

```
> Conc <- myConcomitant(~w)
> Fitted.3 <- stepFlexmix(cbind(yes, no) ~ 1, data = BregFix,
+     cluster = posterior(Fitted.1[["3"]]), model = Model.2,
+     concomitant = Conc)

 * * *

> prior <- function(object) {
+     x <- object@concomitant@x
+     object@concomitant@fit(x, posterior(object))[!duplicated(x),
+         ]
+ }
> prior(Fitted.3)

       [,1]      [,2]      [,3]
1 0.1895370 0.6268413 0.1836217
2 0.2449401 0.2355387 0.5195212

> prior(Fitted.2)

       [,1]      [,2]      [,3]
1 0.1895382 0.6268390 0.1836229
3 0.2449390 0.2355429 0.5195182
```

Figure 5.10: Fitting a mixture model with the new concomitant model driver.

package **flexmix** by defining an appropriate model class with a construction function and model-dependent methods for the M-step, if a component is removed and to get the model matrix. In addition new methods for function `refit()` can be defined.

The model class `"FLXMRziglm"` extends `"FLXMRglm"` and for construction `FLXglm()` can be used. Only the family is restricted to binomial or Poisson, an appropriate name is assigend and the correct class returned. In order to ensure that the mean of the first component is equal to zero the model matrix has to contain an intercept and the coefficients are fixed to be -infinity for the intercept and zero for all other variables. The existence of the intercept is checked in `FLXgetModelmatrix()`. `FLXremoveComponent()` is called if one component is removed during the EM algorithm. It checks if this is the first and in this case the model is changed to `"FLXMRglm"`. In the M-step the coefficients of the first component are fixed and not estimated, while for the remaining components the M-step of `"FLXMRglm"` objects can be used. A similar modification is necessary for refitting the model.

```
setClass("FLXMRziglm", contains = "FLXMRglm")
FLXziglm <- function(formula = . ~ ., family = c("binomial",
    "poisson"), ...) {
    family <- match.arg(family)
    new("FLXMRziglm", FLXglm(formula, family, ...),
        name = paste("FLXziglm", family, sep = ":"))
}
setMethod("FLXgetModelmatrix", signature(model = "FLXMRziglm"),
    function(model, data, formula, lhs = TRUE, ...) {
        model <- callNextMethod(model, data, formula,
            lhs)
        if (attr(terms(model@fullformula), "intercept") ==
            0)
            stop("please include an intercept")
        new("FLXMRziglm", model)
    })
setMethod("FLXremoveComponent", signature(model = "FLXMRziglm"),
    function(model, nok, ...) {
        if (1 %in% nok)
            model <- as(model, "FLXMRglm")
        model
    })
setMethod("FLXmstep", signature(model = "FLXMRziglm"),
    function(model, weights, ...) {
        coef <- c(-Inf, rep(0, ncol(model@x) - 1))
        names(coef) <- colnames(model@x)
        comp.1 <- with(list(coef = coef, df = 0,
            offset = NULL, family = model@family),
            eval(model@defineComponent))
        c(list(comp.1), FLXmstep(as(model, "FLXMRglm"),
            weights[, -1, drop = FALSE]))
    })
setClass("FLXRMRziglm", contains = "FLXRM")
setMethod("refit", signature(object = "FLXMRziglm"),
    function(object, weights, ...) {
        coef <- c(-Inf, rep(0, ncol(object@x) - 1))
        names(coef) <- colnames(object@x)
        comp.1 <- new("FLXRMRziglm", fitted = list(coef))
        c(list(comp.1), callNextMethod(object, weights[,
            -1, drop = FALSE]))
    })
```

Figure 5.11: Driver for a zero-inflated component specific model.

```
> data(dmft)
> Model <- FLXziglm(family = "poisson")
> Fitted <- flexmix(End ~ log(Begin + 0.5) + Gender +
+     Ethnic + Treatment, model = Model, k = 2, data = dmft,
+     control = list(minprior = 0.01))
```

Figure 5.12: Fitting a zero-inflated Poisson model to the `dmft` data set

Example: Using the driver This new M-step driver can be used to estimate a zero-inflated Poisson model to the data given in Böhning et al. (1999). The data set `dmft` is count data from a dental epidemiological study for evaluation of various programs for reducing caries collected among school children from an urban area of Belo Horizonte (Brazil). It includes the number of decayed, missing or filled teeth at the beginning and at the end of the observation period, the gender, the ethnic background and the treatment of 797 children.

The commands for fitting this model are given in Figure 5.12 and the estimated coefficients are given in Figure 5.13. The estimated coefficients with corresponding confidence intervals are also given in Figure 5.14. As the coefficients of the first component are not interesting, we plot only the second component. In this case scaling of the coefficients does not make sense. The box ratio modified can be modified as for `barchart()` in package **lattice**. The code to produce this figure is given by `plot(refit(Fitted), components = 2, scale = FALSE)`.

5.4 Summary

The modifications of package **flexmix** to allow for concomitant variable models and mixtures of regressions of varying and fixed effects as well as the implementation of the new plot methods brought the package one step closer to the final aim which is to be a toolbox for fitting general finite mixture models providing functionality for fitting the models as well as model selection or diagnostics.

Missing features are for example automated diagnostic tools based on

```
> refit(Fitted)

Call:
refit(Fitted)

Number of components: 2

$Comp.1
     (Intercept) log(Begin + 0.5)       Gendermale      Ethnicwhite
            -Inf                0                0                0
     Ethnicblack    Treatmenteduc     Treatmentall Treatmentenrich
               0                0                0                0
  Treatmentrinse Treatmenthygiene
               0                0

$Comp.2
                  Estimate Std. Error z value  Pr(>|z|)
(Intercept)      -0.146671   0.092856 -1.5796 0.1142097
log(Begin + 0.5)  0.730110   0.039221 18.6154 < 2.2e-16
Gendermale        0.006799   0.052999  0.1283 0.8979235
Ethnicwhite       0.050307   0.057455  0.8756 0.3812533
Ethnicblack      -0.046808   0.087302 -0.5362 0.5918479
Treatmenteduc    -0.236686   0.087209 -2.7140 0.0066476
Treatmentall     -0.327512   0.096234 -3.4033 0.0006658
Treatmentenrich   0.017153   0.081894  0.2095 0.8340948
Treatmentrinse   -0.240981   0.084555 -2.8500 0.0043721
Treatmenthygiene -0.102740   0.089279 -1.1508 0.2498288
```

Figure 5.13: Parameters of the zero-inflated Poisson model.

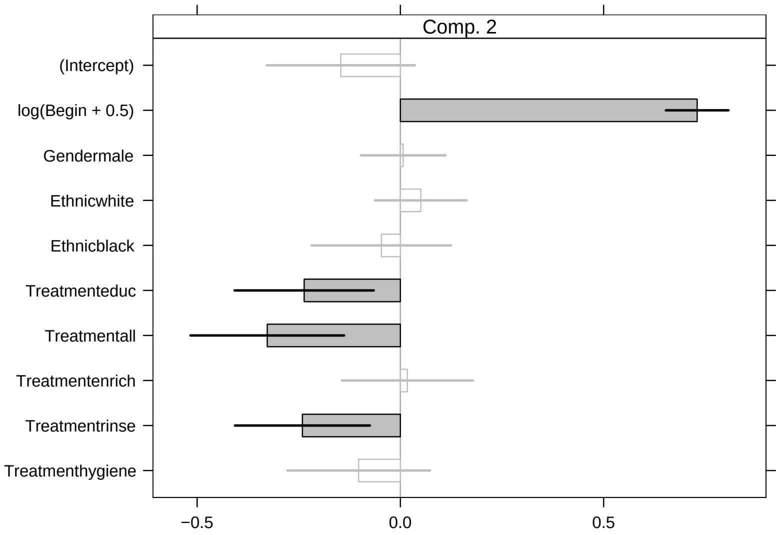


Figure 5.14: The estimated coefficients of the zero-inflated model for the `dmft` data set.

resampling methods as bootstrap results might give valuable insights into the model fit (see Section 4). To improve the model matrix determination and the data management for repeated estimation of the same model with different data package **modeltools** (Hothorn et al. 2006) should be integrated simultaneously with the implementation of the bootstrap methods.

To provide functionality for fitting zero-inflated Poisson and binomial regression models is a first step towards relaxing the assumption that all component specific distributions are from the same parametric family. As mixtures with components which follow distributions from different parametric families can be useful for example to model outliers (Dasgupta and Raftery 1998) it would be nice to also have this functionality readily available in **flexmix**.

Chapter 6

Conclusions

Finite mixture models have been used for more than 100 years, but have seen a real boost in popularity over the last decades due to the tremendous increase in available computing power. Applications in disjoint scientific communities have led to the development of a lot of variants and extensions for special cases without proper analysis of many structural and statistical properties of the general model class. The EM algorithm provides a unifying framework for maximum likelihood estimation of parameters. However, the identification of these models was only considered for special cases and a thorough investigation of recent extensions and variants, as, e.g., mixtures of generalized linear models, is still missing.

This thesis tries to fill the present gaps in research by providing sufficient identifiability constraints for important model classes included in the GLIMMIX framework, where mixtures of generalized linear models are defined. Mixtures of Gaussian, Poisson and binomial/multinomial regression models are popular in applications and different kinds of constraints are used to model the regression coefficients leading to a framework of varying and fixed effects of the regression coefficients and dispersion parameters. Theoretic identifiability constraints can be used to formally check if a given model class is theoretically identifiable. In addition they can also indicate how much information is needed to be included in the data in order to be able to sensibly estimate complex and flexible models such as finite mixture models. This is especially true for binomial or multinomial logit models where the repetition parameters plays a crucial role in determining the maximum number of components which can be distinguished. If there are not enough repetitions available, finite mixture models can still be estimated and can also give good results. However, one has to be aware that heterogeneity can potentially remain undetected as the true underlying data generating process is observational equivalent to a mixture with less components given the available data.

While a theoretic understanding of the model class is important, further insights can be gained using resampling methods for model diagnostics. In a frequentist maximum likelihood setting resampling methods can be used to detect competing model parameterizations for the given data which can all be considered to describe the true underlying data generating process. In Bayesian modelling the posterior probabilities are supposed to convey the same information given that the MCMC sampler used for estimating these distributions moves around the whole parameter space and visits all modes.

The constrained clustering approach together with the relabelling algorithm under the assumption of multi-modality are an important tool to simultaneously determine the different modes present together with a suitable labelling of the respective components. The constrained clustering method even has the advantage that it reveals if there are only a subset of components where different competing parameterizations can be given.

A thorough analysis of the model fit is necessary in order to be able to select the most appropriate model to describe the underlying data generating process or to gain important insights. Depending on the application and the modelling aims the focus will be on the estimated parameters of the components, the component weights or the a-posteriori probabilities, i.e. the implied clusterings. The diagnostics should be suitably chosen depending on the purpose. Different aspects are covered in this thesis and the application for analyzing the model is demonstrated on several examples.

A flexible implementation for model estimation and the availability of tools for convenient comparison of different models is important for the development of new models and to gain further insights into the fitted models.

Especially graphical tools are invaluable in conveying important information contained in the data or the fitted model. The statistical computing environment R is clearly suited and the R package **flexmix** is aimed to be a computational toolbox for flexible finite mixture modelling with the EM algorithm. New component specific models can be easily defined and different concomitant variable models are possible. As the EM algorithm provides a common framework for estimation, the fitted models are also from the same class and model analysis methods which are component specific model or concomitant variable model independent, such as for example the investigation of a-posteriori probabilities or the induced clusterings, are applicable for all these models. This provides the opportunity to reuse a lot of the existing code for new mixture models and provide only a few additional methods which take care of the peculiarities of the new component specific or concomitant variable model.

