Essays on estimation strategies addressing label-switching in Gaussian mixtures of semi- and non-parametric regressions

by

Sphiwe Bonakele Skhosana

Submitted in partial fulfilment of the requirements for the degree
Doctor of Philosophy (Mathematical Statistics)
in the Faculty of Economics and Management Sciences
University of Pretoria

Supervisor: Prof. Sollie Millard
Co-supervisor: Prof. Frans Kanfer

March 2024

Essays on estimation strategies addressing label-switching in Gaussian mixtures of semi- and non-parametric regressions

by

Sphiwe Bonakele Skhosana
E-mail: spiwe.skhosana@up.ac.za

Abstract

Gaussian mixtures of semi-parametric regressions (GMNRs) are a flexible class of mixture models. These models assume that some or all of the parameters of the classical Gaussian mixture of regressions (GMRs) model are semi- and non-parametric functions of the covariates. This flexibility gives these models wide applicability for studying the dependence of one variable on one or more covariates when the underlying population is made up of unobserved subpopulations.

The predominant approach used to estimate the GMRs model is maximum likelihood via the Expectation-Maximisation (EM) algorithm. Due to the presence of non-parametric terms in GMNRs, the model estimation poses a computational challenge. A local-likelihood estimation of the non-parametric functions via the EM algorithm may be subject to label-switching.

To estimate the non-parametric functions, we have to define a local-likelihood function for each local grid point on the domain of a given covariate. If we separately maximise each local-likelihood function, using the EM algorithm, the labels attached to the mixture components may experience label switching from one local grid point to the next. The practical consequence of this label-switching is characterised by non-parametric estimates that are non-smooth and discontinuous, exhibiting irregular behaviour at local points where the switch took place.

In this thesis, we propose effective estimation strategies to address label-switching when fitting GMNRs models. The common thread that underlies the proposed strategies is the replacement of the separate maximisations of the local-likelihood functions with simultaneous maximisation. The effectiveness of the proposed methods is demonstrated on finite sample data using Monte Carlo simulations. Furthermore, the practical usefulness of the proposed methods is demonstrated through applications on real data.

Acknowledgements

I would like to extend my heartfelt gratitude to everyone who has contributed towards the success of this research project.
Funding for this research project was provided by

1. the new Generation of Academics Programme (nGAP), Department of Higher Education, South Africa;
2. STATOMET, Department of Statistics, University of Pretoria;
3. the National Graduate Academy for Mathematical and Statistical Sciences (NGA-MASS); and
4. the National Research Foundation (NRF).

Research Outputs

The following is a list of research outputs related to this thesis:

Journal articles

1. **Skhosana, S. B.**, Kanfer, F. H. J. and Millard, S. M. (2022). Fitting Non-Parametric Mixture of Regressions: Introducing an EM-Type Algorithm to Address the Label-Switching Problem. *Journal of Symmetry*, 14 (5), 1058. doi: https://doi.org/10.3390/sym14051058

2. **Skhosana, S. B.**, Millard, S. M. and Kanfer, F. H. J. (2023). A Novel EM-Type Algorithm to Estimate Semi-Parametric Mixtures of Partially Linear Models. *Journal of Mathematics*, 11 (5), 1087. doi: https://doi.org/10.3390/math11051087

3. **Skhosana, S. B.**, Millard, S. M. and Kanfer, F. H. J. (2024). A modified EM-type algorithm to estimate semi-parametric mixtures of non-parametric regressions. *Statistics and Computing*, 34, 125. doi: https://doi.org/10.1007/s11222-024-10435-3

Book chapters

1. **Skhosana, S. B.**, Millard, S. M., Kanfer, F. H. J. (2024). A new approach to estimate semi-parametric Gaussian mixtures of regressions with varying mixing proportions. Submitted as a chapter in the book: Emerging Topics in Mathematical and Statistical Modelling (Editors: Coelho, C. A and Chen, D. G.) in the book series Emerging Topics in Statistics and Biostatistics

Conference presentations

1. *A possible solution to the label-switching problem in fitting nonparametric mixture of regressions*, 14th International Conference on Computational and Methodological Statistics (CMStatistics2021), 18-20 December 2021, hosted by King's College London, UK

2. *A one-step backfitting algorithm for estimating the semi-parametric mixture of partial linear models*, 15th International Conference on Computational and Methodological Statistics (CMStatistics2022), 17-19 December 2022, hosted by King's College London, UK

3. *Estimating semi-parametric Gaussian mixtures of non-parametric regressions*, Southern African Mathematical Sciences Association (SAMSA) Conference, 21-24 November 2023, hosted by the University of Pretoria, South Africa

4. *A new approach to estimate semi-parametric Gaussian mixtures of non-parametric regressions*, 16th International Conference on Computational and Methodological Statistics (CMStatistics2023), 16-18 December 2023, hosted by HTW Berlin, University of Applied Sciences (Wilhelminenhof campus), Berlin, Germany

Declaration by Author

I, *Sphiwe Bonakele Skhosana*, declare that this thesis, which I hereby submit in partial fulfillment of the degree of Doctor of Philosophy in Statistics at the University of Pretoria, is my own work and has not been previously submitted at this or any other tertiary institution.

Sphiwe Bonakele Skhosana

Date

Contents

List of Figures

List of Tables

Nomenclature

Notation

$\mathbf{t} = (t_1, t_2, \ldots, t_D)$ denotes a $D-$dimensional vector of variables.

t_{id} denotes the i^{th} value of the d^{th} covariate from an observed sample $\{t_{id} : i = 1, 2, \ldots, n\}$

$\mathbf{t}_{-d}$ denotes the vector $\mathbf{t}$ excluding the d^{th} variable.

$\mathbf{t}_{i,-d}$ denotes the i^{th} value of $\mathbf{t}_{-d}$ from an observed sample $\{\mathbf{t}_{i,-d} : i = 1, 2, \ldots, n\}$

$g_d(t_d)$ denotes a function of the d^{th} variable. We will use Roman letters to represent such functions.

$g_{k,d}(t_d)$ denotes a function of the d^{th} variable belonging to the k^{th} component.

$\beta_j(t_d)$ denotes a regression coefficient function of the j^{th} variable as a function of the d^{th} variable. We will use Greek letters to represent such functions.

$\beta_{k,j}(t_d)$ denotes the k^{th} component regression function of the j^{th} variable as a function of the d^{th} variable.

$\boldsymbol{\theta}_{\cdot j} = (\theta_{1,j}, \theta_{2,j}, \ldots, \theta_{K,j})$ or $\boldsymbol{\theta}_{j\cdot} = (\theta_{j,1}, \theta_{j,2}, \ldots, \theta_{j,K})$ denotes a $K-$dimensional vector.

$(\theta_{ij})_{1 \leq i \leq n, 1 \leq j \leq N}$ denotes an $(n \times N)-$dimensional vector

K denotes the number of mixture components.

k denotes the k^{th} component.

$\mathrm{tr}(\mathbf{A})$ denotes the trace of a square matrix $\mathbf{A}$

$\mathbb{E}\{Y \,|\, X\}$ denotes a conditional expectation of Y given X

$\mathcal{N}(\cdot|\mu, \sigma^2)$ denotes the Gaussian density function with mean μ and variance σ^2

$\lceil x \rceil$ denotes the greatest integer part of x

Abbreviations

AIC Akaike information criterion

ASE Average squared error

BIC Bayesian information criterion

CO_2 Carbon dioxide

COD Curse of dimensionality

CRF Component regression function

CV Cross-validation

ECM Expectation-Conditional Maximisation

EM Expectation Maximisation

FIB Fully iterative backfitting

GCV Generalised cross-validation

GMLRs Gaussian mixture of linear regressions

GMM Gaussian mixture model

GMNRs Gaussian mixture of semi-parametric regressions

GMPLAMs Gaussian mixtures of partially linear additive models

GMRs Gaussian mixture of regressions

GMVCPLAMs Gaussian mixture of varying coefficients partially linear additive models

IC Information criteria

LCE Local-constant estimator

LEM Local EM

LLE Local-linear estimator

LPL Local polynomial likelihood

LPLS Local polynomial least squares

LPPL Local polynomial profile likelihood

LPR Local polynomial regression

MB-ECM Model-based ECM

MEs Mixtures of Experts

NPGMGMs Non-parametric Gaussian mixture of graphical models

NPGMNRs Non-parametric Gaussian mixture of non-parametric regressions

NPGMSIMs Non-parametric Gaussian mixture of single index models

NPGMVCMs Non-parametric Gaussian mixture of varying coefficient models

OB-EM Objective-based EM

OB-PL-EM Objective-based profile likelihood EM

OSBK One-step backfitting kernel

OSBLPL One-step backfitting LPL

OSLL One-step local-linear

PL-EM Profile Likelihood EM

PLAM Partially linear additive model

PLM Partial linear model

pp piecewise polynomial

RASE Root average squared errors

SBK Spline-backfitting kernel

SBLL Spline-backfitted local-linear

SBLPL Spline-backfitting LPL

SIM Single index model

SPGMAMs Semi-parametric Gaussian mixture of additive models

SPGMHSIM semi-parametric Gaussian mixtures of heteroscedastic single index model

SPGMNRs Semi-parametric Gaussian mixture of non-parametric regressions

SPGMPLAMs Semi-parametric Gaussian mixture of partially linear additive models

SPGMPLMs Semi-parametric Gaussian mixture of partially linear models

SPGMPLSIMs Semi-parametric Gaussian mixtures of partially linear single index models

SPGMRVPs Semi-parametric Gaussian mixture of regressions with varying proportions

SPGMRVSIPs Semi-parametric Gaussian mixture of varying single-index proportions

SPGMSIMs Semi-parametric Gaussian mixture of single-index models

SPGMSIMVSIPs Semi-parametric Gaussian mixture of single index models with varying single index proportions

Chapter 1

Introduction

Gaussian mixtures of regressions (GMRs) are a useful tool for regression analysis whenever the underlying population is made up of unobserved heterogeneous sub-populations whose size and composition is *unknown*. Even if the underlying population is homogeneous, GMRs have an advantage over simple regression models. In the presence of outliers, GMRs can serve as a robust alternative to simple regression models (see Box and Tiao [1968] and see subsection 8.2.4 of Frühwirth-Schnatter [2006]). Moreover, they can be used as an informal test of whether or not the underlying population is indeed homogeneous. A review of GMRs and mixtures of regressions, in general, is given in chapter 8 of Frühwirth-Schnatter [2006].

Due to their many advantages, GMRs have received widespread adoption in areas such as economics as switching regression models (Quandt [1972]); machine learning as mixtures-of-experts (Jacobs et al. [1991]); marketing (DeSarbo and Cron [1988]) as latent class models, among many other areas of research.

A flexible alternative to GMRs is the Gaussian mixtures of semi-parametric regressions (GMNRs). The GMNRs model assumes that the mixing proportions, variances and/or regression functions are semi- and/or non-parametric functions of the covariates. This flexibility allows the data to discover appropriate functional forms of, say the component regression functions, or even verify putative parametric forms of these functions.

Due to their flexibility, there has recently been growing interest in the study and practical use of GMNRs, see Xiang and Yao [2016], Wu and Liu [2017], Huang et al. [2018], Xiang and Yao [2020] and more recently Zhang and Pan [2022] and Xue et al. [2024]. However, there is no general formulation of this class of models. Moreover, their computational challenges have not yet been well covered in the literature and thus are not well understood. For these reasons, the main objectives of this thesis are to: (1) give a general formulation of the GMNRs model and an overview of its main special cases focusing on their practical utility and (2) discuss the difficulties in estimating these models, develop appropriate estimation procedures for these models

and demonstrate the effectiveness and practical usefulness of these estimation procedures.

1.1 A general Gaussian mixture of semi-parametric regressions (GMNRs) model

Let Y be the response variable whose behaviour can be explained by a set of covariates $(S, \mathbf{X}, \mathbf{Z}, \mathbf{T})^\intercal$, where $\mathbf{X} = (X_1, \ldots, X_{D_1})^\intercal$, $\mathbf{Z} = (Z_1, \ldots, Z_{D_2})^\intercal$ and $\mathbf{T} = (T_1, \ldots, T_{D_3})^\intercal$. Given $S = s$, $\mathbf{X} = \mathbf{x}$, $\mathbf{Z} = \mathbf{z}$ and $\mathbf{T} = \mathbf{t}$, the conditional distribution of Y is given by the GMRs model

$$f(Y|S = s, \mathbf{X} = \mathbf{x}, \mathbf{Z} = \mathbf{z}, \mathbf{T} = \mathbf{t}) = \sum_{k=1}^{K} \pi_k(s)\mathcal{N}(Y|m_k(s, \mathbf{x}, \mathbf{z}, \mathbf{t}), \sigma_k^2(s)), \tag{1.1}$$

where the mixing proportions $\pi_k(s) > 0$, for $k = 1, 2, \ldots, K$, satisfying $\sum_{k=1}^{K} \pi_k(s) = 1$, and the variances $\sigma_k^2(s)$, for $k = 1, 2, \ldots, K$, are assumed to be smooth unknown functions of the covariate s. The function $\mathcal{N}(\cdot|\mu, \sigma^2)$ is a Gaussian density function with mean μ and variance σ^2. For simplicity, we assume that s is univariate. The component regression functions (CRFs), $m_k(s, \mathbf{x}, \mathbf{z}, \mathbf{t})$, are given by

$$\begin{aligned} m_k(s, \mathbf{x}, \mathbf{z}, \mathbf{t}) &= \sum_{a=1}^{D_1} \beta_{k,a} x_a + \sum_{b=0}^{D_2} \gamma_{k,b}(s) z_b + \sum_{c=1}^{D_3} g_{k,c}(t_c), \\ &= \mathbf{x}^\intercal \boldsymbol{\beta}_k + \mathbf{z}^\intercal \boldsymbol{\gamma}_k(s) + \sum_{c=1}^{D_3} g_{k,c}(t_c), \qquad \text{for } k = 1, 2, \ldots, K, \end{aligned} \tag{1.2}$$

where $\mathbf{z} = (z_0, z_1, \ldots, z_{D_2})^\intercal$ with $z_0 = 1$ for the intercept term, $\boldsymbol{\beta}_k = (\beta_{k,1}, \ldots, \beta_{k,D_1})^\intercal$ is a vector of regression coefficients and $\boldsymbol{\gamma}_k(s) = (\gamma_{k,0}(s), \gamma_{k,1}(s), \ldots, \gamma_{k,D_2}(s))^\intercal$ is a vector of regression coefficient functions associated with the k^{th} component.

The covariates $\mathbf{x}$ and $\mathbf{z}$ are assumed to enter the model (1.1) linearly. The component regression coefficients, $\boldsymbol{\beta}_k$, of the covariates $\mathbf{x}$ are assumed to be constant across all values of the respective covariates. On the other hand, the component regression coefficients, $\boldsymbol{\gamma}_k(s)$, of the covariates $\mathbf{z}$ are assumed to be smooth unknown functions of the covariate s varying across all the values of the respective covariates. The covariates $\mathbf{t}$ are assumed to be characterised by a set of smooth unknown univariate functions $g_{k,c}(\cdot) : c = 1, 2, \ldots, D_3$. Thus, the first term of the CRF is parametric whereas the second and third term are non-parametric.

Focusing on the CRFs (1.2) for now, model (1.1) is a combination of a linear model, a varying coefficients model (see Hastie and Tibshirani [1993]) and an additive regression model (see Hastie and Tibshirani [1990]), given by the first, second and third terms in (1.2), respectively. Consequently, the desirable properties embodied in these models are inherited by model (1.1).

These properties include the interpretability of the covariate effects, the non-linear interaction between s and the covariates $\mathbf{z}$ and the flexibility of the univariate additive functions $g_{k,c}(t_c)$. Note that, it is not our interest in this thesis to estimate the number of mixture components K, thus we assume this to be known. Therefore, model (1.1) can be referred to as a finite Gaussian mixture of varying coefficients partially linear additive models (GMVCPLAMs). However, in chapter 2, we briefly discuss how K is calculated in practice.

If $D_1 = D_3 = 0$, then model (1.1) reduces to the non-parametric Gaussian mixture of varying coefficients models (NPGMVCMs) (Huang et al. [2018]). If $D_2 = D_3 = 0$ and the variances and mixing proportions are constant then model (1.1) reduces to a Gaussian mixture of linear regressions (GMLRs) (Quandt [1972] and Quandt and Ramsey [1978]). If $D_2 = 0$ and the variances and mixing proportions are assumed to be constant then model (1.1) reduces to a class of semi-parametric Gaussian mixture of partially linear additive models (SPGMPLAMs) (Zhang and Pan [2022]). Thus, model (1.1) is a natural extension of the GMLRs and many interesting Gaussian mixtures of semi- and non-parametric regression models first introduced and studied by Huang [2009] and subsequently by Huang and Yao [2012] and Huang et al. [2013]. These models have been applied in areas such as environmental economics (Huang and Yao [2012] and Huang et al. [2018]), growth economics (Huang et al. [2013], Xiang and Yao [2016]), development economics (Wu and Liu [2017] and Zhang and Pan [2022]) and more recently in sports (Xiang and Yao [2020]), among many other areas and potential areas of application.

In mixture modelling and statistical modelling, in general, it is important to consider the identifiability of the underlying model (see section 2.1.1 for more details). Failure to do this may lead to incorrectly defined estimation procedures. For the identifiability of model (1.1), we assume that each $\boldsymbol{\beta}_k$ does not include the intercept and $\mathbb{E}\{g_{k,c}(t_c)\} = 0$, for $c = 1, 2, \ldots, D_3$ and $k = 1, 2, \ldots, K$. We follow convention and assume that the covariates t_c, for $c = 1, \ldots, D_3$, take values on the compact interval $[a, b]$, where $b > a$.

In this thesis, we are interested in estimating a class of models of the form (1.1). In an effort to highlight the practical importance of model (1.1) and its relevance across various fields of research, in the remainder of this section, we provide a comprehensive description of some of the main special cases of model (1.1). This brief overview includes the models' specification, advantages, shortcomings and examples of their application in practice.

Non-parametric Gaussian mixture of non-parametric regressions (NPGMNRs)

If $D_1 = D_2 = D_3 = 0$, then model (1.1) reduces to the following form

$$f(Y|S = s) = \sum_{k=1}^{K} \pi_k(s)\mathcal{N}(Y|m_k(s), \sigma_k^2(s)). \tag{1.3}$$

Model (1.3) is the non-parametric Gaussian mixture of non-parametric regressions (NPGMNRs). The model was first introduced and studied by Huang [2009] and subsequently by Huang et al. [2013]. Model (1.3) is non-parametric in the sense that the mixing proportions, regression functions and variances are all smooth unknown (hence, non-parametric) functions of a univariate covariate s.

Model (1.3) was proposed as a flexible alternative to the traditional GMLRs model by relaxing the linearity assumption. Being fully non-parametric, model (1.3) is more robust than the GMLRs model as it does not suffer from the risk of being mispecified. Moreover, the fitted non-parametric functions can be used to suggest appropriate parametric forms for the respective component non-parameter functions. Model (1.3) is suitable for studying the relationship between two variables from a population made up of unknown homogeneous subpopulations (that is, unobserved heterogeneity). In addition, there is no explicit form to characterise this relationship. The practical use of model (1.3) was demonstrated by Huang et al. [2013].

In an analysis of how the size of an economy influences the size of its housing market, the authors proposed a $K = 2$ component NPGMNRs model for the United States (US) using monthly data for the period January 1990 to December 2002. The size of the US economy and housing market was measured by the gross domestic product (GDP) and housing price index, respectively. The authors interpreted the two components as two macro-economic cycles. The first economic cycle corresponds to the period January 1990 and September 1997 where an increase in the size the US economy had a positive effect on the housing market. The second economic cycle corresponds to the period October 1997 to December 2002 where a modest increase in the size of the US economy had a negative effect on the housing market.

Although it has greater flexibility than the GMLRs, model (1.3) has three major drawbacks. First, if a multivariate covariate $S = \mathbf{S} \in \mathbb{R}^D$, where $D >> 1$, is incorporated in the model, the model will suffer from the well-known *curse of dimensionality (COD)* that is present when estimating multivariate non-parametric functions (see subsection 2.2.4 for more details). This in turn limits the practical applicability of the model to low-dimensional problems. Second, for a fully non-parametric model such as model (1.3), the resulting non-parametric estimators are generally known to suffer from low convergence rates (Yousof and Gad [2015]). Third, in

estimating the non-parametric terms using a likelihood approach, we must use the EM algorithm to locally maximise the likelihood function of model (1.3). Doing this separately for each local-likelihood function may lead to a mismatch in the labels at different values of the non-parametric functions. The practical consequence of this is non-smooth or wiggly estimates of the non-parametric functions. The third drawback is the main theme of this thesis. In section 1.2, we describe the nature of this problem.