Appendix A

Proof of Theorem 2

If the model is not identifiable, there exist two different parameterizations $\boldsymbol{\Theta}$ and $\hat{\boldsymbol{\Theta}}$ with at most S components such that

$$H(\cdot \mid \boldsymbol{X}, \boldsymbol{Z}, \boldsymbol{\Theta}) \equiv H(\cdot \mid \boldsymbol{X}, \boldsymbol{Z}, \hat{\boldsymbol{\Theta}})$$

where $\boldsymbol{\Theta} = (\pi_l, \boldsymbol{\beta}^l, \boldsymbol{\gamma})_{l=1,\dots,s}$ and $\hat{\boldsymbol{\Theta}} = (\hat{\pi}_m, \hat{\boldsymbol{\beta}}^m, \hat{\boldsymbol{\gamma}})_{m=1,\dots,\hat{s}}$

With condition (1a) we show that the binomial distributions with alternatives $\{k, K\}$ are identifiable $\forall i^* \in \tilde{I}_k\ \forall k$ (Step (a) and (b)). The covariate points where binomial identifiability was shown can be used to prove that no intra-component label switching is possible (Step (c)). This gives us that the coefficients of the fixed and varying effects are identical up to arbitrary constants. The rank condition is needed to prove that the constants are equal to zero (Step (d)).

(a) We show that $\forall k = 1, \dots, K$:

$$\forall i^* \in \tilde{I}_k : \boldsymbol{z}'_{k,j}(\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}) = c_{k,i^*} \quad \forall j \in \bigcup_{i \in E_{k,i^*}} J_i \tag{A.1}$$

(b) We show that given an arbitrary $k \in \{1, \dots, K-1\}$ it holds $\forall i^* \in \tilde{I}_k$ that $\hat{s}(i^*) = s(i^*)$ and that there exists a suitable ordering of the components such that:

$$\alpha^l_{k,i^*} = \hat{\alpha}^l_{k,i^*} + c_{k,i^*} \tag{A.2}$$

$\forall l = 1, \ldots, s(i^*)$ with

$$\alpha^l_{k,i^*} \in \{\boldsymbol{x}'_{k,i^*}\boldsymbol{\beta}^u : u = 1, \ldots, s\}$$

and $\hat{\alpha}^l_{k,i^*}$ analogously defined.

(c) We show with condition (1b) analogously to Hennig (2000) that $\hat{s} = s$ and that for a suitable ordering of the components it holds that $\forall l = 1, \ldots, s$:

$$\hat{\pi}_l = \pi_l \tag{A.3}$$

$$\hat{\boldsymbol{\beta}}^l = \boldsymbol{\beta}^l + \boldsymbol{\delta} \tag{A.4}$$

where $\boldsymbol{\delta} \in \mathbb{R}^U$ is suitably chosen.

(d) We show $\boldsymbol{\delta} = \mathbf{0}$ and $\boldsymbol{\gamma} = \hat{\boldsymbol{\gamma}}$.

ad (a): The equation trivially holds for $k = K$ and the following holds $\forall k = 1, \ldots, K-1$. If the mixture distributions are equivalent, this equivalence must also hold for a subset of the covariate points. Hence, we have $\forall i^* \in \tilde{I}_k$:

$$\sum_{l=1}^{s} \pi_l \prod_{i \in E_{k,i^*}} \prod_{j \in J_i} F(\boldsymbol{y}_{ij}; N_{ij}, \boldsymbol{\theta}^l_{ij}) \;=\; \sum_{m=1}^{\hat{s}} \hat{\pi}_m \prod_{i \in E_{k,i^*}} \prod_{j \in J_i} F(\boldsymbol{y}_{ij}; N_{ij}, \hat{\boldsymbol{\theta}}^m_{ij})$$

The following holds $\forall u \in E_{k,i^*}$ and $v \in J_u$ where $(\boldsymbol{y}_{ij})_{j \in J_i, i \in E_{k,i^*}}$ is given by $y_{k,ij} = \delta_{iu,jv}$ and $y_{K,ij} = N_{ij} - y_{k,ij}$. $\delta_{iu,jv}$ is the Kronecker delta, i.e. it is one if $i = u$ and $j = v$ and zero otherwise.

If we cancel the multinomial coefficients on both sides, we have:

$$\sum_{l=1}^{s}\pi_l\left[e^{\alpha^l_{k,i^*}+\boldsymbol{z}'_{k,v}\boldsymbol{\gamma}}\prod_{i\in E_{k,i^*}}\prod_{j\in J_i}(\sum_{h=1}^{K}e^{\alpha^l_{h,i}+\boldsymbol{z}'_{h,j}\boldsymbol{\gamma}})^{-N_{ij}}\right]=$$
$$\sum_{m=1}^{\hat{s}}\hat{\pi}_m\left[e^{\hat{\alpha}^m_{k,i^*}+\boldsymbol{z}'_{k,v}\hat{\boldsymbol{\gamma}}}\prod_{i\in E_{k,i^*}}\prod_{j\in J_i}(\sum_{h=1}^{K}e^{\hat{\alpha}^m_{h,i}+\boldsymbol{z}'_{h,j}\hat{\boldsymbol{\gamma}}})^{-N_{ij}}\right]$$

The terms which do not depend on l or m can be taken out of the sums. This leads to:

$$e^{\boldsymbol{z}'_{k,v}(\boldsymbol{\gamma}-\hat{\boldsymbol{\gamma}})}\sum_{l=1}^{s}\pi_l\left[e^{\alpha^l_{k,i^*}}\prod_{i\in E_{k,i^*}}\prod_{j\in J_i}(\sum_{h=1}^{K}e^{\alpha^l_{h,i}+\boldsymbol{z}'_{h,j}\boldsymbol{\gamma}})^{-N_{ij}}\right]=$$
$$\sum_{m=1}^{\hat{s}}\hat{\pi}_m\left[e^{\hat{\alpha}^m_{k,i^*}}\prod_{i\in E_{k,i^*}}\prod_{j\in J_i}(\sum_{h=1}^{K}e^{\hat{\alpha}^m_{h,i}+\boldsymbol{z}'_{h,j}\hat{\boldsymbol{\gamma}}})^{-N_{ij}}\right]$$

Finally we have

$$e^{\boldsymbol{z}'_{k,v}(\boldsymbol{\gamma}-\hat{\boldsymbol{\gamma}})}=\frac{\sum_{m=1}^{\hat{s}}\hat{\pi}_m\left[e^{\hat{\alpha}^m_{k,i^*}}\prod_{i\in E_{k,i^*}}\prod_{j\in J_i}(\sum_{h=1}^{K}e^{\hat{\alpha}^m_{h,i}+\boldsymbol{z}'_{h,j}\hat{\boldsymbol{\gamma}}})^{-N_{ij}}\right]}{\sum_{l=1}^{s}\pi_l\left[e^{\alpha^l_{k,i^*}}\prod_{i\in E_{k,i^*}}\prod_{j\in J_i}(\sum_{h=1}^{K}e^{\alpha^l_{h,i}+\boldsymbol{z}'_{h,j}\boldsymbol{\gamma}})^{-N_{ij}}\right]}$$

where the right hand side does not depend on index v which signifies that the left hand side is constant $\forall v \in \bigcup_{u\in E_{k,i^*}} J_u$ given i^*. This constant can be given by $e^{c_{k,i^*}}$. Hence, we have shown equation (A.1).

ad (b): For a given k and i^* we define $y_{k,\cdot\cdot} := \sum_{i\in E_{k,i^*}}\sum_{j\in J_i} y_{k,ij}$. In the following we insert the dependent variable where $y_{K,ij} = N_{ij} - y_{k,ij}$.

Then it holds $\forall i^* \in \tilde{I}_k$:

$$\sum_{l=1}^{s} \pi_l \frac{e^{y_{k,..}\alpha^l_{k,i^*} + \sum_{i \in E_{k,i^*}} \sum_{j \in J_i} y_{k,ij} \boldsymbol{z}'_{k,j} \boldsymbol{\gamma}}}{\prod\limits_{i \in E_{k,i^*}} \prod\limits_{j \in J_i} \left(\sum\limits_{h=1}^{K} e^{\alpha^l_{h,i} + \boldsymbol{z}'_{h,j} \boldsymbol{\gamma}} \right)^{N_{ij}}} =$$

$$\sum_{m=1}^{\hat{s}} \hat{\pi}_m \frac{e^{y_{k,..}\hat{\alpha}^m_{k,i^*} + \sum_{i \in E_{k,i^*}} \sum_{j \in J_i} y_{k,ij} \boldsymbol{z}'_{k,j} \boldsymbol{\gamma} + y_{k,..} c_{k,i^*}}}{\prod\limits_{i \in E_{k,i^*}} \prod\limits_{j \in J_i} \left(\sum\limits_{h=1}^{K} e^{\hat{\alpha}^m_{h,i} + \boldsymbol{z}'_{h,j} \boldsymbol{\gamma} + c_{h,i^*}} \right)^{N_{ij}}} \tag{A.5}$$

As the denominator on the left hand side only depends on i^* and l and not on j and $y_{k,ij}$, we define:

$$\lambda^l_{k,i^*} := \frac{\pi_l}{\prod\limits_{i \in E_{k,i^*}} \prod\limits_{j \in J_i} \left(\sum\limits_{h=1}^{K} e^{\alpha^l_{h,i} + \boldsymbol{z}'_{h,j} \boldsymbol{\gamma}} \right)^{N_{ij}}}$$

and $\hat{\lambda}^m_{k,i^*}$ can be accordingly defined.

Substituting λ^l_{k,i^*} and $\hat{\lambda}^m_{k,i^*}$ into equation (A.5) and eliminating the equal terms on the left and right hand side gives $\forall i^* \in \tilde{I}_k$:

$$\sum_{l=1}^{s} \lambda^l_{k,i^*} \left(e^{\alpha^l_{k,i^*}} \right)^{y_{k,..}} = \sum_{m=1}^{\hat{s}} \hat{\lambda}^m_{k,i^*} \left(e^{\hat{\alpha}^m_{k,i^*} + c_{k,i^*}} \right)^{y_{k,..}} \tag{A.6}$$

with $y_{k,..} \in \{0, \dots, \sum_{i \in E_{k,i^*}} \sum_{j \in J_i} N_{ij}\}$.

With condition (1a) it follows that the sum over the unique elements in equation (A.6) has only the trivial solution $\forall i^* \in \tilde{I}_k$ which signifies that equation (A.2) hold. This also means that the k^{th} marginal binomial distribution with alternatives $\{k, K\}$ is identifiable in point $\boldsymbol{x}_{k,i^*}$.

ad (c): We assume that there can be a $\tilde{\boldsymbol{\beta}}^l$ defined $\forall l$ such that it holds $\forall i \in \tilde{I}_k$ given $k \in \{1, \dots, K-1\}$:

$$\tilde{\boldsymbol{X}}_{k,i} \boldsymbol{\beta}^l + \tilde{\boldsymbol{Z}}_{k,i} \boldsymbol{\gamma} = \tilde{\boldsymbol{X}}_{k,i} \tilde{\boldsymbol{\beta}}^l + \tilde{\boldsymbol{Z}}_{k,i} \hat{\boldsymbol{\gamma}}$$

where $\tilde{\boldsymbol{X}}_{k,i} := (\boldsymbol{x}'_{k,i})_{j \in J_i}$ and $\tilde{\boldsymbol{Z}}_{k,i}$ is analogously defined.

The existence of these $\tilde{\boldsymbol{\beta}}^l$ is guaranteed because the following equation holds due to the fact that the inverse logit function is a one-to-one mapping (due to condition (3)) and that $\forall k = 1, \ldots, K-1$ the k^{th} marginal binomial distribution with alternatives $\{k, K\}$ is identifiable $\forall i \in \tilde{I}_k$:

$$\begin{aligned}
\sum_{l=1}^{s} \pi_l \Big(\tilde{\boldsymbol{X}}_{\tilde{I}} \boldsymbol{\beta}^l + \tilde{\boldsymbol{Z}}_{\tilde{I}} \boldsymbol{\gamma} \Big) &= \sum_{m=1}^{\hat{s}} \hat{\pi}_m \Big(\tilde{\boldsymbol{X}}_{\tilde{I}} \hat{\boldsymbol{\beta}}^m + \tilde{\boldsymbol{Z}}_{\tilde{I}} \hat{\boldsymbol{\gamma}} \Big) \\
\tilde{\boldsymbol{X}}_{\tilde{I}} \Bigg(\sum_{l=1}^{s} \pi_l \boldsymbol{\beta}^l - \sum_{m=1}^{\hat{s}} \hat{\pi}_m \hat{\boldsymbol{\beta}}^m \Bigg) &= \tilde{\boldsymbol{Z}}_{\tilde{I}} (\hat{\boldsymbol{\gamma}} - \boldsymbol{\gamma})
\end{aligned}$$

where $\tilde{\boldsymbol{X}}_{\tilde{I}} := (\tilde{\boldsymbol{X}}_{k,i})_{i \in \tilde{I}_k, k=1,\ldots,K-1}$ and $\tilde{\boldsymbol{Z}}_{\tilde{I}}$ is analogously defined.

As because of condition (1b) $\tilde{\boldsymbol{X}}_{\tilde{I}}$ has full column rank we can define

$$\begin{aligned}
\tilde{\boldsymbol{\beta}}^l &:= \boldsymbol{\beta}^l + \boldsymbol{\delta} \qquad \text{with} \\
\boldsymbol{\delta} &:= \Big(\tilde{\boldsymbol{X}}'_{\tilde{I}} \tilde{\boldsymbol{X}}_{\tilde{I}} \Big)^{-1} \tilde{\boldsymbol{X}}'_{\tilde{I}} \tilde{\boldsymbol{Z}}_{\tilde{I}} (\boldsymbol{\gamma} - \hat{\boldsymbol{\gamma}}).
\end{aligned}$$

We assume without loss of generality that

$$\pi_1 \neq \hat{\pi}_1 \qquad \text{and} \qquad s \geq \hat{s} \tag{A.7}$$

where $\hat{\pi}_1$ is the a-priori probability for $\tilde{\boldsymbol{\beta}}^1$ with $\hat{\pi}_1 \geq 0$.

As the marginal binomial mixture distributions for $k = 1, \ldots, K-1$ with alternatives $\{k, K\}$ are identifiable $\forall i \in \tilde{I}_k$, the following must hold $\forall i \in \tilde{I}_k$ $\forall k = 1, \ldots, K-1$:

$$\sum_{\substack{\forall l=1,\ldots,\hat{s}: \\ \boldsymbol{x}'_{k,i} \hat{\boldsymbol{\beta}}^l = \boldsymbol{x}'_{k,i} \tilde{\boldsymbol{\beta}}^1}} \hat{\pi}_l = \sum_{\substack{\forall h=1,\ldots,s: \\ \boldsymbol{x}'_{k,i} \boldsymbol{\beta}^h = \boldsymbol{x}'_{k,i} \boldsymbol{\beta}^1}} \pi_h \tag{A.8}$$

The assumption $S < q^*$ would be in contradiction to the existence of some $\tilde{\boldsymbol{\beta}} \in \{\tilde{\boldsymbol{\beta}}^u : u = 1, \ldots, s\}$ such that $\forall k = 1, \ldots, K-1$: $\forall i^* \in \tilde{I}_k$:

$\exists l \in \{1, \dots, s\}$ with

$$\tilde{\boldsymbol{\beta}} \neq \hat{\boldsymbol{\beta}}^l \quad \wedge \quad \boldsymbol{x}'_{k,i}\tilde{\boldsymbol{\beta}} = \boldsymbol{x}'_{k,i}\hat{\boldsymbol{\beta}}^l \quad \forall k \in \{1, \dots, K-1\} \wedge i \in I_{t(i^*)} \cap \tilde{I}_k$$

because then $q^* \leq s \leq S$ would hold.

Thus it holds for all $\tilde{\boldsymbol{\beta}}^l$ $l = 1, \dots, s$— and in particular for $\tilde{\boldsymbol{\beta}}^1$—that there exists a $k^* = k(\tilde{\boldsymbol{\beta}}^l)$ and $i^* = i(\tilde{\boldsymbol{\beta}}^l) \in \tilde{I}_k$ such that:

$$\forall \hat{\boldsymbol{\beta}} \in \{\hat{\boldsymbol{\beta}}^m : m = 1, \dots, \hat{s}\} : \boldsymbol{x}'_{k^*,i^*}\tilde{\boldsymbol{\beta}}^l = \boldsymbol{x}'_{k^*,i^*}\hat{\boldsymbol{\beta}} \qquad \Rightarrow \qquad \tilde{\boldsymbol{\beta}}^l = \hat{\boldsymbol{\beta}}$$

Considering the marginal mixture distribution for $k^* := k(\tilde{\boldsymbol{\beta}}^1)$ and $i^* := i(\tilde{\boldsymbol{\beta}}^1)$, we have $\forall l \in \{1, \dots, \hat{s}\}$:

$$\hat{\boldsymbol{\beta}}^l \neq \tilde{\boldsymbol{\beta}}^1 \qquad \Rightarrow \qquad \boldsymbol{x}'_{k^*,i^*}\hat{\boldsymbol{\beta}}^l \neq \boldsymbol{x}'_{k^*,i^*}\tilde{\boldsymbol{\beta}}^1 \tag{A.9}$$

Thus, using condition (A.8),

$$\hat{\pi}_1 = \sum_{\substack{\forall h=1,\dots,s: \\ \boldsymbol{x}'_{k^*,i^*}\boldsymbol{\beta}^h = \boldsymbol{x}'_{k^*,i^*}\boldsymbol{\beta}^1}} \pi_h$$

implying $\hat{\pi}_1 > 0$.

Because of (A.7) — $\pi_1 \neq \hat{\pi}_1$ — it must hold:

$$\exists h \in \{2, \dots, s\} : \boldsymbol{\beta}^h \neq \boldsymbol{\beta}^1 \wedge \boldsymbol{x}'_{k^*,i^*}\boldsymbol{\beta}^h = \boldsymbol{x}'_{k^*,i^*}\boldsymbol{\beta}^1 \tag{A.10}$$

Without loss of generality one can assume that this h equals 2.

Consider $\boldsymbol{x}_{k,i} = \boldsymbol{x}_{k(\tilde{\boldsymbol{\beta}}^2),i(\tilde{\boldsymbol{\beta}}^2)}$ and apply the arguments above again to get $\exists l$: $\hat{\beta}^l = \tilde{\beta}^2$. This leads to a contradiction between (A.9) and (A.10). Hence we have shown equation (A.4).

ad (d): As equality of distributions implies equality of means we have

$$\sum_{l=1}^{s} \pi_l \frac{e^{\boldsymbol{x}'_{k,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{k,j}\boldsymbol{\gamma}}}{\sum_{h=1}^{K} e^{\boldsymbol{x}'_{h,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{h,j}\boldsymbol{\gamma}}} = \sum_{l=1}^{s} \pi_l \frac{e^{\boldsymbol{x}'_{k,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{k,j}\hat{\boldsymbol{\gamma}} + \boldsymbol{x}'_{k,i}\boldsymbol{\delta}}}{\sum_{h=1}^{K} e^{\boldsymbol{x}'_{h,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{h,j}\hat{\boldsymbol{\gamma}} + \boldsymbol{x}'_{h,i}\boldsymbol{\delta}}}$$

$\forall k = 1, \ldots, K;\ \forall j \in J_i; \forall i \in I.$

The equation above can be transformed to:

$$\sum_{l=1}^{s} \pi_l e^{\boldsymbol{x}'_{k,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{k,j}\boldsymbol{\gamma}} \\ \left[\frac{1}{\sum_{h=1}^{K} e^{\boldsymbol{x}'_{h,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{h,j}\boldsymbol{\gamma}}} - \frac{e^{\boldsymbol{x}'_{k,i}\boldsymbol{\delta} + \boldsymbol{z}'_{k,j}\boldsymbol{\vartheta}}}{\sum_{h=1}^{K} e^{\boldsymbol{x}'_{h,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{h,j}\boldsymbol{\gamma} + \boldsymbol{x}'_{h,i}\boldsymbol{\delta} + \boldsymbol{z}'_{h,j}\boldsymbol{\vartheta}}}\right] = 0$$

$$\sum_{l=1}^{s} \pi_l e^{\boldsymbol{x}'_{k,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{k,j}\boldsymbol{\gamma}} \left[\left(\sum_{h=1}^{K} e^{\boldsymbol{x}'_{h,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{h,j}\boldsymbol{\gamma}}\right)^{-1} - \right. \\ \left. \left(\sum_{h=1}^{K} e^{\boldsymbol{x}'_{h,i}\boldsymbol{\beta}^l + \boldsymbol{z}'_{h,j}\boldsymbol{\gamma} + (\boldsymbol{x}_{h,i} - \boldsymbol{x}_{k,i})'\boldsymbol{\delta} + (\boldsymbol{z}_{h,j} - \boldsymbol{z}_{k,j})'\boldsymbol{\vartheta}}\right)^{-1}\right] = 0$$

$\forall k = 1, \ldots, K;\ \forall j \in J_i$ and $\forall i \in I$ with $\boldsymbol{\vartheta} := \hat{\boldsymbol{\gamma}} - \boldsymbol{\gamma}$.

For every $j \in J_i, i \in I$ there can be a $\tilde{u}_{ij}$ defined with

$$\tilde{u}_{ij} \in \arg\max_{k=1,\ldots,K} \left\{ \boldsymbol{x}'_{k,i}\boldsymbol{\delta} + \boldsymbol{z}'_{k,j}\boldsymbol{\vartheta} \right\}$$

This choice of $\tilde{u}_{ij}$ implies

$$\boldsymbol{x}'_{\tilde{u}_{ij},i}\boldsymbol{\delta} + \boldsymbol{z}'_{\tilde{u}_{ij},j}\boldsymbol{\vartheta} \geq \boldsymbol{x}'_{k,i}\boldsymbol{\delta} + \boldsymbol{z}'_{k,j}\boldsymbol{\vartheta} \quad \forall k = 1, \ldots, K$$

We will contradict the following assumption:

$$\exists k \in \{1, \ldots, K\} : \boldsymbol{x}'_{k,i}\boldsymbol{\delta} + \boldsymbol{z}'_{k,j}\boldsymbol{\vartheta} \neq \mathbf{0} \tag{A.11}$$

This assumption together with the normalization condition (3), which implies $\boldsymbol{x}'_{K,i}\boldsymbol{\delta} + \boldsymbol{z}'_{K,j}\boldsymbol{\vartheta} = 0$, gives that there exists a $\tilde{v}_{ij} \in \{1, \ldots, K\}$ for which it holds

$$\boldsymbol{x}'_{\tilde{u}_{ij},i}\boldsymbol{\delta} + \boldsymbol{z}'_{\tilde{u}_{ij},j}\boldsymbol{\vartheta} > \boldsymbol{x}'_{\tilde{v}_{ij},i}\boldsymbol{\delta} + \boldsymbol{z}'_{\tilde{v}_{ij},j}\boldsymbol{\vartheta}$$

And therefore we get

$$\sum_{h=1}^{K} e^{\boldsymbol{x}'_{h,i}\boldsymbol{\beta}^l+\boldsymbol{z}'_{k,j}\boldsymbol{\gamma}} > \sum_{h=1}^{K} e^{\boldsymbol{x}'_{h,i}\boldsymbol{\beta}^l+\boldsymbol{z}'_{k,j}\boldsymbol{\gamma}+(\boldsymbol{x}_{h,i}-\boldsymbol{x}_{\tilde{u}_{ij},i})'\boldsymbol{\delta}+(\boldsymbol{z}_{h,j}-\boldsymbol{z}_{\tilde{u}_{ij},j})'\boldsymbol{\vartheta}}$$

$\forall l = 1, \ldots, s.$

This leads to a contradiction of assumption (A.11), because a linear combination of negative numbers using only positive coefficients cannot give 0. This means that $\boldsymbol{x}'_{k,i}\boldsymbol{\delta} - \boldsymbol{z}'_{k,j}\boldsymbol{\vartheta} = \mathbf{0}$ $\forall k$; $\forall j \in J_i$ and $\forall i \in I$. Because of condition (2) it follows $\boldsymbol{\delta} = \boldsymbol{\vartheta} = \mathbf{0}$. Hence we get $\hat{\boldsymbol{\gamma}} = \boldsymbol{\gamma}$ and $\hat{\boldsymbol{\beta}}^l = \boldsymbol{\beta}^l$ $\forall l = 1, \ldots, s$ and $\forall k = 1, \ldots, K$.