To address the first problem mentioned above, many proposals were made, some of which are mentioned later in this section. A more recent proposal was made by Xiang and Yao [2020]. The authors proposed non-parametric Gaussian mixtures of single index models (NPGMSIMs). The NPGMSIMs is a special case of model (1.3) when $s = \mathbf{s}^\intercal\boldsymbol{\alpha} \in \mathbb{R}$, where $\mathbf{s} \in \mathbb{R}^D$, $D >> 1$ and $\boldsymbol{\alpha}$ is a vector of single index parameters satisfying $\|\boldsymbol{\alpha}\| = 1$ for identifiability.

The NPGMSIMs overcomes the COD by incorporating multivariate covariates $\mathbf{s}$ as a univariate index $\mathbf{s}^\intercal\boldsymbol{\alpha}$. Given the estimate of $\boldsymbol{\alpha}$, say $\hat{\boldsymbol{\alpha}}$, we make use of $\mathbf{s}^\intercal\hat{\boldsymbol{\alpha}}$ as the covariate s in model (1.3). To address the second problem mentioned above, Huang and Yao [2012] proposed a semi-parametric Gaussian mixture of regressions with varying mixing proportions (SPGMRVPs). The SPGMRVPs model is a special case of model (1.3) when the CRFs are assumed to be linear functions and the variances are constant. Another proposal to address the second problem was made by Xiang and Yao [2016]. The authors proposed semi-parametric Gaussian mixtures of non-parametric regressions (SPGMNRs). The SPGMNRs model is a special case of model (1.3) when the mixing proportions and variances are assumed to be constant. The proposed models overcome the second problem because they are semi-parametric (they have both parametric and non-parametric terms). Theoretically, if we achieve optimal convergence rate ($\sqrt{n}-$consistency) for the parametric term, then we can improve the convergence rate of the non-parametric term (see Xiang and Yao [2016]).

Note that the CRFs of the SPGMRVPs can take the multivariate covariates, albeit parametrically, without suffering from the COD. However, the model might suffer from bias as a result of mis-specification if the imposed parametric form is not adequate. On the other hand, the CRFs of the SPGMNRs will suffer from the COD for multivariate covariates. However, the model will not suffer from bias due to misspecification. We discuss these models as well as their extensions in the following subsections.

Semi-parametric Gaussian mixture of non-parametric regressions (SPGMNRs)

In the previous paragraph, we mentioned that the SPGMNRs is a special case of the NPGMNRs (1.3) which makes it a special case of model (1.1). The model has the form

$$f(Y|S=s)=\sum_{k=1}^{K}\pi_k\mathcal{N}(Y|m_k(s),\sigma_k^2). \tag{1.4}$$

Model (1.4) is a semi-parametric model in the sense that it contains both a parametric term (π_k,σ_k^2) and a non-parametric term $(m_k(s))$. The model combines the flexibility of a non-parametric model and the parsimony of a parametric model. Thus, model (1.4) is a bridge between the GMLRs model (2.19) and the NPGMNRs (1.3). If the parametric assumption holds, the parametric estimators of model (1.3) enjoy $\sqrt{n}-$consistency which improves the efficiency of the non-parametric estimators (see Xiang and Yao [2016]). Model (1.4) is suitable for application in areas where the probability of belonging to a given subpopulation is not influenced by any known factor (or covariate). In addition, the assumption of homoscedasticity (see page 64 of Gujarati and Porter [2011]) holds for each subpopulation.
Xiang and Yao [2016] demonstrated the practical use of model (1.4) using the same data used by Huang et al. [2013]. The authors couldn't find statistical evidence in support of covariate varying mixing proportions and heteroscedastic (covarite-dependent) variances and thus proposed a $K=2$ component SPGMNRs. The two components corresponds to the same two macro-economic cycles found by Huang et al. [2013]. The $K=2$ component SPGMNRs provides a parsimonious explanation to the relationship between the US GDP and housing market. The authors also demonstrated that the predictive ability of their proposed model is slightly better than the model proposed by Huang et al. [2013] for the US GDP-house price data.
Despite its advantage over the NPGMNRs (1.3), model (1.4) has challenges of its own. Extending the model to include multivariate covariates in the CRFs is infeasible due to the COD. Thus, the practical use of model (1.4) is limited to low-dimensional covariates. This is concerning as most areas of interest in practice have high-dimensional covariates. For instance, in energy economics, the relationship between CO_2 emissions and a group of economic variables (such as, energy consumption, financial development and economic growth) may differ for different unobserved groups of countries at different stages of development. In medicine, the relationship between beta-carotene levels (concentrations of anti-oxidants in blood and body tissues that protect against oxidative stress) and factors such as age, gender, body mass index, dietary carotene and smoking status may differ for different groups of patients (Schlattmann [2009]). Moreover, the functional relationship between the response and some of the covariates

may be unknown.

Mixture of single index models

One way to deal with the dimensionality problem when modelling the non-parametric CRFs is to make use of a single index (Zeng [2012] and Xiang and Yao [2020]). A single index projects multiple covariates into a univariate covariate by making use of a linear combination of these covariates.

In model (1.4), let $s = \mathbf{s}^\intercal\boldsymbol{\alpha}$, where $\mathbf{s} \in \mathbb{R}^D$ with $D >> 1$ is a $D-$dimensional vector of covariates and $\boldsymbol{\alpha}$ is the corresponding $D-$dimensional vector of single index coefficients. Thus, model (1.4) has the following extended form

$$f(Y|\mathbf{S} = \mathbf{s}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(Y|m_k(\mathbf{s}^\intercal\boldsymbol{\alpha}_k), \sigma_k^2), \tag{1.5}$$

where $\|\boldsymbol{\alpha}_k\| = 1$ for identifiability and the first non-zero element of $\boldsymbol{\alpha}_k$, for $k = 1, 2, \ldots, K$, is positive. Model (1.5) is a special case of the semi-parametric Gaussian mixtures of heteroscedastic single index models (SPGMHSIMs) (Zeng [2012]) when the component variances are constant and the NPGMSIMs (Xiang and Yao [2020]) when both the component variances and mixing proportions are constant. Note that model (1.5) can be obtained from model (1.1) by letting $D_2 = D_3 = 0$, $s = \mathbf{x}^\intercal\boldsymbol{\beta}_k$ and taking the mixing proportions and variances to be constants.

Model (1.5) has all the advantages of model (1.4) and more. Model (1.5) can non-parametrically take multiple covariates without suffering from the COD. If the CRFs are monotonic functions, the single index parameters $\boldsymbol{\alpha}_k$ have the same meaning and interpretation as in traditional mixture of linear models (Carroll et al. [1997]). This interpretability feature makes them relevant for describing the data within, and conducting meaningful statistical inference about, each sub-population. Since model (1.5) combines the features of a GMLRs model, a single index model and a SPGMNRs model, it can be referred to as a semi-parametric Gaussian mixtures of single index models (SPGMSIMs).

In demonstrating the practical usefulness of the NPGMSIMs, Xiang and Yao [2020] considered a study of the dependence of a company's market value, as determined by the stock market, on company fundamentals (such as, cash flow, earnings per share, return on equity etc.). For a sample of 196 manufacturing companies for the year 2017, the authors proposed a $K = 2$ component NPGMSIMs. The authors compared the proposed model with, amongst other models, the GMLRs model. The GMLRs model was found to have slightly better predictive ability compared to the NPGMSIMs. This might possibly be attributable to the strong constraint

imposed on the index parameter $\boldsymbol{\alpha}_k$. The authors assume that $\boldsymbol{\alpha}_k = \boldsymbol{\alpha}$, for $k = 1, 2, \ldots, K$. Another possible explanation of this outcome might be due to the over-relaxation of the linearity assumption. By fitting a NPGMSIMs which assumes that all the model parameters are non-parametric functions without any statistical evidence, there is a possibility that the model might overfit the underlying data. It would be interesting to see the predictive performance of model (1.5) on the same data.

Another way to address the dimensionality problem when modelling multivariate non-parametric CRFs is to use an additive structure. In its general form, an additive structure is a linear combination of a parametric function (usually linear) of some of the covariates and a sum of univariate non-parametric functions of the rest of the covariates (see page 150 of Härdle et al. [2004]). This gives rise to a class of mixtures of regression models referred to as semi-parametric Gaussian mixtures of partially linear additive models (SPGMPLAMs) (Zhang and Pan [2022]). The SPGMPLAMs is a subclass of model (1.1). We discuss this class of models in the next section.

Semi-parametric Gaussian mixture of partially linear additive models (SPGMPLAMs)

In order to handle multivariate covariates when fitting the SPGMNRs model (1.4), various extensions of this model were proposed. The proposed models have the general form

$$f(Y|\mathbf{X} = \mathbf{x}, \mathbf{T} = \mathbf{t}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(Y|m_k(\mathbf{x}, \mathbf{t}), \sigma_k^2). \tag{1.6}$$

Model (1.6) is a special case of (1.1) when $D_2 = 0$ and the variances and mixing proportions are assumed to be constant. Thus, from (1.2), the CRFs have the form

$$m_k(\mathbf{x}, \mathbf{t}) = \mathbf{x}^\intercal \boldsymbol{\beta}_k + \sum_{c=1}^{D_3} g_{k,c}(t_c), \qquad k = 1, 2, \ldots, K. \tag{1.7}$$

Model (1.6) has some interesting features. First, it retains the parsimony and flexibility of model (1.1). Second, through optimal convergence of the estimators of the first term in (1.7) it can improve the convergence rate of the estimators of the last term in (1.7). Finally, it overcomes the dimensionality problems because the multivariate covariate is characterised by a sum of univariate functions of the covariates. All these features render model (1.6) to be a suitable model for describing the data and conducting meaningful statistical inference.

Mixtures of partially linear models

The first appearance of a model of the form (1.6) was due to Wu and Liu [2017]. For a simple setup where only one covariate has a non-parametric relationship with the response variable, that is $D_3 = 1$, the authors proposed a semi-parametric Gaussian mixture of partially linear models (SPGMPLMs). The SPGMPLMs has the form

$$f(Y|\mathbf{X} = \mathbf{x}, T = t) = \sum_{k=1}^{K} \pi_k \mathcal{N}(Y|m_k(\mathbf{x}, t), \sigma_k^2), \tag{1.8}$$

where the CRFs are given by

$$m_k(\mathbf{x}, t) = \mathbf{x}^\intercal \boldsymbol{\beta}_k + g_k(t), \qquad k = 1, 2, \ldots, K. \tag{1.9}$$

Model (1.8) provides an even more parsimonious version of model (1.6). The authors demonstrated the practical significance of the SPGMPLMs in an analysis of how the size and growth of an economy is influenced by three socio-economic variables including the level of education, human capital and real capital stock. Using annual data on 82 countries for the period 1960 to 1987, the authors proposed a $K = 2$ component SPGMPLMs. The authors made use of GDP, mean years of schooling, number of individuals in the workforce and aggregate physical capital stock to measure the size of an economy, the level of its education (education), human capital (labour) and real capital stock (capital), respectively. A comprehensive description of the data used and how it was pre-processed is provided by Duffy and Papageorgiou [2000].

The proposed model assumes that labour and capital are linearly related to the size of an economy (in logarithmic terms), which is consistent with the Cobb-Douglas specification. On the other hand, the relationship between education and the size of an economy is not linear and hence assumed to be non-parametric. The authors estimated the model and found that the regression coefficients had the expected signs.

More recently, based on the same data as above, Zhang and Pan [2022] found significant statistical evidence in support of a non-parametric relationship between labour and the size of an economy. The authors therefore proposed a $K = 2$ component SPGMPLAMs with $D_1 = 1$ and $D_3 = 2$ in (1.7). The authors omitted the interpretation of their results. However, an examination of the results reveal two features. First, a similar explanation of the non-parametric relationship between GDP and education, as in the SPGMPLMs, continues to hold for the SPGMPLAMs. Second, the fitted non-parametric relationship between GDP and labour appears to be only applicable to the first component made up of developed countries and not necessary for the second component made up of developing countries. The CRF of the second component appears to be well fitted using a linear functional form (see Zhang and Pan [2022]

for more details on the fitted model). Let x_1, x_2 and t represent capital, labour and education, respectively. The above observation suggests a model of the form

$$f(Y|\mathbf{X}=\mathbf{x},T=t)=\pi_1\mathcal{N}(Y|\beta_{0,1}+\beta_{1,1}x_1+g_1(x_2)+g_1(t),\sigma_1^2)+ \\ \pi_2\mathcal{N}(Y|\beta_{0,2}+\beta_{1,2}x_1+\beta_{1,2}x_2+g_2(t),\sigma_2^2), \quad (1.10)$$

where the g's are non-parametric functions. A specification test can be conducted to check the adequacy of model (1.10). Since this thesis is only concerned with estimation, specification testing is outside of its scope. Thus, we put the above forward as an idea for future research. The estimation tools proposed in this thesis will play a pivotal role in a project on hypothesis testing.

Mixtures of additive regression models

Another interesting model of the form arises when $D_1 = 0$ which is a reduced form of model (1.6) when the CRFs are made up of only the second term in (1.7). The resulting model has the form

$$f(Y|\mathbf{T}=\mathbf{t})=\sum_{k=1}^{K}\pi_k\mathcal{N}(Y|m_k(\mathbf{t}),\sigma_k^2), \quad (1.11)$$

where the CRFs are given by

$$m_k(\mathbf{t})=\sum_{c=1}^{D_3}g_{k,c}(t_c), \qquad k=1,2,\ldots,K. \quad (1.12)$$

Model (1.11) is a semi-parametric Gaussian mixture of additive models (SPGMAMs) proposed by Zhang and Zheng [2018]. Model (1.11) is most suitable to study the dependence of a response variable on a set of covariates, where the response variable is from a population that is made up of unknown homogeneous sub-populations. In addition, there is no parametric functional form that relates each covariate with the response variable. The model is in general suitable even if we can assume a parametric form. Using model (1.11) will avoid the risk of mis-specification. However, because of the non-parametric additive functions, the non-parametric estimators will have slow convergence rate. This will not be a problem for a large enough sample size, especially if we use local-linear methods.

Other interesting models can be obtained as special cases of model (1.6), most of which have not been studied and deserve to be considered for future research. We mention only one other special case of model (1.6), mainly because of its potential for application in practice.

Mixtures of partially linear single index models

Let $t = \mathbf{t}^\intercal \boldsymbol{\alpha}_k$, for $k = 1, 2, \ldots, K$, in (1.8), where $\mathbf{t} \in \mathbb{R}^D$ with $D >> 1$. The CRF of model (1.8) can be written as

$$m_k(\mathbf{x}, \mathbf{t}) = \mathbf{x}^\intercal \boldsymbol{\beta}_k + g_k(\mathbf{t}^\intercal \boldsymbol{\alpha}_k), \quad k = 1, 2, \ldots, K, \tag{1.13}$$

where $\|\boldsymbol{\alpha}_k\| = 1$, for $k = 1, 2, \ldots, K$, for identifiability.
Note that if $\boldsymbol{\beta}_k = 0$, for $k = 1, 2, \ldots, K$, then model (1.8) reduces to model (1.5). Thus, the resulting model combines combines the features of a SPGMPLMs (1.8) and a SPGMSIMs (1.5). The model can be referred to as a semi-parametric Gaussian mixtures of partially linear single index models (SPGMPLSIMs). The SPGMPLSIMs is a natural extension of the partially linear single index model (Carroll et al. [1997]) for data that arises from a population that is made up of unknown sub-populations (or simply unobserved heterogeneity).
An interesting potential application of the SPGMPLSIMs appears in corporate finance. The literature on corporate finance argues that there is a target level of leverage (debt) that maximises the value of a company (Durand et al. [2022]). The speed at which a company adjusts toward this value is of interest to various stakeholders (such as investors and policy makers). Recent literature on corporate finance is in agreement that there is cross-company heterogeneity in the speed of adjustment (SOA). That is, there exist groups of companies each with its own SOA and the behaviour of each company within each group is different from that of companies in other groups. However, there is no agreement as to the determinants of this heterogeneity (see Durand et al. [2022] for a brief review of the literature). Motivated by this lack of consensus, Durand et al. [2022] proposed an endogenous (data-driven) approach to identify the groups using mixture models. The authors proposed the following GMLRs model

$$f(Y|Y_{-1} = y_{-1}, \mathbf{X} = \mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(y|m_k(y_{-1}, \mathbf{x}), \sigma_k^2). \tag{1.14}$$

The CRFs have the form

$$m_k(y_{-1}, \mathbf{x}) = \delta_k y_{-1} + \mathbf{x}^\intercal \boldsymbol{\beta}, \quad k = 1, 2, \ldots, K, \tag{1.15}$$

where $\lambda_k = 1 - \delta_k$ represents the SOA of companies in the k^{th} group.
The response variable y is given by leverage (debt), y_{-1} is a one-period lagged version of y and $\mathbf{x}$ is a vector of company-specific characteristics (such as, company size and profitability). A flexible alternative to model (1.14) can be obtained using SPGMPLSIMs. The CRFs (1.15)

now have the form

$$m_k(y_{-1}, \mathbf{x}) = \delta_k y_{-1} + g_k(\mathbf{x}^\intercal \boldsymbol{\beta}), \quad k = 1, 2, \ldots, K, \tag{1.16}$$

where $g_k(\cdot)$ is assumed to be a monotone smooth unknown function of the index $\mathbf{x}\boldsymbol{\beta}$. The model specification in (1.16) is robust against a possible mis-specification if the the linearity assumption in (1.15) is not appropriate. Besides, even if the linearity assumption holds, the regression (index and SOA) parameters of the SPGMPLSIMs will have the same interpretation as in the GMLRs model.

Semi-parametric Gaussian mixture of regressions with varying mixing proportions (SPGMRVPs)

The SPGMRVPs model was briefly mentioned in subsection 1.1 as a special case of the NPGMNRs model (1.3) when $\sigma_k^2(s) = \sigma_k^2$ for all s and $m_k(s)$ is a parametric (linear) function of the covariates $\mathbf{x}$. This implies that the model is also a special case of the general model (1.1). To see this, let $D_2 = D_3 = 0$ in model (1.1), the SPGMRVPs results as

$$f(Y|S = s, \mathbf{X} = \mathbf{x}) = \sum_{k=1}^{K} \pi_k(s) \mathcal{N}(Y|\mathbf{x}^\intercal \boldsymbol{\beta}_k, \sigma_k^2). \tag{1.17}$$

Note that s can be part of $\mathbf{x}$. Model (1.17) is semi-parametric in the sense that it contains a parametric term $(\boldsymbol{\beta}_k, \sigma_k^2)$ and a non-parametric term $(\pi_k(s))$. Model (1.17) is more flexible than the GMLRs model and it is not subject to modelling bias in case π_k depends on some covariate. The CRFs of model (1.17) can take multi-dimensional covariates without suffering from the curse-of-dimensionality (COD) (see section 2.2.4) as is the case with model (1.3). Thus, model (1.17) enjoys the best of both the GMLRs model and model (1.3).
Model (1.17) is more suitable for application in areas where the probability of belonging to a given sub-population depends on the state or value of some covariate. That is, the covariate contains some information about the mixing proportions π_k, for $k = 1, 2, \ldots, K$.
Model (1.17) was first introduced and studied by Young and Hunter [2010] for a multi-dimensional covariate $\mathbf{s} \in \mathbb{R}^D$, where $D >> 1$. Due to the COD, the model (1.17) is not useful for large D. Theoretically, its non-parametric estimator has slow convergence rate. In practice, as pointed out by the authors, the performance of the estimators may deteriorate as D increases. An earlier version of the model (1.17) version of Young and Hunter [2010] appeared in the machine learning literature as a mixture of experts (MEs) model (Jacobs et al. [1991]). The MEs model assumes that the mixing proportion functions $\pi_k(\mathbf{s})$ have a parametric form (see chapter 12 of Fruhwirth-Schnatter et al. [2019] for an introduction to the MEs model).

Unfortunately, the parametric assumption renders the model inflexible and subject to bias if the assumption is inappropriate. Besides, a non-parametric form is usually preferred over a parametric form as it can assist in verifying a putative parametric form or even discover a new form.

Recent efforts to effectively address the COD problem without making any parametric assumptions about the form of the mixing proportions, include Xiang and Yao [2020] who proposed a semi-parametric Gaussian mixture of regressions with varying single index proportions (SPGM-RVSIPs). The authors makes use of a single index $\mathbf{s}^\intercal\boldsymbol{\alpha} \in \mathbb{R}$ to non-parametrically model the mixing proportions as $\pi_k(\mathbf{s}^\intercal\boldsymbol{\alpha})$, where $\boldsymbol{\alpha}$ is a single index parameter vector. More recently, Xue and Yao [2022] proposed to non-parametrically estimate the multi-covariate mixing proportion functions $\pi_k(\mathbf{s})$ using a neural network.

Model (1.17) has proven to be of practical use especially to the nascent area of model-based clustering (see chapter 8 of Fruhwirth-Schnatter et al. [2019] for an introduction to model-based clustering). Consider any data point $(s_1, \mathbf{x}_1, y_1)$, we can use model (1.17) to assign the data point to a specific component (cluster) based on which $\pi_k(s_1)$, for $k = 1, 2, \ldots, K$, is the largest. As a real practical example, Young and Hunter [2010] used model (1.17) in a regression analysis of carbon dioxide (CO_2) emissions per capita on gross national product (GNP) per capita for a group of 28 countries in 1996. The authors fitted a $K = 2$ component model where the first group of countries follow a negative linear relationship between CO_2 per capita and GNP per capita (low CO_2 and high GNP cluster). On the other hand, the second group of countries follow a positive linear relationship between CO_2 per capita and GNP per capita (high GNP and high CO_2 cluster). The authors further note that as GNP per capita increases, the probability that a country belongs to a low CO_2 and high GNP cluster increases. For a similar mixture regression analysis, with GNP replaced by gross domestic product (GDP), using 2005 data for a group of 171 countries, Huang and Yao [2012] reached the same conclusion. The authors observed that, as per capita GDP increases, the proportion of low CO_2 and high GDP countries increases.

Another real-data application used model (1.17) to examine the impact of remittance (money transferred from abroad) inflows on economic growth over the period 1970-2010 for a group of 120 developing countries (see Konte [2018]). The author fitted a $K = 2$ component model which corresponds to two different growth regimes. In the first regime, labelled the remittances growth-enhancing regime, remittances were found to have a positive and significant impact on economic growth. On the other hand, in the second regime, the impact of remittances on growth was found to be insignificant. To model the mixing proportion functions, $\pi_k(s)$, the author used variables such as financial development and geographical location. Based on the resulting model, the author observed, among other things, that being a sub-saharan country

increases the probability of belonging to the remittances growth-enhancing regime.
Despite its theoretical and practical advantages, as noted above, model (1.17) still assumes that the CRFs are parametric functions of the covariates $\mathbf{x}$. The model is therefore subject to specification bias in the event that the parametric assumption does not hold. Moreover, given that model (1.17) is a semi-parametric model, having both parametric and non-parametric terms, the bias might affect the performance of the estimators of the non-parametric terms. A possible approach to address this challenge follows as an extension of model (1.5) to characterise each CRF as a non-parametric function of a single index $\mathbf{x}^\intercal\boldsymbol{\beta}_k$. This can be easily seen as a combination of model (1.5) and model (1.17). The resulting model has the form

$$f(Y|S=s,\mathbf{X}=\mathbf{x})=\sum_{k=1}^{K}\pi_k(s)\mathcal{N}(Y|m_k(\mathbf{x}^\intercal\boldsymbol{\beta}_k),\sigma_k^2), \tag{1.18}$$

where $\boldsymbol{\beta}_k$ is the single index parameter vector of the k^{th} component.
Model (1.18) is a parsimonious version of the NPGMSIMs and has more flexibility than both model (1.5) and (1.17). Among other things, this implies that, in general, the non-parametric estimators of model (1.18) are more efficient than the estimators of a NPGMSIMs and less biased than the estimators of either model (1.5) or model (1.17). Due to the presence of the parametric term (σ_k^2), model (1.18) is a semi-parametric model. Moreover, due to its form, we can refer to it as a semi-parametric Gaussian mixture of single index models with varying single index proportions (SPGMSIMVSIPs).
A recent proposal to address this challenge is given in Xue and Yao [2022]. The author proposed to use a multi-layered neural network to model both the mixing proportion functions and the CRFs.
In addition to addressing the concern over the specification bias, the above proposals also avoid the issue of the curse-of-dimensionality.

Non-parametric Gaussian mixture of varying coefficient models (NPGMVCMs)

If $D_1 = D_3 = 0$ then model (1.1) reduces to

$$f(Y|S=s,\mathbf{Z}=\mathbf{z})=\sum_{k=1}^{K}\pi_k(s)\mathcal{N}(y|m_k(s,\mathbf{z}),\sigma_k^2(s)), \tag{1.19}$$

where the CRFs are given by

$$m_k(s,\mathbf{z})=\mathbf{z}^\intercal\boldsymbol{\gamma}_k(s),\qquad k=1,2,\ldots,K. \tag{1.20}$$

Model (1.19) is the finite non-parametric Gaussian mixture of varying coefficient models (NPG-MVCMs) proposed by Huang et al. [2018] when $\phi(\cdot)$ is a Gaussian density function, that is $\phi(\cdot) = \mathcal{N}(\cdot)$. The model assumes that the CRFs are linear in the covariates $\mathbf{z}$, however the coefficients of each of these covariates $\mathbf{z}$ are non-parametric functions of a covariate s. The covariate s is known as an effect modifier Hastie and Tibshirani [1993] or tuning variable Xue and Yang [2006].

Note that, in general, each coefficient function $\gamma_{k,b}(\cdot)$, for $b = 0, 1, \ldots, D_2$, can be a function of a unique covariate. That is, for any $k = 1, 2, \ldots, K$, we can have $\gamma_{k,b}(s_j)$ and $\gamma_{k,b^*}(s_{j^*})$, where $b \neq b^*$ and $j \neq j^*$.

Model (1.19) is fully non-parametric in the sense that all its estimable quantities ($\pi_k(\cdot)$,$\boldsymbol{\gamma}_k(\cdot)$ and $\sigma_k^2(\cdot)$) are assumed to be unknown functions of the covariate s. Even though it is a special case of the general model (1.1), model (1.19) is itself general including, as special cases, some of the models that we have already discussed, see Huang et al. [2018] for more details.

Due to the semi-parametric nature of the CRFs, parametric in the covariates $\mathbf{z}$ and non-parametric in the covariate s, model (1.19) is more flexible compared to the GMLRs model. The model enjoys the interpretability feature of a GMLRs model without making the rigid assumption of a constant change, that is $\gamma_{b,k}(s) \neq \gamma_{k,b}$ for all s. By allowing the coefficients to be smooth functions of the covariate s, the model admits non-linear interactions between s and the covariates $\mathbf{z}$. Moreover, because the coefficient functions are univariate functions of a covariate s, model (1.19) is able to avoid the COD.

The practical significance of model (1.19) was demonstrated by Huang et al. [2018] in an application from environmental economics. The authors studied the relationship between carbon dioxide (CO_2) emissions (as the response y) and GDP per capita (as the covariate z) for a panel of 171 countries over the period 1996 to 2005. The authors fitted model (1.19) with $K = 2$. This corresponds with two (2) groups of countries on different development paths (a low CO_2-high GDP group and a high CO_2-low GDP group). In their study, the authors were interested in investigating the evolution of the development paths of the countries over time. Thus, their tuning variable, s, was time. The authors found that both groups of countries have been able to decrease their CO_2 emissions over the 10-year period, albeit at a different rate with the low CO_2-high GDP group decreasing faster than the high CO_2-low GDP group. This is one of many possible applications of model (1.19).

As another potential use of model (1.19), consider a study of cross-country growth. Assume that countries at the same stage of development grow at the same rate. Thus, we will have different groups of countries at different stages of development, say developing and developed. Using model (1.19), we can study the evolution or path of the economic growth rate over time as a function of, say technological progress or human capital, for countries at both stages of

development. As a further potential use of model (1.19), consider the environmental application in Fan and Zhang [1999] in which interest is in studying the association between the levels of pollutants ($\mathbf{z}$) and the number of daily total hospital admissions (y) for respiratory and circulatory problems. Suppose, instead, that we are interested in the change in the daily hospital admissions between two consecutive days. Moreover, suppose that the prevailing temperature has an influence on the hospital admissions in such a way that, below a certain (unknown, hence unobserved) level of temperature, hospital admission y increase faster as a function of the pollutants $\mathbf{z}$. Thus, the data is no longer homogeneous, as in Fan and Zhang [1999], it now consists of two unknown sub-populations. In this case , the association between y and $\mathbf{z}$ over time s can be best studied using model (1.19) with constant mixing proportions and variances. Model (1.19) is not without any shortcomings. In theory, due to its high level of flexibility, its non-parametric estimators have a slow rate of convergence. In practice, (1) if $\mathbf{s} \in \mathbb{R}^D$, where $D >> 1$, the model will suffer from the COD; (2) a data-driven approach, such a cross-validation method, to choose the optimal degrees of smoothness for each non-parametric function ($\pi_k(s)$,$\boldsymbol{\gamma}_k(s)$ and $\sigma_k^2(s)$) may be computationally too expensive; and (3) as with all the other models discussed so far, traditional likelihood estimation of the model via the EM algorithm may lead to label-switching. The first practical problem was addressed by Xue and Yang [2006] in the homogeneous data setting ($K = 1$). They proposed to model the multivariate coefficient functions $\boldsymbol{\gamma}_k(\mathbf{s}) = (\gamma_{k,0}(\mathbf{s}), \gamma_{k,1}(\mathbf{s}), \ldots, \gamma_{k,D_2}(\mathbf{s}))$ as additive functions of the covariates $\mathbf{s} = (s_1, s_2, \ldots, s_D)$ as

$$\gamma_{k,b}(\mathbf{s}) = \sum_{d=1}^{D} \gamma_{k,b}(s_d). \tag{1.21}$$

This thesis is devoted towards addressing the third practical problem mentioned above. More details will be provided in the following section.

1.2 Motivation

From the previous section, it is clear that model (1.1) is a rich and flexible model with broad applicability in various areas of research. Unfortunately, a naively implemented likelihood-based approach to the estimation of this model can lead to misleading and often unsatisfactory results. In practice, non-parametric functions are estimated at a set of local points in the domain of the covariate (s). A likelihood-based approach requires that we maximise a set of local-likelihood functions defined at each of the local points. Since these are likelihood functions of a mixture model, they can be maximised using the EM algorithm. However, if we maximise each local-likelihood function separately, the estimates may be subject to *label-switching*. This

is because the posterior probabilities (henceforth referred to as the responsibilities) calculated at the E-step of each local EM implementation are not guaranteed to be aligned across the local points. In the event of a misalignment, the estimated component non-parametric functions will be wiggly and non-smooth and consequently not useful in practice. This is the problem that gave rise to this thesis and our quest towards addressing it.
In this section, we give a brief overview of the label-switching problem. A comprehensive description of this phenomenon will be provided in section 3.2. Next, we discuss the aims, objectives and contributions of this thesis. Finally, we give the outline of the thesis and a graphical summary of this thesis, see Figure 1.1.

1.2.1 The label-switching problem

To aid the reader's comprehension of the problem under study, the following discussion will be based on model (1.3). Let $\{(s_i, y_i) : i = 1, 2, \ldots, n\}$ be a random sample of size n from model (1.3). The corresponding log-likelihood function is given as

$$\ell\{\boldsymbol{\theta}\} = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k(s_i)\mathcal{N}\left(y_i|m_k(s_i), \sigma_k^2(s_i)\right)\right], \tag{1.22}$$

where $\boldsymbol{\theta} = (\boldsymbol{\pi}, \boldsymbol{\sigma}^2, \mathbf{m}) = (\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_K; \boldsymbol{\sigma}_1^2, \ldots, \boldsymbol{\sigma}_K^2; \mathbf{m}_1, \ldots, \mathbf{m}_K)^\intercal$, with
$\boldsymbol{\pi}_k = (\pi_k(s_1), \pi_k(s_2), \ldots, \pi_k(s_n))^\intercal$, $\boldsymbol{\sigma}_k^2 = (\sigma_k^2(s_1), \sigma_k^2(s_2), \ldots, \sigma_k^2(s_n))^\intercal$ and
$\mathbf{m}_k = (m_k(s_1), m_k(s_2), \ldots, m_k(s_n))^\intercal$. To estimate $\boldsymbol{\theta}$, we maximise the following locally weighted log-likelihood function

$$\ell\{\boldsymbol{\theta}(u)\} = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k(u)\mathcal{N}\left(y_i|m_k(u), \sigma_k^2(u)\right)\right] K_h(s_i - u), \tag{1.23}$$

where $\boldsymbol{\theta}(u) = (\boldsymbol{\pi}(u), \boldsymbol{\sigma}^2(u), \mathbf{m}(u))$ is a vector of the local parameters at the local point u in the domain of the covariate s, $K_h(\cdot) = K(\cdot/h)/h$ is a rescaled kernel function $K(\cdot)$ used to assign weights to the data points in the neighborhood of a given local point and $h > 0$ is the bandwidth. See subsection 2.2.2 for more details.

Naïve EM algorithm

The local-likelihood function (1.23) can be maximised using the EM algorithm. Let $\{(s_i, y_i, \mathbf{z}_i) : i = 1, 2, \ldots, n\}$ be the complete-data, where $\mathbf{z}_i = (z_{i1}, z_{i2}, \ldots, z_{iK})^\intercal$ is a latent variable with $z_{ik} = 1$, if (s_i, y_i) is from the k^{th} component and zero otherwise. The corresponding complete-

data version of the likelihood function (1.23) is given by

$$\ell^c\{\boldsymbol{\theta}(u)\} = \sum_{i=1}^{n}\sum_{k=1}^{K} z_{ik}\Big[\log\{\pi_k(u)\} + \log\mathcal{N}\big(y_i|m_k(u),\sigma_k^2(u)\big)\Big] \times K_h(s_i-u). \tag{1.24}$$

At the $(r+1)^{th}$ iteration of the E-step, we calculate the expected value of (1.24), denoted by $Q(\boldsymbol{\theta}(u)|\boldsymbol{\theta}^{(r)}(u))$, with respect to the conditional distribution of $\mathbf{z}$. This reduces to calculating the expected value of z_{ik} as

$$p_{ik}^{(r+1)}(u) = \frac{\pi_k^{(r)}(u)\mathcal{N}\big(y_i|m_k^{(r)}(u),\sigma_k^{2(r)}(u)\big)}{\sum_{\ell=1}^{K}\pi_k^{(r)}(u)\mathcal{N}\big(y_i|m_k^{(r)}(u),\sigma_k^{2(r)}(u)\big)}, \tag{1.25}$$

for $i = 1,2,\ldots,n$, $k = 1,2,\ldots,K$ and all $u \in \mathcal{U}$, where $\mathcal{U}$ is the set of grid points on the domain of s.

At the $(r+1)^{th}$ M-step, we maximise $Q(\boldsymbol{\theta}(u)|\boldsymbol{\theta}^{(r)}(u))$, obtained by substituting z_{ik} with $p_{ik}^{(r+1)}$ (1.25), to update the local parameters $\pi_k^{(r)}(u)$, $\sigma_k^{2(r)}(u)$ and $m_k^{(r)}(u)$, for all $u \in \mathcal{U}$, using the expressions

$$\pi_{k0}^{(r+1)}(u) = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)}(u)K_h(s_i-u)}{\sum_{i=1}^{n} K_h(s_i-u)}, \tag{1.26}$$

$$m_{k0}^{(r+1)}(u) = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)}(u)K_h(s_i-u)y_i}{\sum_{i=1}^{n} p_{ik}^{(r+1)}(u)K_h(s_i-u)}, \tag{1.27}$$

$$\sigma_{k0}^{2(r+1)}(u) = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)}(u)K_h(s_i-u)[y_i - m_k^{(r+1)}(u)]^2}{\sum_{i=1}^{n} p_{ik}^{(r+1)}(u)K_h(s_i-u)}, \tag{1.28}$$

where $\pi_{k0}^{(r+1)}(u)$, $m_{k0}^{(r+1)}(u)$ and $\sigma_{k0}^{2(r+1)}(u)$ are the local-constant (or Nadaraya-Watson) estimators at local point u. See section 2.2.1 for more details on local estimation. Note that, at each local point u, we are estimating a Gaussian mixture model (GMM) (2.1). Thus, the derivation of the above estimators follows from the derivation of the parameters of a GMM (see section 2.1.2).

Repeat the above E- and M-steps until convergence. Let $\pi_k^{(R)}(u)$, $\sigma_k^{2(R)}(u)$ and $m_k^{(R)}(u)$, for all $u \in \mathcal{U}$, be the local parameter estimates at convergence of the above EM algorithm, where R is the iteration index at convergence. To obtain $\pi_k(s_i)$, $\sigma_k^2(s_i)$ and $m_k(s_i)$, for $i = 1,2,\ldots,n$, we linearly interpolating over $\pi_k^{(R)}(u)$, $\sigma_k^{2(R)}(u)$ and $m_k^{(R)}(u)$ for all $u \in \mathcal{U}$, for all $u \in \mathcal{U}$, respectively.

From the E-step (1.25), we can see that the responsibilities $p_{ik}(u)$, for $i = 1,2,\ldots,n$ and

$k = 1, 2, \ldots, K$, are calculated at different local points $u \in \mathcal{U}$. This implies that at each local point $u \in \mathcal{U}$, we have an estimate of the latent variable z_{ik} given by $p_{ik}(u)$. The responsibilities are not guaranteed to match across all local points. In the event of a mismatch we have label-switching which is characterised by discontinuous jumps at all the local points where the label switch has occurred. Moreover, the non-parametric function estimates will be non-smooth and exhibit irregular behaviour near the local points where the label switch took place. Stated differently, for each $u \in \mathcal{U}$, the M-step is based on a unique set of local responsibilities $\{p_{ik}(u) : i = 1, 2, \ldots, n; k = 1, 2, \ldots, K\}$. These sets of local responsibilities are not guaranteed to be aligned across the grid points. A misalignment between one or more local points may result in the switch of the labels of the components. For instance, suppose that, for any $k \neq k^*$, $m_k(s) > m_{k^*}(s)$. In the event of a misalignment, say $p_{ik}(u) > p_{ik}(u^*)$, for $i = 1, 2, \ldots, n$ and $u \neq u^*$, the labels on the local parameters $m_k(u)$ and $m_k(u^*)$ may switch such that $m_k(u) < m_{k^*}(u)$ and/or $m_k(u^*) < m_{k^*}(u^*)$. Suppose that $u < s < u^*$, we can obtain $\hat{m}_k(s)$ and $\hat{m}_{k^*}(s)$ by interpolation which results in $\hat{m}_k(s) < \hat{m}_{k^*}(s)$.

This form of label-switching problem was first mentioned by Huang [2009] and then subsequently by Huang and Yao [2012] and Huang et al. [2013]. To address the problem, the authors proposed a modified EM algorithm. Briefly, the local E-steps $p_{ik}(u)$ are replaced by a single (global) E-step p_{ik}. In other words, the same responsibilities p_{ik} are used at each local M-step. More details about this algorithm are given in subsection 3.2.3.

As a forerunner for addressing the label-switching problem, the algorithm received widespread use in estimating a class of models of the form (1.1). To estimate the SPGMNRs model (1.4), Xiang and Yao [2016] proposed a local EM-type as well as a global EM-type algorithm both of which incorporate a global E-step as described above. To estimate the SPGMPLMs (1.8), Wu and Liu [2017] proposed a modified profile-likelihood EM algorithm that makes use of the global responsibilities to simultaneously estimate both the parametric and non-parametric terms. The two-stage spline-backfitted kernel (SBK) estimation procedure of Zhang and Zheng [2018] and Zhang and Pan [2022] incorporates an EM algorithm at each stage. The second stage of the procedure estimates each additive non-parametric function using the modified EM algorithm. Huang et al. [2018] used the modified EM algorithm to estimate the NPGMVCMs (1.19). To estimate the non-parametric Gaussian mixture of graphical models (NPGMGMs), Lee and Xue [2018] proposed a modified EM algorithm that introduces a global E-step. The proposed one-step kernel estimators of Xiang and Yao [2020] are obtained via the modified EM algorithm. The three-stage fully iterative backfitting (FIB) procedure of Xiang and Yao [2020] makes use of the global responsibilities to estimate the varying mixing proportion functions.