Appendix B

Exemplary applications of mixtures of regression models

In the following, data sets from different areas such as medicine, biology and economics are used. There are three sections: for finite mixtures of Gaussian regressions, for finite mixtures of binomial regression models and for finite mixtures of Poisson regression models.

B.1 Gaussian regressions

This artificial data set with 200 observations is given in Grün and Leisch (2006b). The data is generated from a mixture of Gaussian regression models with three components. There is an intercept with varying effects, an independent variable x_1, which is a categorical variable with two levels, with nested effects and another independent variable x_2, which is a numeric variable, with fixed effects. The prior probabilities depend on a concomitant variable w, which is also a categorical variable with two levels. Fixed effects are also assumed for the variance. The data is illustrated in Figure B.1 and the true underlying model is given by:

$$H(y \,|\, (x_1, x_2), w, \boldsymbol{\Theta}) = \sum_{s=1}^{S} \pi_s(w, \boldsymbol{\alpha}) \mathrm{N}(y \,|\, \mu_s, \sigma^2),$$

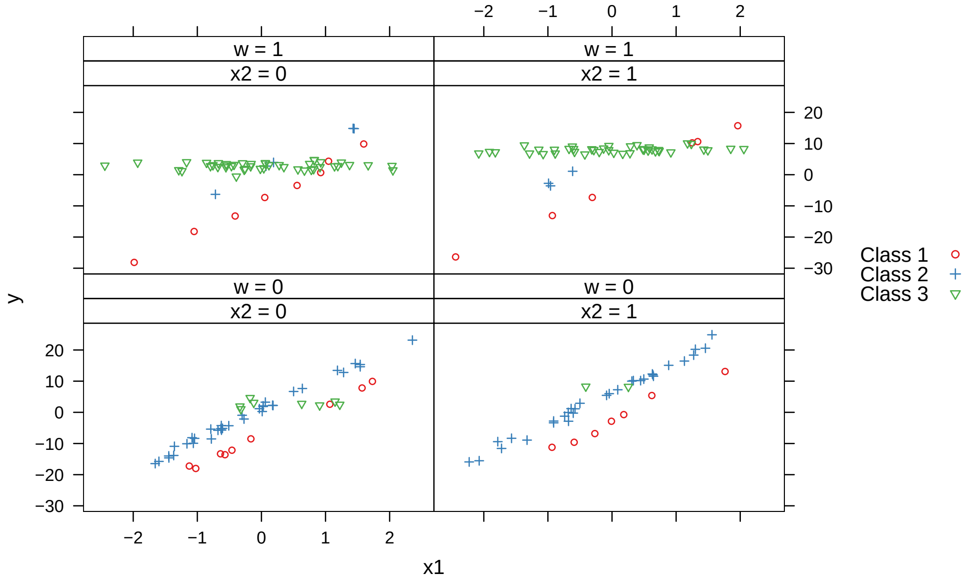


Figure B.1: Sample with 200 observations from the artificial example of finite mixtures of Gaussian regression models.

with $\boldsymbol{\beta}^s = (\beta^s_{\text{Intercept}}, \beta^{c(s)}_{x_1}, \beta_{x_2})$. The nesting signifies that $c(1) = c(2)$ and $\beta^{c(3)}_{x_1} = 0$.

The mixture model is fitted by first loading the package and the data set and then specifying the component specific model. In a first step a component specific model with only varying effects is specified. Then the fitting function `flexmix()` is called repeatedly using `stepFlexmix()`. The code is given in Figure B.2.

The estimated coefficients indicate that the components differ for the intercept, but that they are not significantly different for the coefficients of x_2. For x_1 the coefficient of the third component is not significantly different form zero and the confidence intervals for the other two components overlap. Therefore we fit a modified model, which is equivalent to the true underlying model. The original fitted model is used for initializing the EM algorithm (see Figure B.2). The BIC suggests that the restricted model should be preferred.

Details of the second fitted model are given in Figure B.3. The coeffi-

```
> library(flexmix)
> data(NregFix)
> Model <- FLXglm(~x2 + x1)
> fittedModel <- stepFlexmix(y ~ 1, model = Model,
+     nrep = 5, k = 3, data = NregFix, concomitant = FLXmultinom(~w))

 *Loading required package: nnet
 * * * *

> refit(fittedModel)

Call:
refit(fittedModel)

Number of components: 3

$Comp.1
             Estimate Std. Error t value  Pr(>|t|)
(Intercept) -7.641037   0.092629 -82.491 < 2.2e-16
x21          4.648780   0.141118  32.942 < 2.2e-16
x1           9.935412   0.061812 160.737 < 2.2e-16

$Comp.2
            Estimate Std. Error  t value  Pr(>|t|)
(Intercept) 0.994595   0.102952   9.6607 < 2.2e-16
x21         5.287411   0.148321  35.6485 < 2.2e-16
x1          9.892441   0.072182 137.0481 < 2.2e-16

$Comp.3
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 2.869670   0.087103 32.9455  < 2e-16
x21         5.105260   0.129976 39.2784  < 2e-16
x1          0.134829   0.068085  1.9803  0.04906

> Model2 <- FLXglmFix(fixed = ~x2, nested = list(k = c(2,
+     1), formula = c(~x1, ~0)), varFix = TRUE)
> fittedModel2 <- flexmix(y ~ 1, model = Model2, data = NregFix,
+     cluster = posterior(fittedModel), concomitant = FLXmultinom(~w))
> BIC(fittedModel)

[1] 883.5923

> BIC(fittedModel2)

[1] 856.9138
```

Figure B.2: Code for estimating a finite mixture of Gaussian regression models to the `NregFix` data set.

```
> refit(fittedModel2)

Call:
refit(fittedModel2)

Number of components: 3

$Comp.1
             Estimate Std. Error t value  Pr(>|t|)
x21          5.111686   0.080495  63.504 < 2.2e-16
x1           9.902215   0.052161 189.838 < 2.2e-16
(Intercept) -7.848595   0.107757 -72.836 < 2.2e-16

$Comp.2
            Estimate Std. Error t value  Pr(>|t|)
x21         5.111686   0.080495  63.504 < 2.2e-16
x1          9.902215   0.052161 189.838 < 2.2e-16
(Intercept) 1.072239   0.076863  13.950 < 2.2e-16

$Comp.3
            Estimate Std. Error t value  Pr(>|t|)
x21         5.111686   0.080495  63.504 < 2.2e-16
(Intercept) 2.857667   0.068253  41.869 < 2.2e-16
```

Figure B.3: Details of the fitted model to the artificial example of finite mixture of Gaussian regression models.

cients are ordered such that the fixed coefficients are first, the nested varying coefficients second and the varying coefficients last.

B.2 Binomial logit regressions

Beta blockers

The data set is analyzed in Aitkin (1999a,b) using a finite mixture of binomial regression models. Furthermore, it is described in McLachlan and Peel (2000) on page 165. The data set is from a 22-center clinical trial of beta-blockers for reducing mortality after myocardial infarction. A two-level model is assumed to represent the data, where centers are at the upper level and patients at the lower level. The data is illustrated in Figure B.5 and the model is given by:

$$H(\text{Deaths} \mid \text{Total}, \text{Treatment}, \text{Center}, \boldsymbol{\Theta}) = \sum_{s=1}^{S} \pi_s \text{Bi}(\text{Deaths} \mid \text{Total}, \theta_s).$$

First, the center classification is ignored and a binomial logit regression model with treatment as covariate is fitted using `glm`, i.e. $S = 1$. In the next step the centre classification is included by allowing a random effect for the intercept given the centers, i.e. the coefficients $\boldsymbol{\beta}^s$ are given by $(\beta^s_{\text{Intercept}|\text{Center}}, \beta_{\text{Treatment}})$. This signifies that the component membership is fixed for each center. In order to determine the suitable number of components, the mixture is fitted with different numbers of components and the BIC information criterion is used to select an appropriate model. In this case a model with three components is selected. The code is given in Figure B.4 and the fitted values for the model with three components are illustrated in Figure B.5. The centers are sorted by the relative number of deaths in the control group. The lines indicate the fitted values for each component of the 3-component mixture model with random intercept and fixed effect for treatment.

In addition the treatment effect can be also included in the random part of the model. As then all coefficients for the covariates and the intercept follow a mixture distribution the M-step can be specified using `FLXglm`. The coefficients are $\boldsymbol{\beta}^s = (\beta^s_{\text{Intercept}|\text{Center}}, \beta^s_{\text{Treatment}|\text{Center}})$, i.e. it is assumed that the heterogeneity is only between centers and therefore the aggregated data for each center can be used. The code is given in Figure B.4. The full model with a random effect for treatment has a higher BIC and therefore the smaller model would be preferred.

The default plot of the returned `flexmix` object is a rootogramm of the a-posteriori probabilities where observations with a-posteriori probabilities smaller than `eps` are omitted. Argument `mark` specifies the component such that the observations which are assigned to this component based on the maximum a-posteriori probabilities are marked. This indicates which components overlap and can be used as an indicator of closeness between the

```
> data(betablocker)
> betaGlm <- glm(cbind(Deaths, Total - Deaths) ~ Treatment,
+     family = "binomial", data = betablocker)
> betaMixFix <- stepFlexmix(cbind(Deaths, Total - Deaths) ~
+     1 | Center, model = FLXglmFix(family = "binomial",
+     fixed = ~Treatment), K = 2:4, nrep = 5, data = betablocker)

2 : * * * * *
3 : * * * * *
4 : * * * * *

> sapply(betaMixFix, BIC)

       2        3        4
377.7985 341.4262 341.7815

> betaMix <- stepFlexmix(cbind(Deaths, Total - Deaths) ~
+     Treatment | Center, model = FLXglm(family = "binomial"),
+     k = 3, nrep = 5, data = betablocker)

 * * * * *

> summary(betaMix)

Call:
stepFlexmix(cbind(Deaths, Total - Deaths) ~ Treatment |
    Center, model = FLXglm(family = "binomial"),
    k = 3, data = betablocker, nrep = 5)

       prior size post>0 ratio
Comp.1 0.249   10     22 0.455
Comp.2 0.240   10     20 0.500
Comp.3 0.511   24     32 0.750

'log Lik.' -158.3095 (df=8)
AIC: 332.6190   BIC: 346.8925
```

Figure B.4: Code for fitting a mixture of binomial logit models to the `betablocker` data set.