From the previous paragraph, it is clear that researchers in the area of mixture modelling, in particular, mixture of regression modelling, are becoming more interested in the study and

ultimately application of flexible mixtures of regressions of the form (1.1). The practical utility of these models cannot be understated as discussed in detail in section 1.1. Their flexibility endows them with a great potential to uncover unobserved complex non-linear relationships or patterns present in many fields of research. Moreover, it can be seen from the references that this is an emerging area of research with most of the contributions being made between 2018 - 2020. We believe that there are many more contributions that can and will be made following from model (1.1) as researchers in other fields adopt these models. However, one thing that can hold back progress in the study and practical use of GMNRs is the label-switching problem and its practical implications. To our knowledge, the label-switching problem in the context of fitting a GMNRs model is not well covered in the literature and thus it is not well understood. For this reason, in this thesis we give a comprehensive description, including graphical and simulation-based illustrations, of the label-switching problem under study. Thereafter, we propose EM-type estimation strategies that have a dual purpose of addressing label-switching and producing practically useful estimates of the model's parametric and non-parametric terms. In contrast to the approach proposed by Huang [2009], the proposed methods make use of the local information to address the problem. As we demonstrate in chapter 5, the local responsibilities can provide information that is useful over and above addressing the label-switching problem.

To provide the reader an appreciation of the effectiveness and usefulness of the proposed estimation methods for addressing label-switching, we will use these methods to estimate some of the models mentioned in section 1.1 using simulated and real data in a series of essays. For our purpose in this thesis, an essay has the same form as a journal article. Thus, the results presented in some of these essays have been published in peer-reviewed journals.

1.3 Aims and objectives of this thesis

In summary, this thesis aims to achieve the following objectives:

1. To develop new methods to address the label-switching problem.
2. To give an exposition of Gaussian mixtures of non-parametric regressions of the form (1.1) and their practical utility in other fields of research.
3. To demonstrate the ease of estimating (training) these models by developing comprehensive algorithms for fitting the general model (1.1) and some of its special cases. For illustrative purposes, we will focus on the following models
   (a) The NPGMNRs (1.3), which assumes that the mixing proportions, CRFs and variances are non-parametric functions of a covariate.

(b) The SPGMNRs (1.4), which is a special case of model (1.3) where only the mixing proportions and variances are constants.

(c) The SPGMPLMs (1.8), which is an extension of model (1.4) where the CRFs are partial linear functions.

(d) The SPGMRVPs (1.17), which is a special case of model (1.3) where the CRFs are linear functions and the variances are constants.

Note that the models in (a) and (b) can take only one covariate whereas the models in (c) and (d) can take more than one covariate. These models are selected due to their unique features to demonstrate the wide applicability of the model (1.1) as well as the flexibility of the proposed estimation strategies.

1.4 Contributions of this thesis

This thesis makes the following contributions to the modelling and estimation of mixtures of regression models of the form (1.1):

1. we give a formal introduction of a general class of flexible Gaussian mixtures of a regression models (1.1) which is a natural extension of Gaussian mixtures of linear regressions where the CRFs are a linear combination of a linear term, a set of varying coefficient functions and a set of additive functions;

2. we give a formal description, illustration and a numerical demonstration of the label-switching problem encountered when estimating these models;

3. we develop new methods (the objective-based method and the model-based method) to estimate these models and address the label-switching problem; and

4. we demonstrate, using real data, how these models can be used to address questions that arise in various fields.

1.5 Thesis Outline

The rest of this thesis is structured as follows:

- **Chapter 2** reviews the literature on mixture models, in particular Gaussian Mixture Models (GMMs), and the local polynomial likelihood (LPL) estimation procedure for estimating non-parametric regression models. This chapter provides the preliminary building blocks of this research.

- **Chapter 3** presents two local polynomial likelihood (LPL)-based estimation procedures that we propose to estimate the general Gaussian mixture of semi-parametric regressions (GMNRs) model (1.1) and the consequent label-switching problem encountered if they are naively implemented via the EM algorithm. This chapter also seeks to give an overview of the label-switching problem studied in this thesis.

- **Chapter 4** presents the Objective-based estimation strategy we proposed to address label-switching. The chapter gives a generic description of the approach, its rationale and some computational considerations. Finally, the chapter presents two essays to demonstrate the effectiveness and practical usefulness of the proposed method on simulated and real datasets.

- **Chapter 5** presents the Model-based estimation strategy we proposed to address label-switching. Similar to **Chapter 4**, this chapter gives a description of the approach, its rationale and computational considerations. Finally, the chapter presents two essays to demonstrate the effectiveness and practical usefulness of the proposed method on simulated and real datasets.

- **Chapter 6** concludes the thesis and gives directions on how this research can be extended in the future.

- **Appendix A** provides a link to a public repository where the code used in this thesis has been made available.

Figure 1.1 gives a high-level visual summary of the thesis.

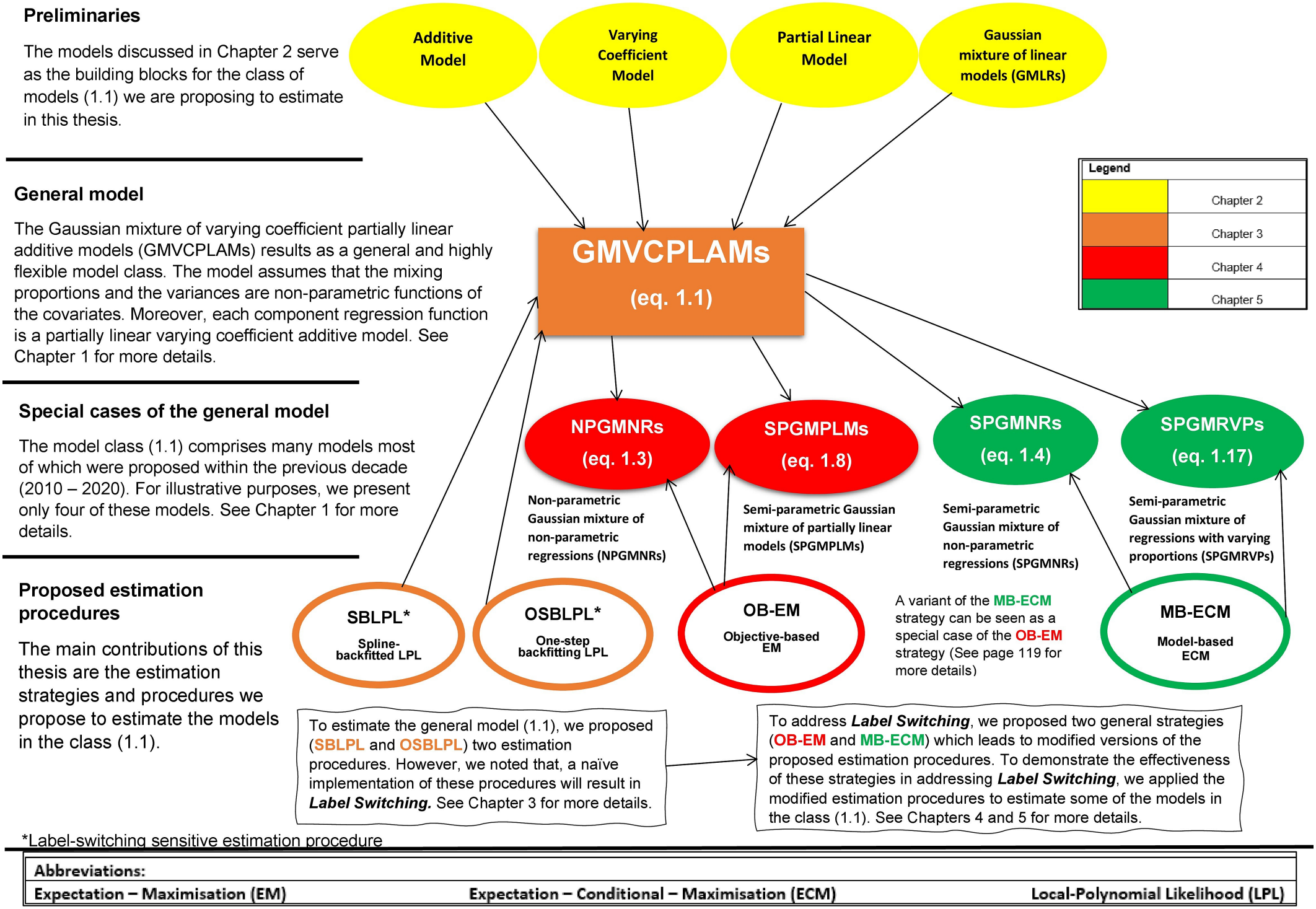


Figure 1.1: (*A high-level map of the thesis*): The map indicates all the estimation strategies proposed in this thesis to estimate the models in the class (1.1) and also address label-switching. The models and the estimation strategies are colour coded by the various chapters in which the corresponding estimation procedure is introduced in this thesis. See the legend for more details.

Chapter 2

Preliminaries

In this chapter, we briefly review the literature on mixture models, in particular Gaussian mixture models (GMMs). We also review the literature on non-parametric regression modelling using local polynomial likelihood (LPL) estimation. In chapter 3, we propose to estimate model (1.1) using LPL estimation procedures.

2.1 Gaussian mixture model (GMM)

The Gaussian distribution, $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, for a random variable $\mathbf{y} \in \mathbb{R}^D$, where $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ represents the mean vector and covariance matrix, respectively, is by far the most used probability distribution in practice due to the central limit theorem. It usually serves as a good approximate model for many natural processes, including crop yields in the agricultural sciences; health metrics in the health sciences and, more closer to the topic of this thesis, measurement errors, among many others. See any introductory book on probability and statistics for more details about the Gaussian distribution. For our purpose in this thesis, we take $D = 1$ and consider the univariate Gaussian distribution $\mathcal{N}(y|\mu, \sigma^2)$, where μ and σ^2 represents the mean and variance, respectively.

The Gaussian distribution is useful for data sampled from a homogeneous population. However, in many cases, the sampled population is made up of a number of usually *a priori* known, say K, sub-populations mixed randomly in proportion to the relative sub-population sizes $\pi_1, \ldots, \pi_K$. This phenomenon is frequently observed in zoology (Pearson [1894]), biology (Titterington et al. [1985]), economics (Frühwirth-Schnatter [2006]) and medicine (Schlattmann [2009] and Ng et al. [2019]), among many other fields in the natural, social and health sciences. In this case, y has a different Gaussian distribution in each sub-population. Moreover, because of the random mixing, we don't know the identity of the objects within each sub-population. Therefore, y is said to follow a finite mixture of Gaussian distributions or Gaussian mixture

model (GMM) given by

$$f(y|\boldsymbol{\theta}) = \pi_1\mathcal{N}(y|\mu_1,\sigma_1^2)+\cdots+\pi_K\mathcal{N}(y|\mu_K,\sigma_K^2) \quad (2.1)$$

$$= \sum_{k=1}^{K}\pi_k\mathcal{N}(y|\mu_k,\sigma_k^2), \quad (2.2)$$

where $\boldsymbol{\theta}=(\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_K)$, with $\boldsymbol{\theta}_k=(\pi_k,\mu_k,\sigma_k^2)$, is the vector of all the unknown model parameters. The density functions $\mathcal{N}(\cdot|\mu_k,\sigma_k^2)=f(\cdot|\mu_k,\sigma_k^2)$, for $k=1,2,\ldots,K$, are the component (sub-population) densities. The relative sizes, the π_k's, positive and summing to one, are also known as mixing proportions, weights or probabilities. They represent the probability that any given observation belongs to the k^{th} component. Note that, in general, K may also be an unknown parameter, but for our purpose in this thesis, we assume that K is known.

GMMs are a member of the broader class of mixture models, which includes, among others, Poisson mixture models, exponential mixture models (Titterington et al. [1985] and Frühwirth-Schnatter [2006] for a comprehensive coverage of finite mixture of distributions and models, respectively, and Fruhwirth-Schnatter et al. [2019] for a recent review of the general mixture model).

Earlier uses of the finite GMM include Newcomb [1886] for modelling outliers and Pearson [1894] for an analysis of zoological data. In the paper by Pearson, the word mixture appears for the first time and the GMM is presented as an "abnormal frequency curve" made up of "homogeneous normal [Gaussian] curves".

2.1.1 Identifiability

A theoretical and, most importantly, practical consideration in the analysis of finite mixture models is the identifiability of the model. An identifiable model guarantees the existence of a unique solution to the model under consideration. This is of course important in practice.

Briefly, a parametric model $f(y|\boldsymbol{\theta})$ is said to be identifiable if, for any two parameters $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^*$, $f(y|\boldsymbol{\theta})=f(y|\boldsymbol{\theta}^*)$, for almost every $y\in\mathcal{Y}$, where $\mathcal{Y}$ is the sample space, if and only if $\boldsymbol{\theta}$ and $\boldsymbol{\theta}^*$ are identical. Note that, for a $K-$component mixture model such as the GMM (2.1), there are $K!$ permutations of the labels $\{1,2,\ldots,K\}$ of the component parameters $\boldsymbol{\theta}_1,\ldots,\boldsymbol{\theta}_K$. For any permutation $\phi_t=(\phi(1),\phi(2),\ldots,\phi(K))$, for $t=1,2,\ldots,K!$, $f(y|\boldsymbol{\theta}^{\phi_t})=f(y|\boldsymbol{\theta})$, where $\boldsymbol{\theta}^{\phi_t}=(\boldsymbol{\theta}_{\phi(1)},\ldots,\boldsymbol{\theta}_{\phi(K)})$ is the t^{th} permutation of $\boldsymbol{\theta}$. Thus, in this sense, a $K-$component mixture model is not identifiable. However, as recognised by Frühwirth-Schnatter [2006], this form of non-identifiability is not severe. In practice, one can handle it by imposing order constraints on the parameters to enforce a unique label. Thus, apart from the label permutation invariance, the GMM (2.1), in particular, is identifiable. In this sense, the model is said to be

identifiable up to the relabelling of the components.
See Titterington et al. [1985] and Frühwirth-Schnatter [2006] for more details on identifiability and other forms of non-identifiability for a mixture model, respectively.

2.1.2 Estimation

In order to estimate the model (2.1), we must estimate the unknown model parameters $\boldsymbol{\theta}$. There exists a variety of estimation methods that can be used for this purpose. In his seminal work, Pearson [1894] used a method of moments-based (MM) approach to estimation. Let $\boldsymbol{\mu}(\boldsymbol{\theta})$ denote the vector of the moments of model (2.1) and $\mathbf{m}$ denote the vector of sample moments. The MM estimator, denoted $\hat{\boldsymbol{\theta}}$, is the value of $\boldsymbol{\theta}$ that satisfies $\boldsymbol{\mu}(\hat{\boldsymbol{\theta}}) = \mathbf{m}$. Another class of methods to estimate the parameters $\boldsymbol{\theta}$ are known as minimum-distance (MD) methods. Let $F(y|\boldsymbol{\theta})$ be the distribution function corresponding to (2.1) and $F_n(y)$ be the empirical distribution function based on a random sample of size n. Based on a suitable distance measure $\mathcal{D}\{F(\cdot|\boldsymbol{\theta}), F_n(\cdot)\}$, the MD estimator $\hat{\boldsymbol{\theta}}$ is a value that minimises $\mathcal{D}(\cdot, \cdot)$.
The above methods have become less popular in modern practice. The dominant approaches to estimation in modern mixture modelling can be categorised in two groups, namely, frequentist likelihood-based methods and Bayesian methods. There are two catalysts responsible for the popularity of these methods for mixture modelling. The first is the seminal papers on the Expectation-Maximisation (EM) algorithm (Dempster et al. [1977]) and Monte Carlo Markov Chain (MCMC) methods (Hastings [1970], Geman and Geman [1984] and Gelfand and Smith [1990]). The second catalyst is the increase in computing power necessary to efficiently implement these methods. In this thesis, we will focus on frequentist likelihood methods via the EM algorithm. The reader is referred to the monograph Frühwirth-Schnatter [2006] for a comprehensive coverage of Bayesian analysis of finite GMMs and finite mixture models, in general.

Maximum likelihood estimation and the EM algorithm

The likelihood approach to estimation begins by defining the likelihood function for an observed random sample $\{y_i : i = 1, 2, \ldots, n\}$ from the model (2.1)

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} \sum_{k=1}^{K} \pi_k \mathcal{N}(y_i|\mu_k, \sigma_k^2). \tag{2.3}$$

For mathematical convenience, we usually work with the log-likelihood $\ell(\boldsymbol{\theta}) = \log L(\boldsymbol{\theta})$. The likelihood estimator (MLE) $\hat{\boldsymbol{\theta}}$ is obtained as a solution to the problem

$$\max_{\boldsymbol{\Theta}} \ell(\boldsymbol{\theta}). \tag{2.4}$$

where $\boldsymbol{\Theta}$ is the parameter space.
Note that $\ell(\boldsymbol{\theta})$ is unbounded and has many spurious local maxima (Frühwirth-Schnatter [2006]). Suppose that $\mu_k = y_i$, if we consider the limit $\sigma_k \to 0$, then $\ell(\boldsymbol{\theta}) \to \infty$, it follows that (2.4) is an ill-posed problem. However, we can still find well-behaved local maxima if we take steps to avoid such pathological solutions. This can be achieved by imposing inequality constraints of the form

$$\min_{j \neq k}(\sigma_j / \sigma_k) \geq c > 0. \tag{2.5}$$

Another issue with likelihood-based inference of mixture models is the fact that the problem (2.4) cannot be solved in closed form. However, the problem can be solved numerically using gradient-based methods (see Xu and Jordan [1996]), Newton-type methods (see the various examples in Titterington et al. [1985]) or the Expectation-Maximisation (EM) algorithm (Dempster et al. [1977]). In this thesis, we will make use of the EM algorithm, for more details on the use of the other methods, the reader is referred to the above references.
The EM algorithm for solving the optimisation problem (2.4) proceeds as follows. First, the observed data is viewed as incomplete-data. Next, we introduce a latent variable $z_{ik} = 1$ if the i^{th} data point comes from the k^{th} component and 0 otherwise. Let $\{(y_i, \mathbf{z}_i) : i = 1, 2, \ldots, n\}$, where $\mathbf{z}_i = (z_{i1}, z_{i2}, \ldots, z_{iK})$, be the complete-data. The log-likelihood function based on the complete-data is given by

$$\ell^c(\boldsymbol{\theta}) = \sum_{i=1}^{n} \sum_{k=1}^{K} z_{ik} [\log \pi_k + \log \mathcal{N}(y_i | \mu_k, \sigma_k^2)]. \tag{2.6}$$

Starting from a given initial value $\boldsymbol{\theta}^{(0)}$, the EM algorithm generates a sequence of estimates $\{\boldsymbol{\theta}^{(r)}\}$ by repeatedly iterating between two steps, the *Expectation* (E-step) and the *Maximisation* (M-step), until convergence.
In the E-step, we evaluate $\mathbb{E}\{\ell^c(\boldsymbol{\theta}) | y_i, \mathbf{z}_i, \boldsymbol{\theta}^{(r)}\}$, denoted by $Q(\boldsymbol{\theta} | \boldsymbol{\theta}^{(r)})$,

$$Q(\boldsymbol{\theta} | \boldsymbol{\theta}^{(r)}) = \sum_{i=1}^{n} \sum_{k=1}^{K} \mathbb{E}\{\mathbf{z}_i | y_i, \boldsymbol{\theta}_k^{(r)}\} [\log \pi_k + \log \mathcal{N}(y_i | \mu_k, \sigma_k^2)] \tag{2.7}$$

This reduces to calculating $\mathbb{E}\{\mathbf{z}_i|y_i, \boldsymbol{\theta}_k^{(r)}\} \equiv P(z_{ik} = 1|y_i)$ as

$$p_{ik}^{(r+1)} = \frac{\pi_k^{(r)} \mathcal{N}(y_i|\mu_k^{(r)}, \sigma_k^{2(r)})}{\sum_{\ell=1}^{K} \pi_\ell^{(r)} \mathcal{N}(y_i|\mu_\ell^{(r)}, \sigma_\ell^{2(r)})}, \tag{2.8}$$

for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, using Bayes' theorem.
The posterior probabilities are the responsibilities (Bishop [2006]). For instance, $p_{ik}^{(r+1)}$ is the responsibility that the k^{th} component takes "explaining" the i^{th} data point.
In the subsequent M-step, we obtain $\boldsymbol{\theta} = \boldsymbol{\theta}^{(r+1)}$ as a solution to the problem

$$\max_{\boldsymbol{\Theta}} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}). \tag{2.9}$$

After substituting (2.8) into (2.7), we obtain the solution to (2.9) with respect to π_k, μ_k and σ_k^2 as

$$\pi_k^{(r+1)} = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)}}{n}, \tag{2.10}$$

$$\mu_k^{(r+1)} = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)} y_i}{\sum_{i=1}^{n} p_{ik}^{(r+1)}}, \tag{2.11}$$

$$\sigma_k^{2(r+1)} = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)} (y_i - \mu_k^{(r+1)})^2}{\sum_{i=1}^{n} p_{ik}^{(r+1)}}. \tag{2.12}$$

An important and useful feature of the EM algorithm is the fact that the observed likelihood function $L(\boldsymbol{\theta})$ is non-decreasing at each iteration. That is,

$$L(\boldsymbol{\theta}^{(r+1)}) \geq L(\boldsymbol{\theta}^{(r)}), \tag{2.13}$$

where equality means that the algorithm has reached a stationary point. This may signal that the algorithm has converged. Thus, in practice, a popular way to check whether the EM algorithm has converged is to choose an arbitrarily small positive constant, say ϵ, and stop the algorithm whenever

$$L(\boldsymbol{\theta}^{(r+1)}) - L(\boldsymbol{\theta}^{(r)}) < \epsilon. \tag{2.14}$$

For more details on the EM algorithm for GMMs and mixture models, in general, see chapter 9 of Bishop [2006], chapter 2 of (Fruhwirth-Schnatter et al. [2019]) and the original paper Dempster et al. [1977] for further exposure on the use of the EM algorithm to other areas of statistical application other than mixture modelling.

Many authors have reported problems with the classical EM algorithm when fitting finite mixture models (see section 4.3.2 of Titterington et al. [1985]). This includes, among others, slow convergence, sensitivity to initial values, intractable M-steps. To address these problems, the EM algorithm has been extended in many ways (see McLachlan and Krishnan [1997] for an overview). A very useful extension of the EM algorithm, is the Expectation Conditional Maximisation (ECM) algorithm (Meng and Rubin [1993]. The ECM replaces the M-step of the EM algorithm by a sequence of several, say S, simple M-steps, known as Conditional-Maximisation (CM-) steps. Let $\boldsymbol{\theta} = (\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, \ldots, \boldsymbol{\psi}_S)$ be a partition of $\boldsymbol{\theta}$. At the s^{th} CM-step, we obtain $\boldsymbol{\psi}_s = \boldsymbol{\psi}_s^{(r+1)}$ as a solution to the problem

$$\max_{\boldsymbol{\psi}_s} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}) \tag{2.15}$$

given (conditional on) $\boldsymbol{\psi}_t$, for $t \neq s$, fixed at their previous or current values.
The E- and the S CM-steps are repeated until convergence. Each iteration of the ECM algorithm satisfies the inequality (2.13), see Meng and Rubin [1993] for more details. Thus, the condition (2.14) can be used to check whether the algorithm has converged.

2.1.3 Choosing the number of mixture components

A very important consideration, prior to estimating a finite mixture model, is to specify the number of components, K. Choosing the appropriate value of K remains a difficult problem in practice. For most practical problems, it is impossible to choose K *a priori*. For this reason, data-driven approaches have been developed for obtaining the optimal value of K. Based on frequentist likelihood methods, there are two possible approaches. The first approach relies on a likelihood ratio test of $H_0 : K = K_0$ versus $H_a : K = K_a$, where $K_0 \geq 0$ and $K_a \geq K_0$. Let $\hat{\boldsymbol{\theta}}_0$ and $\hat{\boldsymbol{\theta}}_a$ be the MLEs of $\boldsymbol{\theta}$ under H_0 and H_a, respectively. The likelihood ratio test statistic is

$$T = 2\{\ell(\hat{\boldsymbol{\theta}}_a) - \ell(\hat{\boldsymbol{\theta}}_0)\}. \tag{2.16}$$

Unfortunately, under H_0, the distribution of T is generally unknown and difficult to obtain. However, many proposals have been made to circumvent this difficulty, for more details, see subsection 7.2.1 of Fruhwirth-Schnatter et al. [2019] and the references therein.
The second approach, and most popular approach used to select K, makes use of information criteria obtained by penalising the mixture log-likelihood $\ell(\boldsymbol{\theta})$ as

$$\text{IC}(K) = -2\ell(\hat{\boldsymbol{\theta}}_K) + \lambda \text{df}_K, \tag{2.17}$$

where $\hat{\boldsymbol{\theta}}_K$ is the MLE, λ is a positive constant and $\text{df}_K = dim(\hat{\boldsymbol{\theta}}_K)$ is the number of parameters in the $K-$component mixture model. The second term in (2.17) is a penalty on the complexity of the fitted mixture model.
In practice, IC(K) is calculated for a range of values of $K \in \{1, 2, \ldots, K_{max}\}$, where K is the largest value of K considered. An estimated value of K, denoted by $\hat{K}$, is calculated as

$$\hat{K} = \min_{K \in \{1,2,\ldots,K_{max}\}} \text{IC}(K). \tag{2.18}$$

The most popular IC are the Akaike information criterion (AIC; Akaike [1974]) where $\lambda = 2$ and the Bayesian information criterion (BIC; Schwarz [1978]) where $\lambda = \log(n)$. The BIC is generally found to provide consistent results compared to the AIC, for more details, see subsection 7.2.2 of Fruhwirth-Schnatter et al. [2019].
For a comprehensive discussion on selecting the value of K using frequentist likelihood methods and also Bayesian methods, see chapter 7 of Fruhwirth-Schnatter et al. [2019].

2.1.4 Gaussian mixture of regressions (GMRs)

Suppose that we are interested in studying the dependence of y, as a response variable, on a set of covariates $\mathbf{x} = (x_1, x_2, \ldots, x_p)$. In this case, model (2.1) has the form

$$f(y|\mathbf{X} = \mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(y|m_k(\mathbf{x}), \sigma_k^2), \tag{2.19}$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K)$, with $\boldsymbol{\theta}_k = (\pi_k, m_k(\mathbf{x}), \sigma_k^2)$. Model (2.19) is a finite Gaussian mixture of regressions (GMRs). Associated with each component is a regression model of y on $\mathbf{x}$ each with its own regression function $m_k(\mathbf{x})$, henceforth component regression function (CRF). Each CRF is typically a linear function of the covariates, hence parametric, having the form

$$\begin{aligned} m_k(\mathbf{x}) &= \sum_{j=0}^{p} \beta_{k,j} x_j \\ &= \mathbf{x}^\intercal \boldsymbol{\beta}_k, \end{aligned} \tag{2.20}$$

where $\mathbf{x} = (x_0, x_1, \ldots, x_p)^\intercal$, with $x_0 = 1$ for the intercept, and $\boldsymbol{\beta}_k = (\beta_{k,0}, \beta_{k,1}, \ldots, \beta_{k,p})$ are the regression coefficients associated with the k^{th} component regression model. Thus, we can rewrite the parameter vector of the k^{th} component as $\boldsymbol{\theta}_k = (\pi_k, \boldsymbol{\beta}_k, \sigma_k^2)$.
Given (2.20), model (2.19) is a Gaussian mixture of linear regressions (GMLRs) model. The GMLRs model was first introduced by Quandt [1972] as switching regression models. Consider a population with two sub-populations. It is assumed that "*nature chooses between regimes*

[regression models] *with probabilities* π *and* $1-\pi$". The GMLRs model has received widespread adoption in areas such as economics, marketing, machine learning and medicine, among many other fields. See chapter 8 of Frühwirth-Schnatter [2006] for more details on the theory and application of GMLRs models and mixtures of regressions, in general.

The GMLRs is identifiable up to relabelling of the components provided the covariates $\mathbf{x}$ have a certain level of variability (Hennig [2000]). See section 8.2.2 of Frühwirth-Schnatter [2006] for more details about the identifiability of finite mixture of regressions.

Frequentist likelihood-based estimation of a GMLRs model is exactly the same as with a GMM. Given a random sample $\{(\mathbf{x}_i, y_i) : i = 1, 2, \ldots, n\}$ from the model (2.19), the log-likelihood is given by

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(y_i | \mathbf{x}_i^\intercal \boldsymbol{\beta}_k, \sigma_k^2). \tag{2.21}$$

The EM algorithm to maximise (2.21) proceeds in a similar manner as the EM algorithm used to solve (2.4) with obvious differences with respect to the estimation of the component means. On the E-step, we calculate the posterior probabilities $p_{ik}^{(r+1)}$ using (2.8) with $\mu_k^{(r)}$ replaced with $\mathbf{x}^\intercal \boldsymbol{\beta}_k^{(r)}$. At the M-step, we update π_k using (2.10), $\boldsymbol{\beta}_k$ using

$$\boldsymbol{\beta}_k^{(r+1)} = \left(\sum_{i=1}^{n} p_{ik}^{(r+1)} \mathbf{x}_i \mathbf{x}_i^\intercal \right)^{-1} \left(\sum_{i=1}^{n} p_{ik}^{(r+1)} \mathbf{x}_i y_i \right) \tag{2.22}$$

and we update σ_k^2 using (2.12) with $\mu_k^{(r+1)}$ replaced with $\mathbf{x}^\intercal \boldsymbol{\beta}_k^{(r+1)}$, where the first term on the right side of (2.22) is assumed to be invertible.

The assumption of a constant mixing proportion and constant variance as well as the linearity assumption imposed on the CRFs (1.2) of model (2.19) are quite restrictive. The main reason for the linearity assumption on the CRFs, in particular, is that an additive covariate effect makes for ease of interpretation (Hastie and Tibshirani [1990]). Efforts to relax this assumption, partly or completely while retaining the desirable additive covariate effect, have emerged in the literature. The proposed models assume that some of the covariates are linearly related to the response variable while the rest of the variables can be characterised by non-parametric univariate functions. The general form of this class of models is given by (1.1). Thereafter, we discuss how the procedure can be used to estimate model (1.1).

2.2 Local polynomial likelihood (LPL) estimation

In this section, we give a brief overview of the local polynomial likelihood (LPL) estimation procedure for non-parametric regression modelling. The LPL estimation procedure is a combination of local-likelihood estimation (Hastie and Tibshirani [1987]) and local polynomial regression estimation (Fan and Gijbels [1996]).
Regression modelling is one of the most useful statistical techniques for studying the dependence of a response variable y on one or more covariates $\mathbf{x} = (x_1, x_2, \ldots, x_p) \in \mathbb{R}^p$. For $p = 1$, a simple linear regression has the parametric form (see Gujarati and Porter [2011])

$$y = \beta_0 + \beta_1 x + \epsilon, \tag{2.23}$$

where ϵ is the error term, assumed to have zero mean and constant variance σ^2. To conduct statistical inference about (2.23), ϵ is assumed to follow a Gaussian distribution, that is $\epsilon \sim \mathcal{N}(0, \sigma^2)$.
If the linearity assumption is appropriate for a given regression problem, then the regression function $m(x) = \mathbb{E}[Y|X = x]$ has the simple linear expression

$$m(x) = \beta_0 + \beta_1 x. \tag{2.24}$$

Unfortunately, this assumption is not satisfied in most practical applications and the use of model (2.23) will result in large modelling biases (Fan and Gijbels [1996]). There are several ways to avoid the pitfalls of parametric regression modelling (see chapter 1 of Fan and Gijbels [1996] for more details). Each approach has its strengths and weaknesses in different domains of application. In this thesis, we focus on the local polynomial regression approach (Fan and Gijbels [1996]). For ease of exposition, we first consider the case when there is only one covariate ($p = 1$) and later we will extend the discussion to accommodate more covariates ($p > 1$).

2.2.1 Local polynomial likelihood (LPL) estimator for a single covariate

In local polynomial regression, the regression function $m(x)$ is estimated locally in a neighborhood around a local grid point u, in the domain of x, using a polynomial regression model. In other words, locally, we assume that $m(\cdot)$ can be approximated by a polynomial function of some degree p. However, globally, we make no assumptions about the form of $m(\cdot)$. This implies that model (2.23) can be expressed as

$$Y = m(x) + \epsilon, \tag{2.25}$$

where the regression function $m(x) = \mathbb{E}[Y|X = x]$ is an unknown function assumed only to be a smooth function of x.
In this section, we give a brief overview of the local polynomial regression approach. To simplify the understanding, we consider first the case of a single covariate. Later in this chapter we extend the method to multiple covariates.
By assuming, as above, that the error term has a Gaussian distribution, it follows that the conditional distribution of the response variable y, given $X = x$, is Gaussian with mean $m(x)$ and variance σ^2, that is $y|X = x \sim \mathcal{N}(y|m(x), \sigma^2)$. Consider, first, the polynomial linear regression model. To estimate this model, the global log-likelihood function

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \log \mathcal{N}(y_i|m(x_i), \sigma^2) \tag{2.26}$$

is maximised with respect to $\boldsymbol{\beta}$, where $\boldsymbol{\beta} = (\beta_0, \beta_1, \ldots, \beta_p)^\intercal$. In local polynomial regression, a local-likelihood function is maximised locally in a neighborhood around a grid point u to fit a polynomial regression model. Towards that end, suppose that at each grid point $u \in \mathcal{U}$, where $\mathcal{U}$ is the set of grid points, the $(p+1)^{th}$ derivative of $m(x)$ exists. By Taylor expansion, the unknown regression function $m(x)$ can be approximated locally using a p^{th} degree polynomial function

$$\begin{aligned} m(x) &\approx m(u) + m^{(1)}(u)(x-u) + m^{(2)}(u)\frac{(x-u)^2}{2!} + \cdots + m^{(p)}(u)\frac{(x-u)^p}{p!} \\ &= \sum_{j=0}^{p} \beta_j(u)(x-u)^j \end{aligned} \tag{2.27}$$

of x in the neighborhood of u, where $m^{(r)}(u)$ denotes the r^{th} derivative of $m(u)$ at grid point u and $\beta_j(u) = \frac{m^{(j)}(u)}{j!}$.
Let $\boldsymbol{\beta}(u) = (\beta_0(u), \beta_1(u), \ldots, \beta_p(u))$ be a vector of the coefficients (or local parameters) of the local polynomial function at grid point u. The estimate of $\boldsymbol{\beta}(u)$, denoted by $\hat{\boldsymbol{\beta}}(u)$, is obtained by maximising the local polynomial log-likelihood function

$$\ell[\boldsymbol{\beta}(u)] = \sum_{i=1}^{n} \log[\mathcal{N}(y_i|\sum_{j=0}^{p} \beta_j(u)(x_i-u)^j, \sigma^2)] \times K_h(x_i - u), \tag{2.28}$$

where $K_h(\cdot) = K(\cdot/h)/h$ is a rescaled kernel function $K(\cdot)$ used to assign weights to the data points in the neighborhood of a given grid point and $h > 0$ is the bandwidth or smoothing parameter. The bandwidth is used to specify the size of the local neighborhood. We will discuss kernel functions and bandwidths later in this section.

The estimate of $m(u)$ can be easily seen from the Taylor expansion (2.27) to be

$$\hat{m}(u) = \hat{\beta}_0. \tag{2.29}$$

We refer to (2.29) as the local polynomial likelihood (LPL) estimator. Since the function $m(\cdot)$ is assumed to be a smooth. The LPL estimator is referred to as a smoother (Buja et al. [1989]). The entire function $\hat{m}(\cdot)$ can be obtained by maximising (2.28) over all grid points $u \in \mathcal{U}$, where $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$ is the set of points in the domain of the covariate x. For any $x_i \notin \mathcal{U}$, $\hat{m}(x_i)$ can be obtained by interpolation.
A simple and very useful expression of $\hat{m}(u)$ can be specified using matrix notation. Let

$$\mathbf{X} = \begin{bmatrix} 1 & (x_1 - u) & \ldots & (x_1 - u)^p \\ \vdots & \vdots & \ddots & \vdots \\ 1 & (x_n - u) & \ldots & (x_n - u)^p \end{bmatrix} \tag{2.30}$$

be the design matrix and

$$\mathbf{W} = diag\{K_h(x_1 - u), \ldots, K_h(x_n - u)\} \tag{2.31}$$

the weight matrix at grid point u. Then the local polynomial log-likelihood function (2.28) can be expressed as

$$-\frac{1}{2\sigma^2}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}(u))\mathbf{W}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}(u)) + \text{const}, \tag{2.32}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)^\intercal$, $\boldsymbol{\beta}(u) = (\beta_0, \beta_1, \ldots, \beta_p)^\intercal$ and const denotes a term that is independent of $\boldsymbol{\beta}(u)$. Thus, the estimator of $\boldsymbol{\beta}(u)$ can be expressed as

$$\hat{\boldsymbol{\beta}}(u) = (\mathbf{X}^\intercal\mathbf{W}\mathbf{X})^{-1}\mathbf{X}^\intercal\mathbf{W}\mathbf{y}. \tag{2.33}$$

It follows that the LPL estimator (2.29) can be written as

$$\hat{m}(u) = \mathbf{e}^\intercal\hat{\boldsymbol{\beta}}(u), \tag{2.34}$$

where $\mathbf{e} = (1, 0, 0, \ldots, 0)^\intercal$ is a $(p+1)-$dimensional vector whose 1^{st} entry is equal to 1 and the other entries are equal to zero.
For $p = 0$ and $p = 1$, respectively, the LPL estimator can be simply expressed as

$$\hat{m}(u) = \frac{\sum_{i=1}^{n} K_h(x_i - u)y_i}{\sum_{i=1}^{n} K_h(x_i - u)} \tag{2.35}$$

and

$$\hat{m}(u) = \frac{\sum_{i=1}[s_2(u) - s_1(u)(x_i - u)]K_h(x_i - u)y_i}{s_2(u)s_0(u) - s_1^2(u)}, \tag{2.36}$$

where $s_r(u) = \sum_{i=1}^{n} K_h(x_i - u)(x_i - u)^r$, for $r = 0, 1, 2$.

The LPL estimator (2.35) is the well-known Nadaraya-Watson kernel estimator (Nadaraya [1964] and Watson [1964]) or local-constant estimator (LCE) (Fan and Gijbels [1996]). The LCE is usually preferred because of its simplicity of expression, computational expediency and mathematical tractability. However, it has well-known limitations such as biases at the boundaries and regions with high peaks in the true function as well as lack of adaptation to non-uniform covariates.

The LPL estimator (2.36) is the local-linear estimator (LLE) (Fan and Gijbels [1996]). The LLE overcomes the limitations of the LCE. It adapts to different forms of the covariates (random, non-uniform or clustered), it automatically corrects boundary bias and theoretically it has a small bias and variance at interior grid points. For more details on the properties of both the LCE and LLE, see Fan [1992] and Fan and Gijbels [1996] and the references therein. Fan and Gijbels [1996] recommend for the use of the LLE for most practical problems.

Let $\hat{\mathbf{y}} = (\hat{y}_1, \ldots, \hat{y}_n)^\intercal$ be the vector of fitted values, where $\hat{y}_i = \hat{m}(x_i)$. This can be expressed as

$$\hat{\mathbf{y}} = \mathbf{S}_h \mathbf{y}, \tag{2.37}$$

where

$$\mathbf{S}_h = (\mathbf{s}^\intercal(x_1), \mathbf{s}^\intercal(x_2), \ldots, \mathbf{s}^\intercal(x_n))^\intercal, \tag{2.38}$$

with

$$\mathbf{s}(x_i) = \mathbf{e}^\intercal \times (\mathbf{X}^\intercal \mathbf{W} \mathbf{X})^{-1} \mathbf{X}^\intercal \mathbf{W} \qquad \text{for } i = 1, 2, \ldots, n. \tag{2.39}$$

The matrix $\mathbf{S}_h$ is known as a smoother matrix (Buja et al. [1989]), where the subscript is included to indicate the dependence of the matrix on the bandwidth. To see this, note that $\mathbf{W}$ is a function of $K_h(\cdot)$, see (2.31), and the latter is a function of the bandwidth h. Therefore, $\mathbf{S}_h$, through (2.39), is a function of h. Furthermore, note that $\mathbf{S}_h$ does not depend on y. For this reason, the LPL estimator (2.29) is referred to as a linear smoother. Linear smoothers have useful properties one of which we discuss next and and will make use of later in this section. For more details on linear smoothers, see the discussion in Buja et al. [1989].