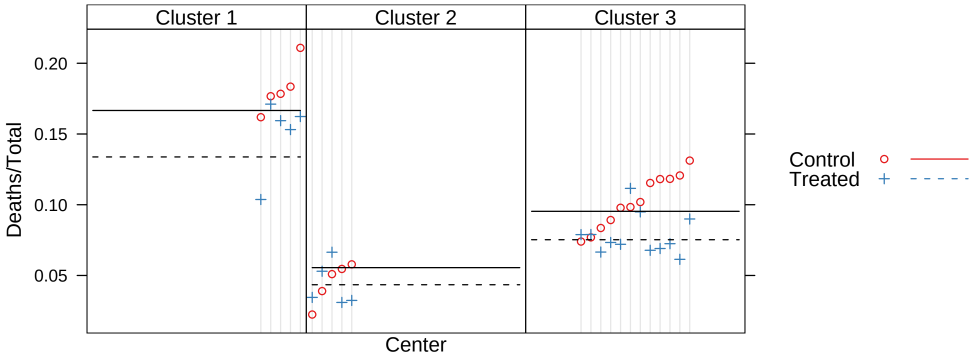


Figure B.5: Relative number of deaths for the treatment and the control group for each center in the betablocker data set.

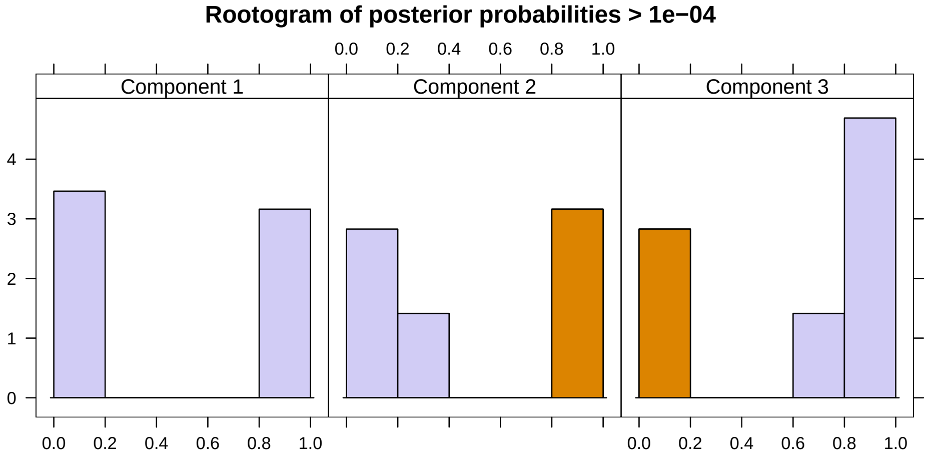


Figure B.6: Default plot for an object of class "flexmix".

components.

In Figure B.6 the fitted model with three components is plotted with the second component marked. The default plot of the fitted model indicates that the components are well separated. In addition component 2 has a slight overlap with component 3, but none with component 1.

Code for analyzing the fitted model is given in Figure B.7. The fitted parameters can be accessed with `parameters()` and the cluster assignments given the maximum a-posteriori probabilities with `cluster()`. The estimated probabilities for each component for the treated patients and those

```
> parameters(betaMix, component = 2)

$coef
     (Intercept) TreatmentTreated
     -2.91634472      -0.08047735

> table(cluster(betaMix))

 1  2  3
10 10 24

> predict(betaMix, newdata = data.frame(Treatment = c("Control",
+     "Treated")))

$Comp.1
       [,1]
1 0.1707940
2 0.1295594

$Comp.2
        [,1]
1 0.05135147
2 0.04756965

$Comp.3
        [,1]
1 0.09554796
2 0.07511130

> fitted(betaMix)[c(1, 23), ]

        Comp.1     Comp.2     Comp.3
[1,] 0.1707940 0.05135147 0.09554796
[2,] 0.1295594 0.04756965 0.07511130

> refit(betaMixFix[["3"]])

Call:
refit(betaMixFix[["3"]])

Number of components: 3

$Comp.1
                  Estimate Std. Error  z value Pr(>|z|)
TreatmentTreated -0.258179   0.049743  -5.1902  2.1e-07
(Intercept)      -1.609734   0.051355 -31.3451  < 2e-16

$Comp.2
                  Estimate Std. Error  z value Pr(>|z|)
TreatmentTreated -0.258179   0.049743  -5.1902  2.1e-07
(Intercept)      -2.833688   0.073686 -38.4563  < 2e-16

$Comp.3
                  Estimate Std. Error  z value Pr(>|z|)
TreatmentTreated -0.258179   0.049743  -5.1902  2.1e-07
(Intercept)      -2.250169   0.039938 -56.3412  < 2e-16
```

Figure B.7: Code for analyzing the model fitted to the `betablocker` data set.

in the control group can be obtained with `predict()` or `fitted()`. A further analysis of the model is possible with function `refit()` which returns the estimated coefficients together with the standard deviations, z-values and corresponding p-values. The printed coefficients are ordered to have the fixed effects before the varying effects.

Mehta et al. trial

This data set is similar to the beta blocker data set and is also analyzed in Aitkin (1999b). The data set is visualized in Figure B.9. The observation for the control group in center 15 is slightly conspicuous and might be classified to be an outlier.

The model is given by:

$$H(\text{Response} \,|\, \text{Total}, \boldsymbol{\Theta}) = \sum_{s=1}^{S} \pi_s \text{Bi}(\text{Response} \,|\, \text{Total}, \theta_s),$$

with $\boldsymbol{\beta}^s = (\beta^s_{\text{Intercept}|\text{Site}}, \beta_{\text{Drug}})$. This model is fitted with the code in Figure B.8.

One component only contains the observations for center 15 and in order to be able to fit a mixture with such a small component it is necessary to modify the default argument for `minprior` which is 0.05. The fitted values for this model are given separately for each component in Figure B.9. The sites are sorted by the relative number of responses in the control group.

The code for also estimating a random effect for the coefficient of Drug, i.e. $\boldsymbol{\beta}^s = (\beta^s_{\text{Intercept}|\text{Site}}, \beta^s_{\text{Drug}|\text{Site}})$, is also given in Figure B.8. The BIC is smaller for the larger model and this indicates that the assumption of an equal drug effect for all centers is not confirmed by the data.

Given Figure B.9 a two-component model with fixed treatment is also fitted to the data where site 15 is omitted. The code is given in Figure B.10.

Tribolium

A finite mixture of binomial regressions is fitted to the Tribolium data set given in Wang and Puterman (1998). The data was collected to investigate

```
> data(Mehta)
> Model <- FLXglmFix(family = "binomial", fixed = ~Drug)
> mehtaMix <- stepFlexmix(cbind(Response, Total - Response) ~
+     1 | Site, model = Model, k = 3, data = Mehta,
+     control = list(minprior = 0.04), nrep = 5)

 * * * * *

> mehtaMix.2 <- stepFlexmix(cbind(Response, Total -
+     Response) ~ Drug | Site, model = FLXglm(family = "binomial"),
+     k = 3, control = list(minprior = 0.04), data = Mehta,
+     nrep = 5)

 * * * * *

> BIC(mehtaMix)

[1] 156.3163

> BIC(mehtaMix.2)

[1] 154.3281
```

Figure B.8: Code for estimating a finite mixture of binomial logit models to the `Mehta` data set.

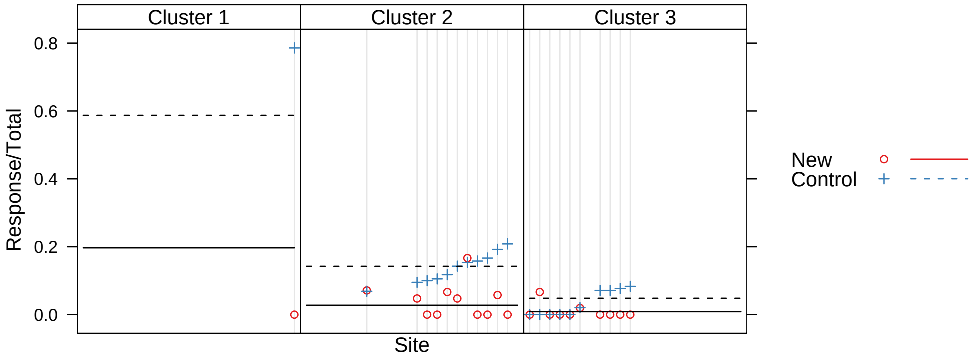


Figure B.9: Relative number of responses for the treatment and the control group for each site in the Mehta et al. trial data set together with the fitted values.

```
> Mehta.sub <- subset(Mehta, Site != 15)
> mehtaMix <- stepFlexmix(cbind(Response, Total - Response) ~
+     1 | Site, model = FLXglmFix(family = "binomial",
+     fixed = ~Drug), data = Mehta.sub, k = 2, nrep = 5)

 * * * * *
```

Figure B.10: Code for estimating a finite mixture to only a subset of the `Mehta` datset.

```
> data(tribolium)
> TribMix <- stepFlexmix(cbind(Remaining, Total - Remaining) ~
+     1, K = 2:3, model = FLXglmFix(fixed = ~Species,
+     family = "binomial"), concomitant = FLXmultinom(~Replicate),
+     data = tribolium)

2 : * * *
3 : * * *
```

Figure B.11: Code for fitting a finite mixture to the `tribolium` data set.

whether the adult Tribolium species Castaneum has developed an evolutionary advantage to recognize and avoid eggs of their own species while foraging, as beetles of the genus Tribolium are cannibalistic in the sense that adults eat the eggs of their own species as well as those of closely related species.

The experiment isolated a number of adult beetles of the same species and presented them with a vial of 150 eggs (50 of each type), the eggs being thoroughly mixed to ensure uniformity throughout the vial. The data gives the consumption data for adult Castaneum species. It reports the number of Castaneum, Confusum and Madens eggs, respectively, that remain uneaten after two day exposure to the adult beetles. Replicates 1, 2, and 3 correspond to different occasions on which the experiment was conducted. The data is visualized in Figure B.13 and the model is given by:

$$H(\text{Remaining}\,|\,\text{Total}, \boldsymbol{\Theta}) = \sum_{s=1}^{S} \pi_s(\text{Replicate}, \boldsymbol{\alpha})\text{Bi}(\text{Remaining}\,|\,\text{Total}, \theta_s),$$

with $\boldsymbol{\beta}^s = (\beta^s_{\text{Intercept}}, \boldsymbol{\beta}_{\text{Species}})$. The code for fitting this model is given in Figure B.11.

The model which is selected as the best in Wang and Puterman (1998) can be estimated with the code given in Figure B.12. Wang and Puterman (1998) also considered a model where they omit one conspicous observation. The code for estimating this model is also given in Figure B.12.

Trypanosome

The data is used in Follmann and Lambert (1989). It is from a dosage-response analysis where the proportion of organisms belonging to different populations shall be assessed. It is assumed that organisms belonging to different populations are indistinguishable other than in terms of their reaction to the stimulus. The experimental technique involved inspection under the microscope of a representative aliquot of a suspension, all organisms appearing within two fields of view being classified either alive or dead. Hence the total numbers of organisms present at each dose and the number showing the quantal response were both random variables. The data is illustrated in Figure B.15.

The model which is proposed in Follmann and Lambert (1989) is given by:

$$H(\text{Dead} \,|\, \boldsymbol{\Theta}) = \sum_{s=1}^{S} \pi_s(\text{Dead}, \boldsymbol{\alpha}) \text{Bi}(\text{Dead} \,|\, \theta_s),$$

where Dead $\in \{0, 1\}$ and with $\boldsymbol{\beta}^s = (\beta^s_{\text{Intercept}}, \boldsymbol{\beta}_{\log(\text{Dose})})$. This model is fitted with the code given in Figure B.14.