Given a fitted regression function $\hat{m}(\cdot)$, we might be interested in the number of parameters

that were used to obtain this estimate. This is useful for quantifying the complexity of the fitted model. In analogy with parametric linear regression models, this quantity is the degrees of freedom (df). The df can be obtained by taking the sum of the diagonal elements of the smoother matrix

$$\text{df} = \text{tr}(\mathbf{S}_h) = \sum_{i=1}^{n} s_{ii}, \tag{2.40}$$

where the s'_{ii}s are the diagonal entries of $\mathbf{S}_h$ and $\text{tr}(\mathbf{A})$ denotes the trace of matrix $\mathbf{A}$. The degrees of freedom of an LPL smoother is generally not an integer. Thus, it is usually referred to as the effective (or equivalent) degrees of freedom (edf). For more details on the concept of the effective degrees of freedom, see Buja et al. [1989] and the discussion in section 3.3 of Green and Silverman [1994].

2.2.2 Components of local regression

In order to be useful in practice, the LPL estimation procedure depends on the appropriate choice of three components: the bandwidth h, the degree of the local polynomial function p and the kernel function $K(\cdot)$. These components must be specified, either subjectively or objectively, before the procedure is implemented.

Smoothing parameter or bandwidth, h

As already mentioned in subsection 2.2.1, the bandwidth h specifies the size of the local neighborhood around a given grid point u. This implies that h controls the smoothness of the LPL estimate (Fan and Gijbels [1996]). Thus, the choice of h must be given careful consideration. Ideally, the optimal bandwidth can be chosen objectively by making use of the statistical properties of the LPL estimator (see subsection 3.2.4 of Wu and Zhang [2006]). However, in practice, a data-driven approach, such as the cross-validation (CV), is used for bandwidth or, in general, smoothing parameter selection (Hastie et al. [2009]).
The CV approach proceeds by randomly splitting the observed data into J partitions each of size n_j, for $j = 1, 2, \ldots, J$. Fit the model to the data in the $J-1$ partitions and calculate the prediction error of the fitted model on the data in the remaining partition. Denote by $\hat{m}_h^{(-j)}(x)$, for $j = 1, 2, \ldots, J$, the fitted model based on all the data except the data in the j^{th} partition. Repeat this process for all the J partitions to get the CV error

$$\text{CV}(h) = \sum_{j=1}^{J} \sum_{i=1}^{n_j} (y_i - \hat{m}_h^{(-j)}(x_i))^2. \tag{2.41}$$

The CV approach selects h to minimise CV(h) (5.63) over a carefully chosen grid of values for h. For more details on the CV approach to smoothing parameter selection, see Hastie et al. [2009].

For $J = n$, a very useful approximation to the CV error (2.41) can be obtained by first noting that, for most linear smoothers,

$$\sum_{j=1}^{J}\sum_{i=1}^{n_j}(y_i - \hat{m}_h^{(-j)}(x_i))^2 \approx \frac{1}{n}\sum_{i=1}^{n}\left(\frac{y_i - \hat{m}_h(x_i)}{1 - s_{ii}}\right)^2, \tag{2.42}$$

where $\hat{m}_h(x_i)$, for $i = 1, 2, \ldots, n$, is the fitted value of y_i at x_i obtained from the LPL estimate based on all the observations.

This approximation can be found by replacing each diagonal entry of $\mathbf{S}_h$, s_{ii} for $i = 1, 2, \ldots, n$, by df_h/n, resulting in the generalised cross-validation (GCV) error

$$\begin{aligned} \mathrm{GCV}(h) &= \frac{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{(1 - \mathrm{df}_h/n)^2} \\ &= \frac{\mathrm{MSE}}{(1 - \mathrm{df}_h/n)^2}, \end{aligned} \tag{2.43}$$

where $\hat{y}_i = \hat{m}_h(x_i)$ and $\mathrm{MSE}=\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$ and the subscript is included to explicitly indicate that df is a function of h. The GCV approach selects the smoothing parameter that minimises the GCV error (2.43). The GCV approach was first introduced by Wahba [1977] and then subsequently studied by Craven and Wahba [1979]. For more details on the theoretical and practical aspects of the GCV criterion see the above-mentioned references. It is important to mention that, recently, Patil et al. [2024] showed that the GCV error is inconsistent in a parametric high-dimensional setting such as (2.23) with $p \to \infty$. Since the non-parametric functions we consider in this thesis are univariate functions, that is $p = 1$, this should not be a concern in our case. However, it remains to be proven whether their results also apply in a non-parametric low-dimensional setting, such as is the case in this thesis. This is an interesting topic for future research.

Another popular class of data-driven smoothing parameter selection methods is the class of information criteria (IC). As discussed in subsection 2.1.3, an IC is defined as

$$\mathrm{IC}(h) = -2\hat{\ell} + \lambda \times \mathrm{df}_h, \tag{2.44}$$

where $\hat{\ell}$ is the maximum value of the likelihood function. As with the cross-validation approach, an appropriate value of the smoothing parameter is chosen to minimise the information criterion (2.44). The AIC(h) or BIC(h) can be used for this purpose. For more details on the use of

information criteria for smoothing parameter selection, see chapter 7 of Hastie et al. [2009].

Local polynomial degree, p

Another component of local polynomial fitting that affects the bias and variance trade-off of the fitted model is the degree of the local polynomial function. This component plays a much more crucial role in reducing the bias at local regions with higher order effects, such as curvature, in the true regression function (Hastie and Loader [1993]). However, the choice of p is not as crucial as the choice of the bandwidth. Usually small values of p, such as $p = 1$, are preferred and one can just concentrate on choosing an appropriate value for the bandwidth. For more details, see chapter 3 of Fan and Gijbels [1996] and chapter 2 of Loader [1999].

Kernel function, $K_h(\cdot)$

Finally, LPL estimation depends on a local weighting function known as a kernel function, $K_h(\cdot)$. The kernel function is usually chosen to be a continuous function that is symmetric around 0. The most widely used kernel function is the Gaussian kernel based on the standard Gaussian density function

$$K_h(t) = \frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}t^2\right). \tag{2.45}$$

Other commonly used kernel functions are members of the symmetric Beta family

$$K_h(t) = \frac{1}{Beta(1/2, \gamma+1)}(1-t^2)_+^\gamma, \qquad \gamma = 0, 1, \ldots, \tag{2.46}$$

where $x_+^\gamma = [\max(0, x)]^\gamma$ and $Beta(\alpha, \beta)$ denotes the beta function with parameters α and β. The choice of $\gamma = 0, 1, 2$ and 3 yields the uniform kernel, Epanechnikov kernel, biweight kernel and triweight kernel, respectively. The Gaussian kernel (2.45) is also a member of the family in the limit $\gamma \to \infty$.
In practice, the choice of the kernel function is not as important as the bandwidth or local polynomial degree because it has less influence on the bias and variance of the LPL estimator (Loader [1999]). Nevertheless, Fan and Gijbels [1996] recommend the use of the Epanechnikov as a universal optimal kernel function.

2.2.3 LPL estimator for multiple covariates

In many real world problems the behaviour of a response variable cannot be adequately explained by a single covariate and more covariates have to be considered. In this section, we

extend the univariate local polynomial regression procedure to accommodate multivariate covariates.

Consider a random sample $\{(\mathbf{x}_i^\intercal, y_i) : i = 1, 2, \ldots, n\}$ with $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iD})^\intercal$, from the population $(\mathbf{X}^\intercal, Y) \in \mathbb{R}^D \times \mathbb{R}$. Suppose that it is of interest to estimate the regression function $m(\mathbf{x}) = \mathbb{E}[Y|\mathbf{X} = \mathbf{x}]$, a $D-$variate smooth unknown function. As before, the above random sample can be assumed to have been generated by the following model

$$Y = m(\mathbf{X}) + \epsilon, \tag{2.47}$$

where ϵ is the error term assumed to have a Gaussian distribution with mean zero and variance σ^2, that is $\epsilon \sim \mathcal{N}(0, \sigma^2)$.

By Taylor expansion, $m(\mathbf{x})$ can be locally approximated by a polynomial of degree p in the neighborhood of $\mathbf{u} = (u_1, u_2, \ldots, u_D)^\intercal$ as

$$\begin{aligned} m(\mathbf{x}) &\approx \beta_0 + \sum_{j=1}^{D} \beta_{1j}(x_j - u_j) + \cdots + \sum_{j=1}^{D} \beta_{pj}(x_j - u_j)^p \\ &= \beta_0 + \sum_{k=1}^{p} \sum_{j=1}^{D} \beta_{kj}(x_j - u_j)^k, \end{aligned} \tag{2.48}$$

where

$$\beta_0 = m(\mathbf{u}) \quad \text{and} \quad \beta_{kj} = \frac{1}{k!} \frac{\partial^k m(\mathbf{u})}{\partial x_j^k}, \tag{2.49}$$

for $k = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, D$.

Let $\boldsymbol{\beta}(\mathbf{u}) = (\beta_0, \boldsymbol{\beta}_1, \boldsymbol{\beta}_2, \ldots, \boldsymbol{\beta}_p)$, where $\boldsymbol{\beta}_k = (\beta_{k1}, \beta_{k2}, \ldots, \beta_{kD})$, be a vector of the coefficients of the polynomial function (or local parameters) at the local point $\mathbf{u}$. Recall that in local-likelihood estimation, the likelihood function is maximised around a neighborhood of a given local point. Thus, at the local point $\mathbf{u}$, the estimate of $\boldsymbol{\beta}(\mathbf{u})$, denoted $\hat{\boldsymbol{\beta}}(\mathbf{u})$, is obtained by maximising the following local polynomial log-likelihood function as

$$\ell\{\boldsymbol{\beta}(\mathbf{u})\} = \sum_{i=1}^{n} \log \mathcal{N}(y_i | \beta_0 + \sum_{k=1}^{p} \sum_{j=1}^{D} \beta_{kj}(x_{ij} - u_j)^k, \sigma^2) K_{\mathbf{H}}(\mathbf{x}_i - \mathbf{u}), \tag{2.50}$$

where $K_{\mathbf{H}}(\mathbf{u}) = \frac{1}{|\mathbf{H}|} K(\mathbf{H}^{-1}\mathbf{u})$ is a rescaled $D-$variate kernel function with $\mathbf{H}$ a non-singular $D \times D$ bandwidth matrix. The expression $|\mathbf{H}|$ denotes the determinant of the matrix $\mathbf{H}$ (For more details on the multivariate kernel function, see chapter 7 of Fan and Gijbels [1996]).

From (2.49), the multivariate local polynomial regression (LPR) estimator is given by

$$\hat{m}(\mathbf{u}) = \hat{\beta}_0. \tag{2.51}$$

Let $\mathcal{U} = \{\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_N\}$ be the set of local points. To recover the whole regression function, the local-likelihood function (2.50) is maximised over all N local points. To obtain $\hat{m}(\mathbf{x}_i)$ for $\mathbf{x}_i \notin \mathcal{U}$, we use linear interpolation.

2.2.4 Curse of dimensionality

Although the above generalisation of the local polynomial regression procedure is quite straightforward, there is a serious problem that arises when working with high-dimensional data locally. This problem has to do with how to specify local neighborhoods. Consider a neighborhood with a fixed number of data points m. For a $D-$dimensional covariate, each of the D axis will have m data points associated with them. Thus, there are m^D data points in the corresponding $D-$dimensional neighborhood. For $m = 5$ and $D = 2$, we have 25 data points. For $m = 5$ and $D = 10$, m^D is almost 10 million data points. Suppose that our sample size is, say 10 million, this implies that the local neighborhood is made up of about 98% of the data points, this is no longer local. Unfortunately, the problem cannot be solved by dramatically reducing the size of the local neighborhood m, since the less observations we use for local fitting the higher the variance of the local estimate. Another consequence of local fitting in high dimension is sparsity. If the local neighborhood size is not fixed, which is common in practice, it is possible to have empty local neighborhoods (Bishop [2006]).
Consequently, much larger data sets are needed in order: (1) for the neighborhood to be "local", (2) to obtain variance-stable local estimates (Hastie and Tibshirani [1990]) and (3) to ensure that there are no empty local neighborhoods. This problem has been aptly termed the "curse-of-dimensionality" (Bellman [1961]). The dimensionality problem arises in various forms and has far greater consequences. For more details on the curse of dimensionality and its practical and theoretical consequences, see Hastie et al. [2009] and Scott [1992], respectively.

2.2.5 Extensions

The above discussion on the curse of dimensionality highlights the challenges of working with high-dimensional (multivariate) covariates when fitting a non-parametric regression model. We now discuss some of the proposed methods to address this problem. These methods are all geared towards reducing the dimension of the non-parametric functions to a one-dimensional space and non-parametric fitting proceeds as in section 2.2.1. Thus, they involve a loss of flexibility however the gain is practical usefulness of the fitted non-parametric regression model.

Additive regression modelling

We begin our discussion with additive regression models. These models are a generalisation of the traditional multiple linear regression model. At the beginning of this section, we assumed that the regression function $m(\mathbf{x})$ is a linear function and hence additive in the covariates. This implies that the covariate effects can be interpreted separately. This is an important feature whenever interest lies in quantifying the individual effect of each covariate on the response. Additive regression models retain this important feature while relaxing the linearity assumption. These models have the form

$$Y = \alpha + \sum_{d=1}^{D} g_d(x_d) + \epsilon, \tag{2.52}$$

where, as before, ϵ represents the error term assumed to follow a Gaussian distribution, that is $\epsilon \sim \mathcal{N}(0, \sigma^2)$, and also independent of the covariates $x_1, x_2, \ldots, x_D$. The $g_d(\cdot)'s$ are unknown univariate functions assumed to be smooth. To ensure that model (2.52) is identifiable, the functions $g_d(\cdot)'s$ must satisfy

$$\mathbb{E}\{g_d(x_d)\} = 0 \qquad \text{for } d = 1, 2, \ldots, D. \tag{2.53}$$

Consequently, $\mathbb{E}(Y) = \alpha$.

Additive models were first introduced and studied by Stone [1985] for a continuous response variable y. Hastie and Tibshirani [1987] and Hastie and Tibshirani [1990] extended the models to a generalised modelling framework to accommodate non-continuous (and/or non-Gaussian) response variables. This gave rise to the generalised additive models (GAMs).

Various methods have been proposed to estimate the additive regression model (2.52). This includes the classical backfitting algorithm of Hastie and Tibshirani [1990] and its modified version, the marginal integration estimation procedure of Linton and Nielsen [1995] and the smoothed backfitting algorithm of Mammen et al. [1999]. These procedures can be referred to as one-stage estimation procedures. See section 8.1 of Härdle et al. [2004] for an excellent review of these methods.

To improve the performance of the above one-stage estimation procedures, a class of so-called two-stage estimation procedures were proposed. In the first-stage, preliminary or pilot estimators of the additive non-parametric functions are constructed. In the second-stage, the first-stage estimators are used to initialise the classical backfitting algorithm. The first-stage estimators of Linton [1997] and Horowitz et al. [2006] were constructed using the marginal integration procedure and the smoothed backfitting procedure, respectively. To improve the computational speed, Wang and Yang [2007] proposed to use spline estimators in the first-

stage. We will provide more details on the implementation of this two-stage procedure, also known as the spline-backfitting procedure. Before we proceed, we briefly introduce the classical backfitting algorithm because it is an intermediate part of the spline-backfitting procedure.

Backfitting algorithm Note that if the additive model specification (2.52) is adequate for the underlying regression problem, then

$$\mathbb{E}\{Y - \alpha - \sum_{d \neq h} g_d(X_d)|X_h\} = g_h(X_h). \tag{2.54}$$

This suggests the following iterative algorithm for estimating all the univariate functions $g_d : d = 1, 2, \ldots, D$. For given α and the univariate functions $g_d : d \neq h$, obtained at the previous iteration, g_h can be estimated using the LPL estimator (2.29).
For some $h \in \{1, 2, \ldots, D\}$, we can approximate the unknown function $g_h(\cdot)$ locally using a polynomial function of degree p as

$$\begin{aligned} g_h(x_h) &\approx g_h(u) + g_h^{(1)}(u)(x_h - u) + \cdots + g_h^{(p)}(u)(x_h - u)^p/p! \\ &\equiv \beta_{h0} + \beta_{h1}(x_h - u) + \cdots + \beta_{hp}(x_h - u)^p, \end{aligned} \tag{2.55}$$

for x_h in the neighborhood of $u \in \mathcal{U}$, where $\mathcal{U}$ is the set of local grid points in the domain of x_h and $\beta_{hj} = g_h^{(j)}(u)$ for $j = 0, 1, \ldots, p$.
Consider a random sample of pairs of data $\{(\mathbf{x}_i, y_i) : i = 1, 2, \ldots, n\}$, where $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iD})$, from the model (2.52). The conditional distribution of y, given $\mathbf{X} = \mathbf{x}$, is Gaussian with mean $m(\mathbf{x})$ and variance σ^2.
Let $\boldsymbol{\beta}(u) = (\beta_{k0}, \beta_{k1}, \ldots, \beta_{kp})$ be the vector of all local parameters at local point u. Let $\hat{g}_d(x_d)$, for $d \neq h$, be the current estimates or initial estimates of the additive functions $g_d(x_d)$, for $d \neq h$, we estimate $g_h(x_h)$ by first maximising the following local polynomial log-likelihood function

$$\ell\{\boldsymbol{\beta}(u)\} = \sum_{i=1}^{n} \log \mathcal{N}(y_i | \sum_{j=0}^{p} \beta_{hj}(x_{ih} - u)^j + \sum_{d \neq h} \hat{g}_d(x_{id}), \sigma^2) K_h(x_{ih} - u). \tag{2.56}$$

Then we estimate $g_h(u)$ as $\hat{g}_h(u) = \hat{\beta}_{h0}$ for $u \in \mathcal{U}$. In order to meet condition (2.53), $\hat{g}_h(\cdot)$ can be expressed as follows

$$\hat{g}_h^*(u) = \hat{g}_h(u) - \frac{1}{N} \sum_{u \in \mathcal{U}} \hat{g}_h(u). \tag{2.57}$$

Using interpolation, we can obtain $\hat{g}_h^*(x_{ih})$ for $i = 1, 2, \ldots, n$. The above procedure is repeated to obtain the other univariate functions. After cycling through all the functions $d = 1, 2, \ldots, D$ once, the procedure is repeated until apparent convergence. Condition (2.53) suggests that, at each cycle, α can be estimated as $\frac{1}{n}\sum_{i=1}^{n} y_i$. This procedure is known as the backfitting algorithm. This idea of fitting univariate functions using partial residuals was first used by Friedman and Stuetzle [1981] in the their projection pursuit procedure. The backfitting algorithm was formally introduced by Breiman and Friedman [1985] as an intermediate part of the alternating conditional expectation algorithm (Hastie and Tibshirani [1987]). The algorithm is studied by Buja et al. [1989] and Hastie and Tibshirani [1990]. For a good summary of the backfitting algorithm, see chapter 7 of Fan and Gijbels [1996] or section 8.1 of Härdle et al. [2004].

Spline-backfitting local polynomial likelihood estimation According to Ma and Yang [2014], for an estimator of an additive non-parametric function to be considered satisfactory, it should be, among other things, computationally expedient and theoretically reliable. LPL estimators are theoretically attractive, however, for large n, they are computationally intensive. On the other hand, spline estimators are fast to compute but lack theoretical reliability. By combining the best of both spline estimators (computational expediency) and LPL estimators (theoretical reliability), Wang and Yang [2007] and Wang and Yang [2009] proposed spline-backfitted kernel (SBK) estimators and spline-backfitted local-linear (SBLL) estimators, respectively. These procedures are special cases of the general spline-backfitted local polynomial likelihood (SBLPL) procedure with $p = 0$ and $p = 1$, respectively. The SBLPL estimation procedure is a two-stage estimation procedure. In the first-stage, spline estimators, such as cubic splines, are used to obtain initial or pilot estimators of the constant and the additive functions α and $g_d(x_d) : d = 1, 2, \ldots, D$, respectively. In the second-stage, using the first-stage estimators $\hat{\alpha}$ and $\hat{g}_d(x_d) : d = 1, 2, \ldots, D$, we construct a pseudo response variable $\hat{y}_h = y - \hat{\alpha} - \sum_{d \neq h}^{D} \hat{g}_d(x_d)$ for the covariate x_h, for $h \neq d$, and use the LPL estimator to estimate $g_h(x_h)$ based on the data $\{(x_{ih}, \hat{y}_{ih} : i = 1, 2, \ldots, n\}$.
The underlying idea of the SBLPL procedure is that, if all the $d \neq h$ additive functions were provided by some "oracle", we could define the response variable $\hat{y}_j = g_h(x_h) + \epsilon$ corresponding to the covariate x_h. Thus, model (2.52) reduces to model (2.25) and the usual LPL estimation procedure can be used to estimate $g_h(x_h)$ based on (x_h, y_h).
In the first-stage, Wang and Yang [2009] used first-order B-spline basis functions to approximate the additive functions. In the second-stage, they used local-constant estimators (LCEs) and local-linear estimators (LLEs). In the following brief outline of the SBLPL estimation procedure, we use the general $k^{th}-$order B-spline basis functions to approximate the additive func-

tions in the first-stage and the general LPL estimators to estimate the additive non-parametric functions in the second-stage.