The fitted values are visualized in Figure B.15 together with the fitted values of a generalized linear model in order to facilitate comparison of the two models.

```
> modelWang <- FLXglmFix(fixed = ~I(Species == "Confusum"),
+     family = "binomial")
> concomitantWang <- FLXmultinom(~I(Replicate == 3))
> TribMixWang <- stepFlexmix(cbind(Remaining, Total -
+     Remaining) ~ 1, data = tribolium, model = modelWang,
+     concomitant = concomitantWang, k = 2)

 * * *

> refit(TribMixWang)

Call:
refit(TribMixWang)

Number of components: 2

$Comp.1
                              Estimate Std. Error z value
I(Species == "Confusum")TRUE -0.559904   0.124641 -4.4921
(Intercept)                  -0.645144   0.095503 -6.7552
                              Pr(>|z|)
I(Species == "Confusum")TRUE 7.051e-06
(Intercept)                  1.426e-11

$Comp.2
                              Estimate Std. Error z value
I(Species == "Confusum")TRUE -0.559904   0.124641 -4.4921
(Intercept)                   0.194718   0.083817  2.3231
                              Pr(>|z|)
I(Species == "Confusum")TRUE 7.051e-06
(Intercept)                    0.02017

> TribMixWangSub <- stepFlexmix(cbind(Remaining, Total -
+     Remaining) ~ 1, k = 2, data = tribolium[-7, ],
+     model = modelWang, concomitant = concomitantWang)

 * * *
```

Figure B.12: Code for fitting the finite mixture given in Wang and Puterman 1998 to the tribolium data set.

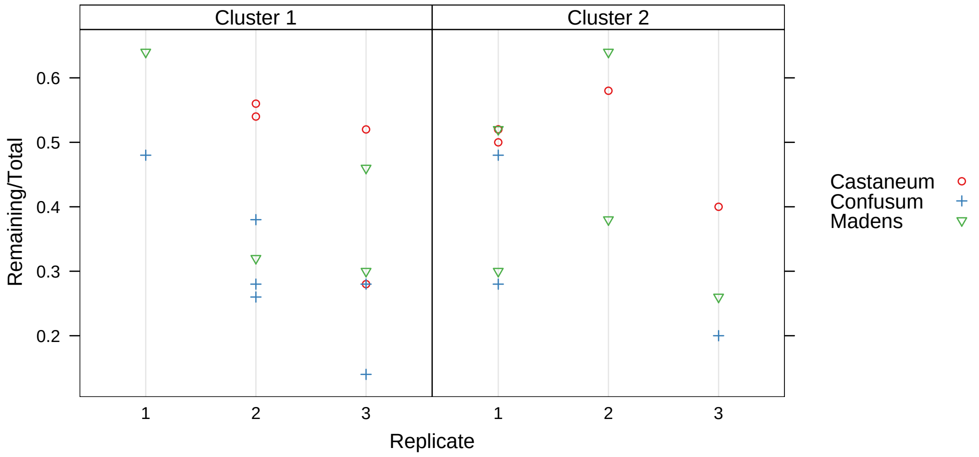


Figure B.13: Relative number of remaining beetles for the number of replicate. The different panels are according to the cluster assignemnts based on the a-posteriori probabilities of the model suggested in Wang and Puterman (1998).

```
> data(trypanosome)
> Model <- FLXglmFix(family = "binomial", fixed = ~log(Dose))
> TrypMix <- stepFlexmix(cbind(Dead, 1 - Dead) ~ 1,
+     k = 2, data = trypanosome, model = Model, nrep = 5)

 * * * * *

> refit(TrypMix)

Call:
refit(TrypMix)

Number of components: 2

$Comp.1
            Estimate Std. Error z value  Pr(>|z|)
log(Dose)    124.856     15.417  8.0983 5.573e-16
(Intercept) -196.269     24.251 -8.0932 5.814e-16

$Comp.2
            Estimate Std. Error z value  Pr(>|z|)
log(Dose)    124.856     15.417  8.0983 5.573e-16
(Intercept) -205.804     25.414 -8.0979 5.590e-16
```

Figure B.14: Code for fitting a finite mixture of the `trypanosome` data set.

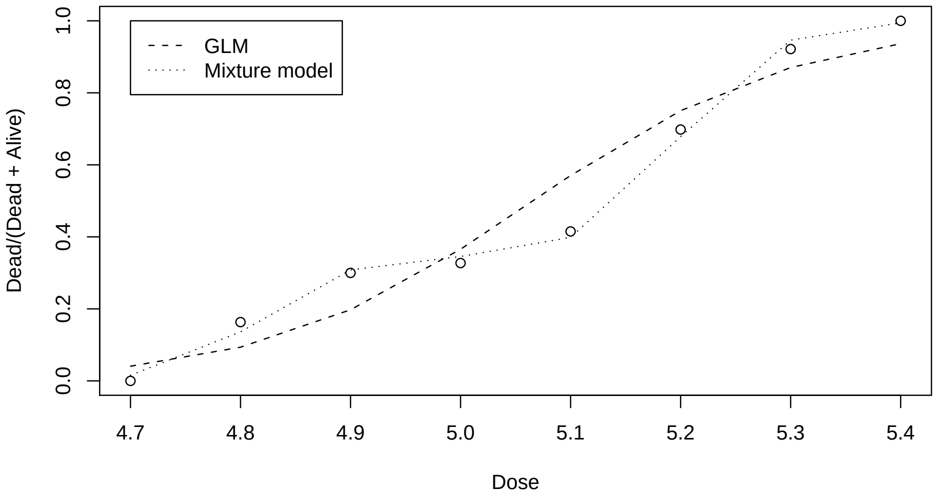


Figure B.15: Relative number of deaths for each dose level together with the fitted values for the generalized linear model ("GLM") and the random intercept model ("Mixture model")

B.3 Poisson regressions

Fabric faults

The data set is analyzed using a finite mixture of Poisson regression models in Aitkin (1996). Furthermore, it is described in McLachlan and Peel (2000) on page 155. A random intercept model is used where a fixed effect is assumed for the logarithm of length. The code is given in Figure B.16.

The intercept of the first component is not significantly different from zero for a signficance level of 0.01. We therefore also fit a modified model where the intercept is a-priori set to zero for the first components. This nested structure is given as part of the model specification with argument `nested`. In this case the argument `k` in `flexmix()` can be omitted. The code is given in Figure B.17. The fitted values for both models together with data are visualized in Figure B.18.

```
> data(fabricfault)
> Model <- FLXglmFix(family = "poisson", fixed = ~log(Length))
> fabricMix <- stepFlexmix(Faults ~ 1, model = Model,
+     data = fabricfault, k = 2, nrep = 5)

 * * * * *

> refit(fabricMix)

Call:
refit(fabricMix)

Number of components: 2

$Comp.1
            Estimate Std. Error z value  Pr(>|z|)
log(Length)  0.80072    0.16776  4.7730 1.815e-06
(Intercept) -2.37389    1.10083 -2.1565   0.03105

$Comp.2
            Estimate Std. Error z value  Pr(>|z|)
log(Length)  0.80072    0.16776  4.7730 1.815e-06
(Intercept) -3.13888    1.07437 -2.9216  0.003482
```

Figure B.16: Code for estimating a finite mixture of Poisson regression models to the `fabricfault` data set.

```
> fabricMix2 <- stepFlexmix(Faults ~ 0, data = fabricfault,
+     nrep = 5, model = FLXglmFix(family = "poisson",
+         fixed = ~log(Length), nested = list(k = c(1,
+             1), formula = list(~1, ~0))))

 * * * * *

> refit(fabricMix2)

Call:
refit(fabricMix2)

Number of components: 2

$Comp.1
             Estimate Std. Error z value  Pr(>|z|)
log(Length)  0.448961   0.014894 30.1441 < 2.2e-16
(Intercept) -0.896638   0.119806 -7.4841 7.205e-14

$Comp.2
            Estimate Std. Error z value  Pr(>|z|)
log(Length) 0.448961   0.014894  30.144 < 2.2e-16
```

Figure B.17: Code for estimating a finite mixture of Poisson regression models with nested varying effects to the `fabricfault` data set.

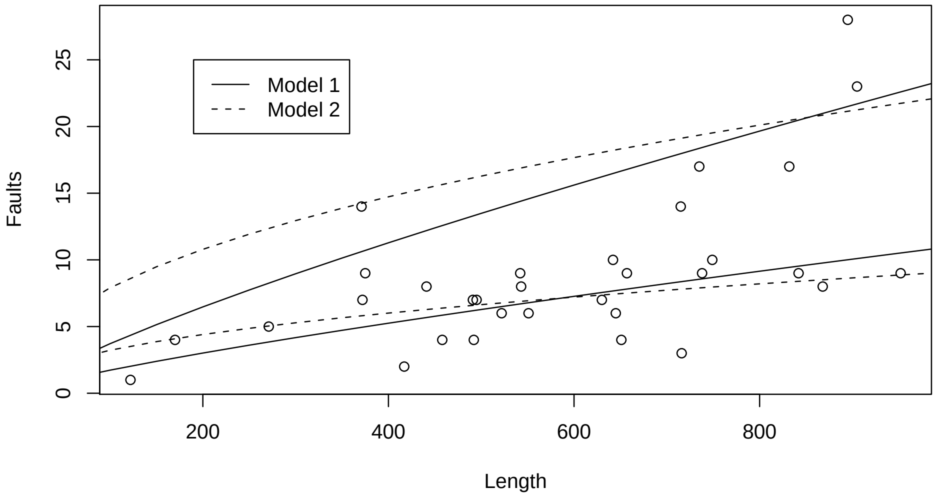


Figure B.18: Observed values of the fabric faults data set together with the fitted values for the components of each of the two fitted models

Seizure

The data is used in Wang et al. (1996) and in Section 4.1.2. It is also included as an example in this section in order to illustrate how the model can be estimated in R using package **flexmix**. The data is from a clinical trial where the effect of intravenous gamma-globulin on suppression of epileptic seizures is studied. There are daily observations for a period of 140 days on one patient, where the first 27 days are a baseline period without treatment, the remaining 113 days are the treatment period. The model proposed in Wang et al. (1996) is given by:

$$H(\text{Seizures} \,|\, (\text{Treatment}, \log(\text{Day}), \log(\text{Hours})), \boldsymbol{\Theta}) = \sum_{s=1}^{S} \pi_s \text{Poi}(\text{Seizures} \,|\, \lambda_s),$$

where $\boldsymbol{\beta}^s = (\beta^s_{\text{Intercept}}, \beta^s_{\text{Treatment}}, \beta^s_{\log(\text{Day})}, \beta^s_{\text{Treatment:}\log(\text{Day})})$ and log(Hours) is used as offset. This model is fitted with the code given in Figure B.19.

A different model with different contrasts to directly estimate the coefficients for the jump in the change between base and treatment period is

```
> data(seizure)
> Model <- FLXglm(family = "poisson", offset = log(seizure$Hours))
> seizMix <- stepFlexmix(Seizures ~ Treatment * log(Day),
+     data = seizure, k = 2, nrep = 5, model = Model)

 * * * * *

> BIC(seizMix)

[1] 796.8272

> refit(seizMix)

Call:
refit(seizMix)

Number of components: 2

$Comp.1
                       Estimate Std. Error z value  Pr(>|z|)
(Intercept)            2.845121   0.234020 12.1576 < 2.2e-16
TreatmentYes           1.301616   0.473909  2.7465  0.006023
log(Day)              -0.406364   0.088249 -4.6048 4.130e-06
TreatmentYes:log(Day) -0.430834   0.133414 -3.2293  0.001241

$Comp.2
                       Estimate Std. Error  z value  Pr(>|z|)
(Intercept)            2.070458   0.089179  23.2169 < 2.2e-16
TreatmentYes           7.431700   0.518023  14.3463 < 2.2e-16
log(Day)              -0.270713   0.038208  -7.0852 1.388e-12
TreatmentYes:log(Day) -2.276095   0.139691 -16.2938 < 2.2e-16
```

Figure B.19: Code for estimating a finite mixture of Poisson regression models to the `seizure` data set.

```
> seizMix2 <- stepFlexmix(Seizures ~ Treatment * log(Day/27),
+     data = seizure, k = 2, nrep = 5, model = Model)

 * * * * *

> BIC(seizMix2)

[1] 796.8272

> refit(seizMix2)

Call:
refit(seizMix2)

Number of components: 2

$Comp.1
                          Estimate Std. Error  z value  Pr(>|z|)
(Intercept)               1.178570   0.058784  20.0491 < 2.2e-16
TreatmentYes             -0.070186   0.103189  -0.6802    0.4964
log(Day/27)              -0.270574   0.038195  -7.0840 1.401e-12
TreatmentYes:log(Day/27) -2.276288   0.139675 -16.2970 < 2.2e-16

$Comp.2
                          Estimate Std. Error z value  Pr(>|z|)
(Intercept)               1.506165   0.076395 19.7155 < 2.2e-16
TreatmentYes             -0.118539   0.118805 -0.9978  0.318396
log(Day/27)              -0.406109   0.088309 -4.5987 4.251e-06
TreatmentYes:log(Day/27) -0.431218   0.133468 -3.2309  0.001234
```

Figure B.20: Code for estimating a finite mixture of Poisson regression models to the `seizure` data set with different contrasts.

fitted in Figure B.20. As the treatment effect is not significant for this model a more parsimonious model is fitted which allows no jump at the change between base and treatment period. The code is given in Figure B.21.

With respect to the BIC criterion the smaller model with no jump is preferred. This is also the more intuitive model from a practitioner's point of view, as it does not seem to be plausible that starting the treatment already gives a significant improvement, but that improvement develops over time. The data points together with the fitted values for each component of the two models are given in Figure B.22. It can clearly be seen that the fitted values are nearly equal which also supports the smaller model.

```
> seizMix3 <- stepFlexmix(Seizures ~ log(Day/27)/Treatment,
+     data = seizure, k = 2, nrep = 5, model = Model)

 * * * * *

> BIC(seizMix3)

[1] 787.8906

> refit(seizMix3)

Call:
refit(seizMix3)

Number of components: 2

$Comp.1
                          Estimate Std. Error z value  Pr(>|z|)
(Intercept)               1.458916   0.057775 25.2518 < 2.2e-16
log(Day/27)              -0.447677   0.074909 -5.9763 2.282e-09
log(Day/27):TreatmentYes -0.458614   0.130660 -3.5100 0.0004481

$Comp.2
                          Estimate Std. Error z value  Pr(>|z|)
(Intercept)               1.149979   0.048972  23.483 < 2.2e-16
log(Day/27)              -0.283968   0.034399  -8.255 < 2.2e-16
log(Day/27):TreatmentYes -2.311302   0.123914 -18.652 < 2.2e-16
```

Figure B.21: Code for estimating a finite mixture of Poisson regression models to the `seizure` data set with no jump between basement and treatment period.

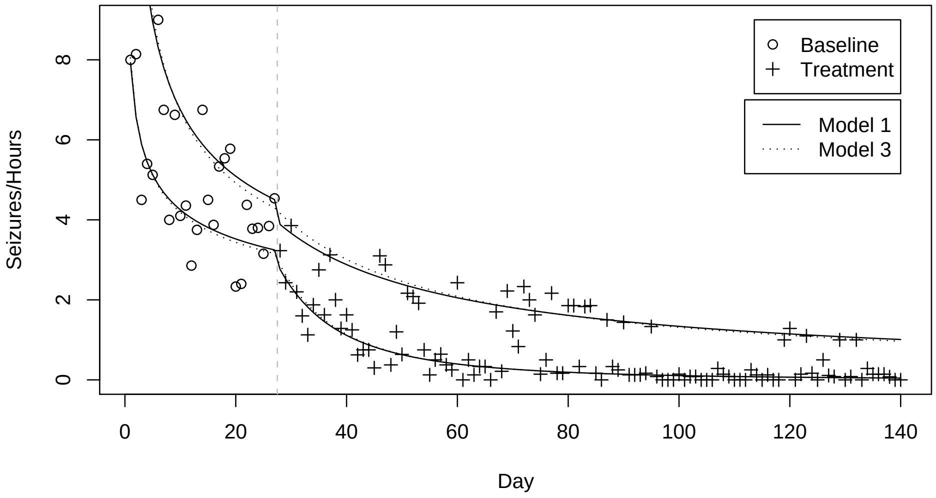


Figure B.22: Observed values for the seizure data set together with the fitted values for the components of the two different models.

Ames salmonella assay data

The ames salmonella assay data set was used in Wang et al. (1996). They propose a model given by:

$$H(\mathrm{y} \mid \mathrm{x}, \boldsymbol{\Theta}) = \sum_{s=1}^{S} \pi_s \mathrm{Poi}(\mathrm{y} \mid \lambda_s),$$

where $\boldsymbol{\beta}^s = (\beta^s_{\mathrm{Intercept}}, \beta_{\mathrm{x}}, \beta_{\log(\mathrm{x}+10)})$. The code for fitting this model is given in Figure B.23. The data together with the fitted lines for each component are given in Figure B.24.

```
> data(salmonellaTA98)
> salmonMix <- stepFlexmix(y ~ 1, data = salmonellaTA98,
+     model = FLXglmFix(family = "poisson", fixed = ~x +
+         log(x + 10)), k = 2, nrep = 5)

 * * * * *
```

Figure B.23: Code for estimating a finite mixture to the `salmonellaTA98` data set.

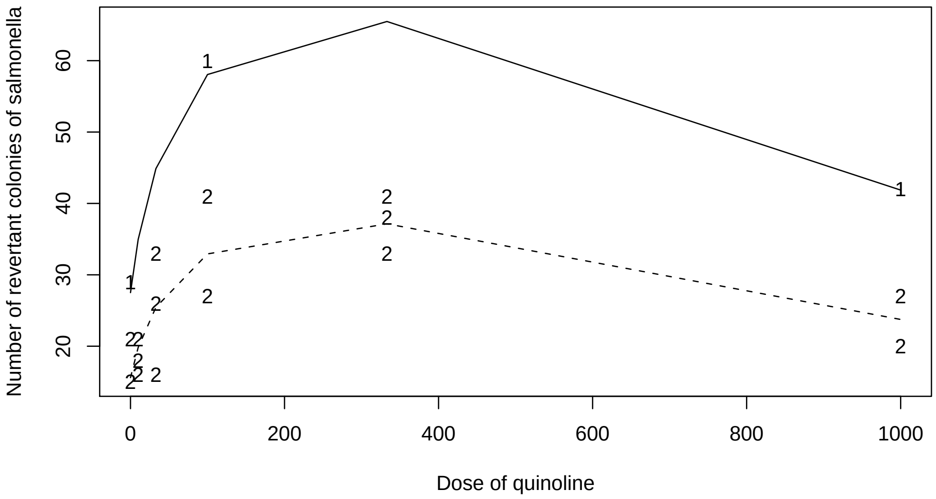


Figure B.24: Means and classification for assay data according to the estimated posterior probabilities based on the fitted model

List of Figures

List of Tables

Bibliography

Agresti, A. (1990). *Categorical Data Analysis*. Wiley, first edition.

Aitkin, M. (1996). A general maximum likelihood analysis of overdispersion in generalized linear models. *Statistics and Computing*, 6:251–262.

Aitkin, M. (1999a). A general maximum likelihood analysis of variance components in generalized linear models. *Biometrics*, 55:117–128.

Aitkin, M. (1999b). Meta-analysis by random effect modelling in generalized linear models. *Statistics in Medicine*, 18(17–18):2343–2351.

Basford, K. E., Greenway, D. R., McLachlan, G. J., and Peel, D. (1997). Standard errors of fitted means under normal mixture model. *Computational Statistics*, 12:1–17.

Biernacki, C., Celeux, G., and Govaert, G. (2000). Assessing a mixture model for clustering with the integrated completed likelihood. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(7):719–725.

Biernacki, C., Celeux, G., and Govaert, G. (2003). Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixture models. *Computational Statistics & Data Analysis*, 41:561–575.

Blischke, W. R. (1964). Estimating the parameters of mixtures of binomial distributions. *Journal of the American Statistical Association*, 59(306):510–528.

Böhning, D., Dietz, E., Schlattmann, P., Mendonça, L., and Kirchner, U. (1999). The zero-inflated Poisson model and the decayed, missing and filled teeth index in dental epidemiology. *Journal of the Royal Statistical Society A*, 162(2):195–209.

Böhning, D., Schlattmann, P., and Lindsay, B. (1992). Computer-assisted analysis of mixtures (C.A.MAN): Statistical algorithms. *Biometrics*, 48(1):283–303.

Celeux, G. (1998). Bayesian inference for mixture: The label-switching problem. In Payne, R. and Green, P., editors, *Compstat 1998 — Proceedings in Computational Statistics*, pages 227–232. Physica Verlag, Heidelberg.

Celeux, G. and Diebolt, J. (1988). A random imputation principle: The stochastic EM algorithm. Rapports de Recherche 901, INRIA.

Celeux, G. and Govaert, G. (1992). A Classification EM algorithm for clustering and two stochastic versions. *Computational Statistics & Data Analysis*, 14(3):315–332.

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

Celeux, G., Hurn, M., and Robert, C. P. (2000). Computational and inferential difficulties with mixture posterior distributions. *Journal of the American Statistical Association*, 95(451):957–970.

Chambers, J. M. (1998). *Programming with Data*. Springer, New York.

Cheng, M.-Y. and Hall, P. (1998). Calibrating the excess mass and dip tests of modality. *Journal of the Royal Statistical Society B*, 60(3):279–289.

Chung, H., Loken, E., and Schafer, J. L. (2004). Difficulties in drawing inferences with finite-mixture models: A simple example with a simple solution. *The American Statistician*, 58(2):152–158.

Dasgupta, A. and Raftery, A. E. (1998). Detecting features in spatial point processes with clutter via model-based clustering. *Journal of the American Statistical Association*, 93(441):294–302.

Davison, A. C. and Hinkley, D. V. (1997). *Bootstrap methods and their application.* Cambridge series on statistical and probabilistic mathematics. Cambridge University Press, Cambridge, UK.

Dayton, C. M. and Macready, G. B. (1988). Concomitant-variable latent-class models. *Journal of the American Statistical Association*, 83(401):173–178.

de Menezes, L. M. (1999). On fitting latent class models for binary data: The estimation of standard errors. *British Journal of Mathematical and Statistical Psychology*, 52(2):149–168.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM-algorithm. *Journal of the Royal Statistical Society B*, 39:1–38.

Diebolt, J. and Ip, E. H. S. (1996). Stochastic EM: method and application. In Gilks, W. R., Richardson, S., and Spiegelhalter, D. J., editors, *Markov Chain Monte Carlo in Practice*, pages 259–273. Chapman and Hall.

Diebolt, J. and Robert, C. P. (1994). Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society B*, 56:363–375.

Diggle, P. J., Liang, K.-Y., and Zeger, S. L. (1994). *Analysis of Longitudinal Data.* Oxford University Press, New York.

Elmore, R. T. and Wang, S. (2003). Identifiability and estimation in finite mixture models with multinomial components. Technical Report 03-04, Department of Statistics, Pennsylvania State University.

Everitt, B. S. (1988). A finite mixture model for the clustering of mixed-mode data. *Statistics & Probability Letters*, 6:305–309.

Everitt, B. S. and Hand, D. J. (1981). *Finite Mixture Distributions*. London: Chapman and Hall.

Feng, Z. D. and McCulloch, C. E. (1996). Using bootstrap likelihood ratios in finite mixture models. *Journal of the Royal Statistical Society B*, 58(3):609–617.

Follmann, D. A. and Lambert, D. (1989). Generalizing logistic regression by non-parametric mixing. *Journal of the American Statistical Association*, 84(405):295–300.