Assume that each covariate $x_d, d = 1, 2, \ldots, D$ is distributed on a compact interval $[a_d, b_d]$, for $d = 1, 2, \ldots, D$, and, without loss of generality, we assume that $[a_d, b_d] = [0, 1]$, for $d = 1, 2, \ldots, D$. Furthermore, we preselect an integer $N_n = \lceil n^{1/3} \log(n) \rceil$. Next, we define an ordered sequence of equally spaced points referred to as knots $\{\xi_r : r = 0, 1, \ldots, N_n + 1\}$. The knots $\{\xi_r : r = 1, 2, \ldots, N_n\}$ and $\{\xi_0, \xi_{N_n+1}\}$ are referred to as the internal and boundary knots, respectively. We also introduce $2(k-1)$ degenerate knots $\xi_{-(k-1)} = \xi_{-(k-1)+1} = \cdots = \xi_0$ and $\xi_{N_n+k} = \xi_{N_n+k-1} = \cdots = \xi_{N_n+1}$. The distance between any two neighbouring knots $\{\xi_{J-1}, \xi_J\}$ is fixed at $H = (N_n + 1)^{-1}$. For each covariate x_d, for $d = 1, 2, \ldots, D$, we can use the recursive relation on page 90 of De Boor [1978] to define the $k^{th}-$order B-spline basis functions, denoted b_{Jk}, for the $N_n + 1$ equally-spaced sub-intervals on the finite interval $[0, 1]$. Thus, for $J = 0, 1, \ldots, N_n + 1$ and $k > 1$

$$b_{Jk}(x) = \eta_{Jk}(x) b_{J,k-1}(x) + (1 - \eta_{J+1,k}) b_{J+1,k-1}, \tag{2.58}$$

where $b_{Jk}(x) > 0$ if $x \in \{\xi_{J-(k-1)}, \xi_{J+k}\}$ and 0 otherwise,

$$\eta_{Jk}(x) = \frac{x - \xi_J}{\xi_{J+k-1} + \xi_J} \tag{2.59}$$

and

$$b_{J1}(x) = \begin{cases} 1 & \text{for } \xi_J \leq x \leq \xi_{J+1} \\ 0 & \text{otherwise} \end{cases} \tag{2.60}$$

To obtain the explicit expressions of the $k^{th}-$order B-spline basis functions $b_{Jk}(x)$, we substitute the expressions of the $(k-1)^{th}$ order B-spline basis functions into (2.58).

For mathematical convenience, let $B_{Jk}(x)$ denote the centered and standardised version of the $k^{th}-$order B-spline basis piecewise polynomial (pp) functions (Wang and Yang [2007]). Next, we define the additive spline functions as

$$g(\mathbf{x}) = \sum_{d=1}^{D} g_d(x_d) = \omega_0 + \sum_{J=1}^{N_n+k-1} \sum_{d=1}^{D} \omega_{J,d} B_{Jk}(x_d). \tag{2.61}$$

Recall that the conditional distribution of Y given $\mathbf{X} = \mathbf{x}$ in (2.52) is assumed to be normally distributed. Given a random sample $\{(x_{id}, y_i) : i = 1, 2, \ldots, n; d = 1, 2, \ldots, D\}$ from model

(2.52), the log-likelihood function, in terms of the B-spline basis functions, is given by

$$\begin{aligned}\ell\{\boldsymbol{\omega}\} &= \sum_{i=1}^{n} \log\mathcal{N}(y_i|\omega_0 + \sum_{J=1}^{N_n+k-1}\sum_{d=1}^{D}\omega_{J,d}B_{Jk}(x_{id}),\sigma^2) \\ &\propto -\sum_{i=1}^{n}[y_i - \omega_0 - \sum_{J=1}^{N_n+k-1}\sum_{d=1}^{D}\omega_{J,d}B_{Jk}(x_{id})]^2, \end{aligned} \tag{2.62}$$

where $\boldsymbol{\omega} = (\omega_0, \omega_{1,1}, \ldots, \omega_{N_n+k-1,D})$. The spline estimator of $g(\mathbf{x})$ is given by

$$\hat{g}(\mathbf{x}) = \hat{\omega}_0 + \sum_{J=1}^{N_n+k-1}\sum_{d=1}^{D}\hat{\omega}_{J,d}B_{Jk}(x_d), \tag{2.63}$$

where $\hat{\boldsymbol{\omega}} = (\hat{\omega}_0, \hat{\omega}_{1,1}, \ldots, \hat{\omega}_{N_n+k-1,D})$ are the maximum likelihood estimators of $\boldsymbol{\omega}$ defined as

$$\hat{\boldsymbol{\omega}} = \max\ell\{\boldsymbol{\omega}\}. \tag{2.64}$$

Denote α in (2.52) as g_α, the first-stage estimators of the additive functions $g_d(x_d)$, for $d = 1, 2, \ldots, D$ and g_α, respectively, are given by

$$\begin{aligned}\hat{g}_d(x_d) &= \sum_{J=1}^{N_n+k-1}\hat{\omega}_{J,d}B_{Jk}(x_d) - \frac{1}{n}\sum_{i=1}^{n}\sum_{J=1}^{N_n+k-1}\hat{\omega}_{J,d}B_{Jk}(x_{id}) \\ \hat{g}_\alpha &= \hat{\omega}_0 + \frac{1}{n}\sum_{d=1}^{D}\sum_{i=1}^{n}\sum_{J=1}^{N_n+k-1}\hat{\omega}_{J,d}B_{Jk}(x_{id}). \end{aligned} \tag{2.65}$$

In the second-stage, we use the first-stage estimators to construct the pseudo response $\hat{y}_d$ for y_h, for $h \neq d$

$$\hat{y}_h = y - \hat{g}_\alpha - \sum_{d\neq h}\hat{g}_d(x_d). \tag{2.66}$$

Typically, g_α is estimated by $\hat{g}_\alpha = \frac{1}{n}\sum_{i=1}^{n} y_i$ which is the $\sqrt{n}-$consistent estimator of g_α by the central limit theorem (Wang and Yang [2007]).
Given the response variable $\hat{y}_h$, we can define the sub-model

$$\hat{y}_h = g_h(x_h) + \epsilon. \tag{2.67}$$

Based on the sample $\{(x_{ih}, \hat{y}_{ih}) : i = 1, 2, \ldots, n\}$, the LPL estimator of $g_h(x_h)$, denoted by $\tilde{g}_h(x_h)$, can be obtained in the same manner as the LPL estimator of $m(x)$ in (2.25). The LPL

estimators of the other additive functions follow the same procedure.
Let $\tilde{\boldsymbol{g}} = (\tilde{\boldsymbol{g}}_1, \tilde{\boldsymbol{g}}_2, \ldots, \tilde{\boldsymbol{g}}_D)$ be the second-stage estimator of $\boldsymbol{g}$. We refer to these estimators as the SBLPL estimators.
The above estimation procedure is said to reduce bias by undersmoothing in the first-stage. Thus, k should be reasonably large. This results in an estimator with a large variance. In the second-stage, the variance is averaged-out.
For more details about the theory and application of the spline-backfitting estimation procedure, see Wang and Yang [2007] and Wang and Yang [2009] and Ma and Yang [2014].

Partial linear modelling

A special case of the additive regression model (2.52) arises when the response variable y depends linearly on all except one of the covariates $\mathbf{X} = (X_1, X_2, \ldots, X_{D-1})^\intercal$ whereas its dependence on the other covariate X_D, hereafter denoted by S, is non-linear. This gives rise to the partially linear model (PLM)

$$Y = \mathbf{X}^\intercal \boldsymbol{\beta} + g(S) + \epsilon, \qquad \epsilon \sim \mathcal{N}(0, \sigma^2), \tag{2.68}$$

where $g(\cdot)$ is an unknown univariate (hence non-parametric) function and $\boldsymbol{\beta}$ is a vector of unknown global parameters. The regression function $m(\mathbf{x}, s)$ is given by

$$m(\mathbf{x}, s) \equiv \mathbb{E}[Y|\mathbf{X} = \mathbf{x}, S = s] = \mathbf{X}^\intercal \boldsymbol{\beta} + g(S). \tag{2.69}$$

The PLM (2.68) belongs to a class of models referred to as semi-parametric models. This is because it includes a parametric term $\mathbf{X}^\intercal \boldsymbol{\beta}$ and a non-parametric term $g(S)$. The model combines the advantages of a parametric model (notably, interpretability) and the flexibility of a non-parametric model. If the specification (2.68) is correct, the PLM provides a flexible parsimonious description of the data compared to (2.52).
In order to estimate (2.68), we must estimate the parameter vector $\boldsymbol{\beta}$ and the unknown univariate function $g(\cdot)$. The non-parametric term can be estimated using any of the available non-parametric smoothers such as the LPL estimator (2.29). Next, we present an LPL estimation procedure for a PLM.
As before, we approximate the unknown function $g(\cdot)$ locally using a p^{th} degree polynomial function

$$\begin{aligned} g(s) &\approx g(u) + g^{(1)}(u)(s-u) + \cdots + g^{(p)}(u)(s-u)^p \\ &= \gamma_0 + \gamma_1(s-u) + \cdots + \gamma_p(s-u)^p, \end{aligned} \tag{2.70}$$

Algorithm 1 Fitting a PLM (2.68) using an LPL estimator (2.29)

Step 0: Obtain an initial estimate of β, say $\hat{\beta}^{(0)}$

Step 1: Obtain $\hat{g}(\cdot)$ by maximising (2.71) after substituting $\boldsymbol{\beta}$

Step 2: Update $\hat{\boldsymbol{\beta}}$ by regressing $\{y_i - \hat{g}(s_i)\}_{i=1}^n$ on $\{\mathbf{x}_i\}_{i=1}^n$

Step 3: Repeat Step 1 and 2 until convergence

for s in a neighborhood of $u \in \mathcal{U}$, where $\mathcal{U}$ is a set of local grid points in the domain of s and $\gamma_j = g^{(j)}(u)$ for $j = 0, 1, \ldots, p$.

Consider a random sample $\{(\mathbf{x}_i, s_i, y_i) : i = 1, 2, \ldots, n\}$, where $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iD})^\intercal$ from the population $(\mathbf{X}, S, Y) \in \mathbb{R}^{D-1} \times \mathbb{R} \times \mathbb{R}$ where the conditional distribution of Y given $\mathbf{X} = \mathbf{x}$ and $S = s$ is Gaussian with mean $m(\mathbf{x}, s)$ and the variance σ^2. For a given value of $\boldsymbol{\beta}$, we can obtain $\hat{\gamma}_j$ for $j = 0, 1, \ldots, p$ to maximise the local polynomial log-likelihood function

$$\sum_{i=1}^{n} \log \mathcal{N}(y_i | \mathbf{x}_i^\intercal \boldsymbol{\beta} + \sum_{j=0}^{p} \gamma_j (s_i - u)^j, \sigma^2) K_h(s_i - u). \tag{2.71}$$

Let $g(u) = \gamma_0$ and maximise (2.71) to obtain $\hat{g}(u)$ for $u \in \mathcal{U}$. To obtain $\hat{g}(s_i)$ for $s_i \notin \mathcal{U}$, we can make use of, say linear interpolation.

If $\boldsymbol{\beta}$ is unknown, we can make use of Algorithm 1 to simultaneously estimate $g(\cdot)$ and $\boldsymbol{\beta}$. Let $\mathbf{x} = (x_1, x_2, \ldots, x_D)^\intercal$, another approach to estimate a PLM is to simultaneously estimate both $g(\cdot)$ and $\boldsymbol{\beta}$ locally by maximising

$$\sum_{i=1}^{n} \log \mathcal{N}\left(y_i | \sum_{d=1}^{D} \left(\sum_{j=0}^{p} \gamma_{jd} (s_i - u)^j \right) x_{id} + \sum_{j=0}^{p} \gamma_j (s_i - u)^j, \sigma^2 \right) K_h(s_i - u), \tag{2.72}$$

where $\gamma_{jd} = \beta_d^{(j)}(u)$ for $j = 0, 1, \ldots, p$ and $\boldsymbol{\beta}(u) = (\beta_1(u), \beta_2(u), \ldots, \beta_D(u))$.

Let $\hat{g}(u)$ for $u \in \{u_1, u_2, \ldots, u_N\}$ and obtain $\hat{g}(s_i)$ via interpolation. Estimate the global parameters $\boldsymbol{\beta}$ by maximising the global log-likelihood function

$$\sum_{i=1}^{n} \log \mathcal{N}(y_i | \mathbf{x}_i^\intercal \boldsymbol{\beta} + \hat{g}(s_i), \sigma^2), \tag{2.73}$$

with respect to $\boldsymbol{\beta}$ (see Fan and Gijbels [1996] and Carroll et al. [1997]). In contrast to Algorithm 1, the implementation of the above procedure is non-iterative. Other non-iterative procedures for simultaneously estimating $\boldsymbol{\beta}$ and $g(\cdot)$ using the LPL estimation procedure were proposed by Robinson [1988], Speckman [1988] and Hamilton and Truong [1997]. See Härdle et al. [2000] and Härdle et al. [2004], for more details about partial linear models.

Single-index modelling

Another popular approach to overcome the dimensionality problem is to assume that the response $Y \in \mathbb{R}$ depends on the covariates $\mathbf{X} \in \mathbb{R}^D$ through a non-parametric function of some linear combination of $\mathbf{X}$. This gives rise to the single-index model (SIM)

$$y = g(\mathbf{X}^\intercal \boldsymbol{\beta}) + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma^2). \tag{2.74}$$

The problem is now reduced to an estimation of an unknown univariate function $g(\cdot)$ and a $D-$dimensional parameter $\boldsymbol{\beta}$, also known as the index parameter. For identifiability, $\|\boldsymbol{\beta}\| = 1$ and the first non-zero element of $\boldsymbol{\beta}$ is positive.
Model (2.74) has the following properties that makes it attractive in practice (see Carroll et al. [1997]):

1. By projecting $\mathbf{X}$ onto a linear subspace, the dimension reduces to a univariate, model (2.74) provides a readily interpretable way of obtaining this reduction;

2. if $g(\cdot)$ is a monotone function, the index parameter $\boldsymbol{\beta}$ can be interpreted as in ordinary linear models; and

3. Given an estimate of the index parameter $\boldsymbol{\beta}$, the estimation of model (2.74) reduces to the univariate covariate case and the methods in section 2.2.1 are applicable for estimating $g(\cdot)$.

To illustrate the final point above, consider a random sample $\{(\mathbf{x}_i^\intercal, y_i) : i = 1, 2, \ldots, n\}$. For a given value of $\boldsymbol{\beta}$, the regression function

$$m(u) = \mathbb{E}[Y|\mathbf{X}^\intercal \boldsymbol{\beta} = u] = g(u|\boldsymbol{\beta}) \tag{2.75}$$

can be estimated using the local polynomial regression estimator (2.29) at some grid point u. By Taylor's expansion, $g(\cdot)$ can be approximated locally using a polynomial function of degree p

$$\begin{aligned} g(\mathbf{x}^\intercal \boldsymbol{\beta}) &\approx g(u) + g^{'}(u)[\mathbf{x}^\intercal \boldsymbol{\beta} - u] + \cdots + g^{(p)}(u)[\mathbf{x}^\intercal \boldsymbol{\beta} - u]^p \\ &\equiv \gamma_0 + \gamma_1(\mathbf{x}^\intercal \boldsymbol{\beta} - u) + \cdots + \gamma_p(\mathbf{x}^\intercal \boldsymbol{\beta} - u), \end{aligned} \tag{2.76}$$

for $\mathbf{x}^\intercal \boldsymbol{\beta}$ in the neighbourhood of u, where $\gamma_j = g^{(j)}(u)$ for $j = 0, 1, \ldots, p$. Let $\hat{\gamma}_j = g^{(j)}(u)$ for $j = 0, 1, \ldots, p$ be the estimates of the local parameters $\{\gamma_j\}_{j=0}^p$ obtained from maximising the

local log-likelihood

$$\sum_{i=1}^{n} \log \mathcal{N}(y_i | \sum_{j=0}^{p} \gamma_j (\mathbf{x}^\intercal \boldsymbol{\beta} - u)^j, \sigma^2) K_h(\mathbf{x}^\intercal \boldsymbol{\beta} - u) \tag{2.77}$$

with respect to $\{\gamma_j\}_{j=0}^{p}$ at grid point u. The estimate of (2.75) is given by $\hat{\gamma}_0$, that is $\hat{m}(u) = \hat{\gamma}_0$. The practical significance of model (2.74) has been emphasised (as above). The model has been motivated as a model for dimension reduction having some similarities to the projection pursuit regression method (see Li [1991] and Hardle et al. [1993] and the references therein). Due to its flexibility and dimension reduction properties, the SIM has proven to be useful in discrete choice analysis in economics (see Stoker [1986] and Li [2011]); gene expression analysis in genetics (see Jiang and Sun [2021]) and volatility modelling in finance, where σ^2 is assumed to be a non-parametric of a single index $\mathbf{X}^\intercal \boldsymbol{\beta}$ and $g(\cdot) \equiv 0$ (see Xia et al. [2002]). For more details about SIMs (2.74) see Ichimura [1993] and Hardle et al. [1993].

Along with partially linear models, SIMs belong to a broader class of models referred to as partially linear single index models. See Carroll et al. [1997] for a study of this class of models within a generalised modelling framework.

In addition to addressing the dimensionality issue, the models discussed above retain most of the benefits of the multivariate non-parametric regression model (2.47) and the fully parametric regression model (2.23) without taking on their shortcomings. These includes, among others, flexibility and interpretability. A good example of a model that combines both of these advantages is the varying-coefficient model (Hastie and Tibshirani [1993]). The varying-coefficient model assumes that the regression parameters of the linear model (2.23) are non-parametric functions of a covariate s

$$y = \beta_0(s) + \beta_1(s)x_1 + \beta_2(s)x_2 + \cdots + \beta_D(s)x_D + \epsilon. \tag{2.78}$$

Model (2.78) is still a linear model, that is linear in the covariates. However, it is non-linear in the regression parameters. Thus, model (2.78) is a semi-parametric model.

Notice that, due to its flexibility, the varying coefficient functions allow for some form of interaction between the covariates $x_d : d = 1, 2, \ldots, D$ and s. In addition, it is easy to see that the model is not sensitive to the dimensionality issue. That is, we can increase the number of covariates D without affecting the quality of the estimated varying-coefficient functions.

For more details about the varying-coefficient model, it's practical and theoretical advantages, see Hastie and Tibshirani [1993] and the overview (Fan and Zhang [2008]).

Chapter 3

Estimation

In this chapter, we present two local-polynomial likelihood (LPL-) based approaches to estimate the general Gaussian mixture of non-parametric regressions (1.1) via the naïve EM algorithm. The presentation in this chapter is to further highlight the nature and origin of the label-switching. Finally, we give a comprehensive overview of the label-switching problem and the first approach proposed to address it.