Follmann, D. A. and Lambert, D. (1991). Identifiability of finite mixtures of logistic regression models. *Journal of Statistical Planning and Inference*, 27:375–381.

Fraley, C. and Raftery, A. E. (2002). Model-based clustering, discriminant analysis and density estimation. *Journal of the American Statistical Association*, 97(458):611–631.

Fraser, M. D., Hsu, Y.-S., and Walker, J. J. (1981). Identifiability of finite mixtures of von Mises distributions. *The Annals of Statistics*, 9(5):1130–1131.

Frühwirth-Schnatter, S. (2001). Markov chain Monte Carlo estimation of classical and dynamic switching and mixture models. *Journal of the American Statistical Association*, 96(453):194–209.

Frühwirth-Schnatter, S. (2006). *Finite Mixture and Markov Switching Models*. Springer Series in Statistics. Springer, New York.

Gelman, A. (2004). Exploratory data analysis for complex models. *Journal of Computational and Graphical Statistics*, 13(4):755–779.

Gentleman, R. and Ihaka, R. (2000). Lexical scope and statistical computing. *Journal of Computational and Graphical Statistics*, 9(3):491–508.

Goodman, L. A. (1974). Exploratory latent structure analysis using both identifiable and unidentifiable models. *Biometrika*, 61(2):215–231.

Grün, B. (2002). Identifizierbarkeit von multinomialen Mischmodellen. Master's thesis, Technische Universität Wien. Kurt Hornik and Friedrich Leisch, advisors.

Grün, B. and Leisch, F. (2004). Bootstrapping finite mixture models. In Antoch, J., editor, *Compstat 2004 — Proceedings in Computational Statistics*, pages 1115–1122. Physica Verlag, Heidelberg.

Grün, B. and Leisch, F. (2006a). Fitting finite mixtures of generalized linear regressions in R. *Computational Statistics & Data Analysis*. Accepted for publication.

Grün, B. and Leisch, F. (2006b). Fitting finite mixtures of linear regression models with varying & fixed effects in R. In Rizzi, A. and Vichi, M., editors, *Compstat 2006—Proceedings in Computational Statistics*, pages 853–860. Physica Verlag, Heidelberg, Germany.

Hartigan, J. A. and Hartigan, P. M. (1985). The dip test of unimodality. *The Annals of Statistics*, 13(1):70–84.

Hastie, T. and Tibshirani, R. (1993). Varying-coefficient models. *Journal of the Royal Statistical Society B*, 55(4):757–796.

Hennig, C. (2000). Identifiability of models for clusterwise linear regression. *Journal of Classification*, 17(2):273–296.

Hintze, J. L. and Nelson, R. D. (1998). Violin plots: A box plot-density trace synergism. *The American Statistician*, 52(2):181–184.

Holm, S. (1979). A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6:65–70.

Holzmann, H., Munk, A., and Stratmann, B. (2004). Identifiability of finite mixtures — with applications to circular distributions. *Sankhya*, 66:440–450.

Hothorn, T., Leisch, F., and Zeileis, A. (2006). *modeltools: Tools and Classes for Statistical Models.* R package version 0.2-4.

Hothorn, T., Leisch, F., Zeileis, A., and Hornik, K. (2005). The design and analysis of benchmark experiments. *Journal of Computational and Graphical Statistics*, 14(3):1–25.

Hubert, L. and Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2:193–218.

Hunt, L. and Jorgensen, M. (1999). Mixture model clustering using the MULTIMIX program. *Australian & New Zealand Journal of Statistics*, 41(2):153–171.

Hurn, M., Justel, A., and Robert, C. P. (2003). Estimating mixtures of regressions. *Journal of Computational and Graphical Statistics*, 12(1):55–79.

Jasra, A., Holmes, C. C., and Stephens, D. A. (2005). Markov chain Monte Carlo methods and the label switching problem in Bayesian mixture modelling. *Statistical Science*, 20(1):50–67.

Kamakura, W. A. and Russell, G. J. (1989). A probabilistic choice model for market segmentation and elasticity structure. *Journal of Marketing Research*, 26:379–390.

Karlis, D. and Xekalaki, E. (2003). Choosing initial values for the EM algorithm for finite mixtures. *Computational Statistics & Data Analysis*, 41:577–590.

Kaufman, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data.* John Wiley & Sons, Inc., New York, USA.

Kent, J. T. (1983). Identifiability of finite mixtures for directional data. *The Annals of Statistics*, 11(3):984–988.

Kiefer, N. M. (1978). Discrete parameter variation: Efficient estimation of a switching regression model. *Econometrics*, 46(2):427–434.

Kruskal, J. (1977). The relationship between multidimensional scaling and clustering. In Ryzin, J. V., editor, *Classification and Clustering*, pages 17–44. Academic Press, Inc., New York.

Leisch, F. (2004). FlexMix: A general framework for finite mixture models and latent class regression in R. *Journal of Statistical Software*, 11(8).

Leisch, F. (2006). A toolbox for k-centroids cluster analysis. *Computational Statistics & Data Analysis*. Accepted for publication.

Leisch, F. and Grün, B. (2006). Extending standard cluster algorithms to allow for group constraints. In Rizzi, A. and Vichi, M., editors, *Compstat 2006—Proceedings in Computational Statistics*, pages 885–892. Physica Verlag, Heidelberg, Germany.

Lindsay, B. G. (1989). Moment matrices: Applications in mixtures. *The Annals of Statistics*, 17(2):722–740.

Lindsay, B. G. (1995). *Mixture Models: Theory, Geometry, and Applications.* The Institute for Mathematical Statistics, Hayward, California.

Marin, J.-M., Mengersen, K., and Robert, C. P. (2005). Bayesian modelling and inference on mixtures of distributions. In Dey, D. and Rao, C., editors, *Bayesian Thinking, Modeling and Computation*, volume 25 of *Handbook of Statistics*, chapter 16. North–Holland, Amsterdam.

Mazanec, J., Grabler, K., and Maier, G. (1997). *International City Tourism: Analysis and Strategy.* Pinter/Cassel.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models.* Chapman and Hall, second edition.

McFadden, D. (1974). Conditional logit analysis of qualitative choice behavior. In Zarembka, P., editor, *Frontiers in Econometrics*, pages 105–142. Academic Press.

McLachlan, G. J. (1987). On bootstrapping the likelihood ratio test statistic for the number of components in a normal mixture. *Journal of the Royal Statistical Society C*, 36(3):318–324.

McLachlan, G. J. and Basford, K. E. (1988). *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, New York.

McLachlan, G. J. and Krishnan, T. (1997). *The EM Algorithm and Extensions*. John Wiley and Sons Inc., 1st edition.

McLachlan, G. J. and Peel, D. (2000). *Finite Mixture Models*. Wiley.

Minnotte, M. C. (1997). Nonparameteric testing of the existence of modes. *The Annals of Statistics*, 25(4):1646–1660.

Newcomb, S. (1886). A generalized theory of the combination of observations so as to obtain the best result. *American Journal of Mathematics*, 8:343–366.

Papadimitriou, C. and Steiglitz, K. (1982). *Combinatorial Optimization: Algorithms and Complexity*. Prentice Hall, Englewood Cliffs, USA.

Papastefanou, G. (2001). The ZUMA scientific use file of the GfK ConsumerScan household panel 1995. In Papastefanou, G., Schmidt, P., Börsch-Supan, A., Lüdtke, H., and Oltersdorf, U., editors, *Social and Economic Analyses with Consumer Panel Data*, volume 7 of *ZUMA-Nachrichten Spezial*, pages 206–212. Zentrum für Umfragen, Meinungen und Analysen, Mannheim.

Pearson, K. (1894). Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society A*, 185:71–110.

Pinheiro, J. C. and Bates, D. M. (2000). *Mixed-Effects Models in S and S-Plus*. Springer.

R Development Core Team (2006). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria.

Redner, R. A. and Walker, H. F. (1984). Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, 26(2):195–239.

Richardson, S. and Green, P. J. (1997). On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society B*, 59(4):731–92.

Silverman, B. W. (1981). Using kernel density estimates to investigate multimodality. *Journal of the Royal Statistical Society B*, 43(1):97–99.

Soofi, E. S. (1992). A generalizable formulation of conditional logit with diagnostics. *Journal of the American Statistical Association*, 87(419):812–816.

Stephens, M. (1997). *Bayesian Methods for Mixtures of Normal Distributions*. PhD thesis, University of Oxford.

Stephens, M. (2000a). Bayesian analysis of mixtures models with an unknown number of components — An alternative to reversible jump methods. *The Annals of Statistics*, 28(1):40–74.

Stephens, M. (2000b). Dealing with label switching in mixture models. *Journal of the Royal Statistical Society B*, 62(4):795–809.

Tantrum, J., Murua, A., and Stuetzle, W. (2003). Assessment and pruning of hierarchical model based clustering. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 197–205, New York, NY, USA. ACM Press.

Teicher, H. (1960). On the mixture of distributions. *The Annals of Mathematical Statistics*, 31:55–73.

Teicher, H. (1963). Identifiability of finite mixtures. *The Annals of Mathematical Statistics*, 34:1265–1269.

Titterington, D. M., Smith, A. F. M., and Makov, U. E. (1985). *Statistical Analysis of Finite Mixture Distributions*. Wiley.

Turner, T. R. (2000). Estimating the propagation rate of a viral infection of potato plants via mixtures of regressions. *Journal of the Royal Statistical Society C*, 49(3):371–384.

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. Springer, New York, fourth edition.

von Davier, M. (1997). Bootstrapping goodness-of-fit statistics for sparse categorical data—Results of a Monte Carlo study. *Methods of Psychological Research Online*, 2(2):29–48.

Wagstaff, K., Cardie, C., Rogers, S., and Schroedl, S. (2001). Constrained k-means clustering with background knowledge. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 577–584.

Wang, P., Cockburn, I. M., and Puterman, M. L. (1998). Analysis of patent data — A mixed-Poisson-regression-model approach. *Journal of Business & Economic Statistics*, 16(1):27–41.

Wang, P. and Puterman, M. L. (1998). Mixed logistic regression models. *Journal of Agricultural, Biological, and Environmental Statistics*, 3(2):175–200.

Wang, P., Puterman, M. L., Cockburn, I. M., and Le, N. D. (1996). Mixed Poisson regression models with covariate dependent rates. *Biometrics*, 52:381–400.

Wedel, M. and DeSarbo, W. S. (1995). A mixture likelihood approach for generalized linear models. *Journal of Classification*, 12:21–55.

Wegman, E. J. (1990). Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association*, 85:664–675.

Yakowitz, S. J. and Spragins, J. D. (1968). On the identifiability of finite mixtures. *The Annals of Mathematical Statistics*, 39(1):209–214.

Curriculum Vitae

Personal Data

Born:	February 18, 1979 in Mistelbach, Austria
Nationality:	Austria

Education

1989–1997	High School, BG & BRG Gänserndorf, Austria
1997–2002	University studies in Applied Mathematics ("Technische Mathematik"), Vienna University of Technology
2001	ERASMUS exchange programme at the Technical University of Danmark (5 months)
since 2003	Ph.D. Studies in Applied Mathematics, Vienna University of Technology

Career History & Work Experience

10/2000–01/2001	Tutor, Institute for Analysis and Technical Mathematics, Vienna University of Technology
03/2003–09/2003	Research assistent, FFF project "E-direct Marketing Solution"
10/2003–01/2004	Tutor, Department of Statistics and Probability Theory, Vienna University of Technology

10/2003–02/2004	Research assistent, SFB 010 “Adaptive Information Systems and Modelling in Economics and Management Science”
05/2004–08/2004	Research assistent, Project “Statistical Computing with R”, Vienna University of Economics and Business Administration
09/2004-12/2004	Visiting fellow, University of Wollongong, Australia
since 01/2005	DOC-FFORTE scholarship from the Austrian Academy of Sciences (ÖAW)