3.1 LPL estimation procedures for the general model (1.1)

In this section, we discuss LPL estimation for the general model (1.1). Consider a random sample $\{(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i, y_i) : i = 1, 2, \ldots, n\}$ from model (1.1). The corresponding log-likelihood function is given by

$$\ell(\boldsymbol{\theta}) = \sum_{i=1}^{n} \log\Bigg[\sum_{k=1}^{K} \pi_k(s_i)\mathcal{N}\big(y_i|\mathbf{m}_k(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i), \sigma_k^2(s_i)\big)\Bigg], \tag{3.1}$$

where

$$\mathbf{m}_k(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i) = \mathbf{x}_i^\intercal \boldsymbol{\beta}_k + \mathbf{z}_i^\intercal \boldsymbol{\gamma}_k(s_i) + \sum_{c=1}^{D_3} g_{k,c}(t_{ic})$$

and $\boldsymbol{\theta} = (\boldsymbol{\pi}, \boldsymbol{\sigma}^2, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{g}) = (\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_K; \boldsymbol{\sigma}_1^2, \ldots, \boldsymbol{\sigma}_K^2; \boldsymbol{\beta}_1, \ldots, \boldsymbol{\beta}_K; \boldsymbol{\gamma}_1, \ldots, \boldsymbol{\gamma}_K;$
$\boldsymbol{g}_1, \ldots, \boldsymbol{g}_K)$, with $\boldsymbol{\pi}_k = (\pi_k(s_1), \ldots, \pi_k(s_n))$, $\boldsymbol{\sigma}_k^2 = (\sigma_k^2(s_1), \ldots, \sigma_k^2(s_n))$, $\boldsymbol{\beta}_k = (\beta_{k,0}, \beta_{k,1}, \ldots, \beta_{k,D_1})$, $\boldsymbol{\gamma}_k = (\gamma_{k,0}(s_1), \ldots, \gamma_{k,D_2}(s_n))$, $\boldsymbol{g}_k = (\boldsymbol{g}_{k,1}, \ldots, \boldsymbol{g}_{k,D_3})$ and $\boldsymbol{g}_{k,c} = (g_{k,c}(t_{1c}), \ldots, g_{k,c}(t_{nc}))$, is the vector of all the model parameters and non-parametric functions. Direct LPL estimation of $\boldsymbol{\theta}$ poses a computational challenge due to the presence of both parametric and non-parametric terms. We know from section 2.2 that non-parametric estimation uses data in a local neighbour-

hood of a given local grid point. However, estimating global parameters, such as $\boldsymbol{\beta}$, requires the use of all the observed data. In the following, we present two LPL-based estimation procedures for estimating model (1.1): the spline-backfitted LPL (SBLPL) estimation procedure and the one-step backfitting LPL (OSBLPL) estimation procedure. These estimation procedures are multi-stage estimation procedures. In the first-stage, the SBLPL estimation procedure assumes that model (1.1) is fully non-parametric and the OSBLPL estimation procedure assumes that model (1.1) is fully parametric.
In the second-stage, the parametric (non-parametric) term(s) are estimated given the first-stage estimators of the non-parametric (parametric) terms. Further stages, if any, involve improving the estimators obtained at previous stages. More details of these procedures will be given in the following subsections. For each of these estimation procedures, we will also discuss the EM fitting algorithm.

3.1.1 Spline-backfitted LPL (SBLPL) estimation procedure

In order to estimate the SPGMPLAMs (1.6), Zhang and Pan [2022] proposed spline-backfitted kernel (SBK) estimation via an EM-type algorithm. The SBK estimation procedure is a two-stage procedure. In the first-stage, we parameterise the non-parametric functions $g_k(t_c)$, for $k = 1, 2, \ldots, K$ and $c = 1, 2, \ldots, D_3$, using B-spline basis functions. The resulting model reduces into a GMLRs model which can be easily estimated using the usual maximum likelihood estimation via the EM algorithm. In the second-stage, given the estimates of the global parameters, $\boldsymbol{\beta}_k$, for $k = 1, 2, \ldots, K$, and the non-parametric functions $g_k(t_j)$, for $k = 1, 2, \ldots, K$ and $j \neq c$, we use a kernel (local-constant) estimator, hereafter referred to as local-constant estimator (LCE), to estimate the non-parametric function $g_k(t_c)$. This is repeated to estimate each function $g_k(t_c)$, for $c = 1, 2, \ldots, D_3$, using a backfitting approach (see subsection 2.2.5). Prior to Zhang and Pan [2022], the SBK estimation procedure was proposed by Zhang and Zheng [2018] to estimate the SPGMAMs (1.11). As already mentioned in section 1.1, the SPGMAMs is a special case of the SPGMPLAMs (1.6) when the linear additive term $(\mathbf{x}^\intercal\boldsymbol{\beta}_k : k = 1, 2, \ldots, K)$ is equal to zero.

Estimation procedure We propose to extend the above SBK estimation procedure to estimate model (1.1). The proposed procedure will make use of LPL estimators to estimate the non-parametric terms. Hence, we refer to the proposed procedure as the spline-backfitted LPL (SBLPL) estimation procedure. Next, we give details of each stage of the SBLPL estimation procedure for estimating model (1.1).
In the first-stage, towards parameterising the non-parametric functions, we first choose an integer $N_n = \lceil n^{1/3}\log(n) \rceil$ as the number of internal knots for the spline function to be de-

fined below. The knots are an ordered sequence of equally-spaced points given by the set $\{\xi_r : r = 0, 1, \ldots, N_n + 1\}$. The knots $\{\xi_r : r = 1, 2, \ldots, N_n\}$ and $\{\xi_0, \xi_{N_n+1}\}$ are the internal and boundary knots, respectively. As already mentioned in subsection 2.2.5, to reduce the bias in estimating the non-parametric functions in the first-stage, k, the order of the B-spline, must be reasonably large. Thus, we choose $k = 4$ and use a linear combination of fourth-order (cubic) B-splines basis functions. Thus, for $J = 0, 1, \ldots, N_n + 1$, the normalised cubic B-splines basis function is denoted by $B_{J4}(s)$ (see subsection 2.2.5).
We first define the spline functions for the mixing proportion function and variance function, respectively, as

$$\pi(s) = \omega_0 + \sum_{J=1}^{N_n+3} \omega_J B_{J4}(s) \tag{3.2}$$

and

$$\sigma^2(s) = \lambda_0 + \sum_{J=1}^{N_n+3} \lambda_J B_{J4}(s). \tag{3.3}$$

Let $\mathbf{\Gamma}(s, \mathbf{z}) = \mathbf{z}^\intercal \boldsymbol{\gamma}(s)$ and define the varying coefficient spline functions as

$$\mathbf{\Gamma}(s, \mathbf{z}) = \sum_{b=0}^{D_2} \left(\sum_{J=1}^{N_n+3} \alpha_{b,J} B_{J4}(s) \right) z_b. \tag{3.4}$$

Finally, we define the additive spline functions as

$$\boldsymbol{g}(\mathbf{t}) = \sum_{c=1}^{D_3} \sum_{J=1}^{N_n+3} \eta_{c,J} B_{J4}(t_c). \tag{3.5}$$

The log-likelihood function based on (3.2), (3.3), (3.4) and (3.5) is given by

$$\ell\{\boldsymbol{\theta}\} = \sum_{i=1}^{n} \log \Bigg[\sum_{k=1}^{K} \left(\omega_{k,0} + \sum_{J=1}^{N_n+3} \omega_{k,J} B_{J4}(s_i) \right) \mathcal{N}\Bigg(y_i | \mathbf{m}_k(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i), \lambda_{k,0} + \sum_{J=1}^{N_n+3} \lambda_{k,J} B_{J4}(s_i) \Bigg) \Bigg], \tag{3.6}$$

where

$$\mathbf{m}_k(s,\mathbf{x},\mathbf{z},\mathbf{t}) = \sum_{a=1}^{D_1}\beta_{k,a}x_a + \sum_{b=0}^{D_2}\left(\sum_{J=1}^{N_n+3}\alpha_{k,b,J}B_{J4}(s)\right)z_b + \sum_{J=1}^{N_n+3}\sum_{c=1}^{D_3}\eta_{k,c,J}B_{J4}(t_c) \tag{3.7}$$

and $\boldsymbol{\theta} = (\omega_{k,J}, \lambda_{k,J}, \beta_{k,a}, \alpha_{k,b,J}, \eta_{k,c,J})_{1\le k\le K, 1\le J\le N_n+3, 1\le a\le D_1, 0\le b\le D_2, 1\le c\le D_3}$ is the vector of all the parameters.

The respective first-stage (spline) estimators of $\pi_k(s)$ and $\sigma_k^2(s)$, for $k = 1, 2, \ldots, K$, are given by

$$\hat{\pi}_k(s) = \hat{\omega}_{k,0} + \sum_{J=1}^{N_n+3}\hat{\omega}_{k,J}B_{J4}(s), \tag{3.8}$$

$$\hat{\sigma}_k^2(s) = \hat{\lambda}_{k,0} + \sum_{J=1}^{N_n+3}\hat{\lambda}_{k,J}B_{J4}(s). \tag{3.9}$$

Note that the linear combinations in both (3.8) and (3.9) are not guaranteed to satisfy the constraints $0 < \pi_k(s) < 1$ and $0 < \sigma_k^2(s) < \infty$, respectively. In order to ensure that these constraints are satisfied, we set

$$\text{logit}(\hat{\pi}_k(s)) = \hat{\omega}_{k,0} + \sum_{J=1}^{N_n+3}\hat{\omega}_{k,J}B_{J4}(s), \tag{3.10}$$

$$\ln(\hat{\sigma}_k^2(s)) = \hat{\lambda}_{k,0} + \sum_{J=1}^{N_n+3}\hat{\lambda}_{J}B_{J4}(s), \tag{3.11}$$

where $\text{logit}(x) = \ln\left(\frac{x}{1-x}\right)$ and $\ln(x)$ is the natural logarithm of x.

The respective first-stage (spline) estimators of $\mathbf{m}_k(s,\mathbf{x},\mathbf{z},\mathbf{t})$, $\gamma_{k,b}(s)$ and $g_{k,c}(t_c)$, for $1 \le k \le K, 0 \le b \le D_2$ and $1 \le c \le D_3$, are given by

$$\hat{\mathbf{m}}_k(s,\mathbf{x},\mathbf{z},\mathbf{t}) = \sum_{a=0}^{D_1}\hat{\beta}_{k,a}x_a + \sum_{b=0}^{D_2}\left(\sum_{J=1}^{N_n+3}\hat{\alpha}_{k,b,J}B_{J4}(s)\right)z_b + \sum_{J=1}^{N_n+3}\sum_{c=1}^{D_3}\hat{\eta}_{k,c,J}B_{J4}(t_c), \tag{3.12}$$

$$\hat{\gamma}_{k,0}(s) = \sum_{J=1}^{N_n+3} \hat{\alpha}_{k,0,J} B_{J4}(s) - \frac{1}{n}\sum_{i=1}^{n}\sum_{J=1}^{N_n+3} \hat{\alpha}_{k,0,J} B_{J4}(s_i), \tag{3.13}$$

$$\hat{\gamma}_{k,b}(s) = \sum_{J=1}^{N_n+3} \hat{\alpha}_{k,b,J} B_{J4}(s), \tag{3.14}$$

$$\hat{g}_{k,c}(t_c) = \sum_{J=1}^{N_n+3} \hat{\eta}_{k,c,J} B_{J4}(t_c) - \frac{1}{n}\sum_{i=1}^{n}\sum_{J=1}^{N_n+3} \hat{\eta}_{k,c,J} B_{J4}(t_{ic}), \tag{3.15}$$

where $\hat{\boldsymbol{\theta}} = (\hat{\omega}_{k,J}, \hat{\lambda}_{k,J}, \hat{\beta}_{k,a}, \hat{\alpha}_{k,b,J}, \hat{\eta}_{k,c,J})_{1\le k\le K, 1\le J\le N_n+3, 1\le a\le D_1, 0\le b\le D_2, 1\le c\le D_3}$ is the maximum likelihood estimator of $\boldsymbol{\theta}$ obtained as a solution to

$$\hat{\boldsymbol{\theta}} = \max_{\boldsymbol{\theta}} \ell\{\boldsymbol{\theta}\}. \tag{3.16}$$

In the second-stage, we use the LPL estimators to non-parametrically estimate each unknown function in turn given the parametric estimates from the first-stage and/or current estimates of the other unknown functions. Given $(\hat{\beta}_{k,a}, \hat{\gamma}_{k,b}(s), \hat{g}_{k,c}(t_c))_{1\le k\le K, 1\le a\le D_1, 0\le b\le D_2, 1\le c\le D_3}$, we estimate the unknown functions $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}^2$ by maximising the log LPL function

$$\ell\{\boldsymbol{\pi}, \boldsymbol{\sigma}^2\} = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k(u)\mathcal{N}\left(y_i|\hat{\mathbf{m}}_k(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i), \sigma_k^2(u)\right)\right] \times K_h(s_i - u). \tag{3.17}$$

Let $\tilde{\boldsymbol{\pi}}$ and $\tilde{\boldsymbol{\sigma}}^2$ be the LPL estimators of the non-parametric functions $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}^2$, respectively. Next, given $(\hat{\beta}_{k,a}, \hat{g}_{k,c}(t_c))_{1\le k\le K, 1\le a\le D_1, 1\le c\le D_3}$, $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\sigma}}^2$ and the current or first-stage estimates of $\boldsymbol{\gamma}_{\cdot j}(s) = (\gamma_{1,j}(s), \gamma_{2,j}(s), \ldots, \gamma_{K,j}(s))$, for $j \neq b$, we estimate $\boldsymbol{\gamma}_{\cdot b}(s)$ by maximising the log LPL function

$$\ell\{\boldsymbol{\gamma}_{\cdot b}\} = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \tilde{\pi}_k(s_i)\mathcal{N}\left(y_i|\mathbf{x}_i^{\intercal}\hat{\boldsymbol{\beta}}_k + \hat{\boldsymbol{\Gamma}}_k(s_i, \mathbf{z}_{i,-b}) + \gamma_{k,b}(u)z_{ib} + \hat{\boldsymbol{g}}_k(\mathbf{t}_i), \tilde{\sigma}_k^2(s_i)\right)\right] K_h(s_i - u), \tag{3.18}$$

where

$$\hat{\mathbf{\Gamma}}_k(s, \mathbf{z}_{-b}) = \sum_{j \neq b} \hat{\gamma}_{k,b}(s) z_b, \tag{3.19}$$

$$\hat{\boldsymbol{g}}_k(\mathbf{t}) = \sum_{c=1}^{D_3} \hat{g}_{k,c}(t_c) \tag{3.20}$$

and $\mathbf{z}_{-b}$ denotes the vector of covariates $\mathbf{z}$ excluding the covariate z_b.
Let $\tilde{\boldsymbol{\gamma}}_{\cdot b}(s)$ be the resulting estimator of $\boldsymbol{\gamma}_{\cdot b}(s)$. Next, we set $\tilde{\boldsymbol{\gamma}}_{\cdot b}(s)$ to be the new value of $\boldsymbol{\gamma}_{\cdot b}(s)$ and repeat the above procedure to estimate the other varying coefficient functions. Let $\tilde{\boldsymbol{\gamma}}$ be the SBLPL estimator of $\boldsymbol{\gamma}$.
Finally, given $(\hat{\beta}_{k,a})_{1 \leq k \leq K, 1 \leq a \leq D_1}$, $\tilde{\boldsymbol{\gamma}}$, $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\sigma}}^2$ and the current estimates of the functions $g_{k,j}(t_j)$, for $j \neq c$, we estimate $g_{k,c}(t_c)$ by maximising the following log LPL function

$$\ell\{\boldsymbol{g}_{\cdot c}\} = \sum_{i=1}^{n} \log \Bigg[\sum_{k=1}^{K} \tilde{\pi}_k(s_i) \mathcal{N}\Big(y_i | \mathbf{x}_i^\intercal \hat{\boldsymbol{\beta}}_k + \tilde{\mathbf{\Gamma}}_k(s_i, \mathbf{z}_i) + \hat{\mathbf{g}}_k(\mathbf{t}_{i,-c}) + g_{k,c}(u), \tilde{\sigma}_k^2(s_i) \Big) \Bigg] K_h(t_{ic} - u), \tag{3.21}$$

where $\mathbf{t}_{-c}$ denotes the vector of covariates $\mathbf{t}$ excluding the covariate t_c.
Let $\tilde{\boldsymbol{g}}_{\cdot c}$ be the resulting estimator of $\boldsymbol{g}_{\cdot c}$. Next, we set $\tilde{\boldsymbol{g}}_{\cdot c}$ to be the new value of $\boldsymbol{g}_{\cdot c}$ and repeat the above procedure to estimate the other additive functions. Let $\tilde{\boldsymbol{g}}$ be the SBLPL estimator of $\boldsymbol{g}$.
To improve the parameter estimates $(\hat{\beta}_{k,a})_{1 \leq k \leq K, 1 \leq a \leq D_1}$, we can maximise the log-likelihood function

$$\ell\{\boldsymbol{\beta}\} = \sum_{i=1}^{n} \log \Bigg[\sum_{k=1}^{K} \tilde{\pi}_k(s_i) \mathcal{N}\Big(y_i | \mathbf{x}_i^\intercal \boldsymbol{\beta}_k + \tilde{\mathbf{\Gamma}}_k(s_i, \mathbf{z}_i) + \tilde{\boldsymbol{g}}_k(\mathbf{t}_i) + \tilde{\sigma}_k^2(s_i) \Big) \Bigg] \tag{3.22}$$

given the second-stage estimators $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\sigma}}^2$, $\tilde{\boldsymbol{\gamma}}$ and $\tilde{\boldsymbol{g}}$. Let $\tilde{\boldsymbol{\beta}}$ be the resulting improved estimator of $\boldsymbol{\beta}$. The estimators $\tilde{\boldsymbol{\beta}}$, $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\sigma}}^2$, $\tilde{\boldsymbol{\gamma}}$ and $\tilde{\boldsymbol{g}}$ can be referred to as the SBLPL estimators.

Fitting algorithm In order to obtain the SBLPL estimators, we make use of the EM algorithm to maximise the log-likelihood functions (3.6) and (3.22) and the log-LPL functions (3.17)-(3.21). Maximising (3.6) and (3.22) is straightforward, following the traditional EM algorithm. However, as already mentioned in section 1.2, using the traditional EM algorithm to maximise the log-LPL functions may result in label-switching. We now give details of the naïve fitting algorithm (see subsection 1.2.1) used to carry out the above two-stage estima-

tion procedure. Our aim here is to re-emphasise the nature and origin of the label-switching phenomenon studied in this thesis.

First-Stage: The EM algorithm to maximise (3.6)

Define a latent variable q_{ik} which takes a value 1 if $(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i, y_i)$ belongs to the k^{th} component and 0 otherwise. Let $\{(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i, y_i, \mathbf{q}_i) : i = 1, 2, \ldots, n\}$ be the complete-data, where $\mathbf{q}_i = (q_{i1}, q_{i2}, \ldots, q_{iK})$.
Let $\boldsymbol{\theta} = (\boldsymbol{\theta}_{(1)}, \boldsymbol{\theta}_{(2)})$, where $\boldsymbol{\theta}_{(1)} = \boldsymbol{\omega}$ and $\boldsymbol{\theta}_{(2)} = (\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\eta}, \boldsymbol{\lambda})$. The complete-data log-likelihood is given by

$$\ell^c\{\boldsymbol{\theta}\} = \ell^c\{\boldsymbol{\theta}_{(1)}\} + \ell^c\{\boldsymbol{\theta}_{(2)}\}, \tag{3.23}$$

where

$$\ell^c\{\boldsymbol{\theta}_{(1)}\} = \sum_{i=1}^{n}\sum_{k=1}^{K} q_{ik}\log\left[\omega_{k,0} + \sum_{J=1}^{N_n+3}\omega_{k,J}B_{J4}(s_i)\right] \tag{3.24}$$

and

$$\ell^c\{\boldsymbol{\theta}_{(2)}\} = \sum_{i=1}^{n}\sum_{k=1}^{K} q_{ik}\log\mathcal{N}\left(y_i|\mathbf{m}_k(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i), \lambda_{k,0} + \sum_{J=1}^{N_n+3}\lambda_{k,J}B_{J4}(s_i)\right). \tag{3.25}$$

Let $\boldsymbol{\theta}^{(r)}$ be the parameter vector $\boldsymbol{\theta}$ at the r^{th} iteration of the EM algorithm. At the $(r+1)^{th}$ iteration of the E-step, we calculate the conditional expectation of $\ell^c\{\boldsymbol{\theta}\}$, denoted by $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$, which reduces to calculating the conditional expected value of the latent variable q_{ik}, for $k = 1, 2, \ldots, K$ and $i = 1, 2, \ldots, n$, as

$$p_{ik}^{(r+1)} = \frac{\pi^{(r)}(s_i)\mathcal{N}(y_i|\mathbf{m}_k^{(r)}(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i), \sigma_k^{2(r)}(s_i))}{\sum_{\ell=1}^{K}\pi^{(r)}(s_i)\mathcal{N}(y_i|\mathbf{m}_\ell^{(r)}(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i), \sigma_\ell^{2(r)}(s_i))}. \tag{3.26}$$

At the M-step, we update $\boldsymbol{\theta}^{(r)}$, by maximising $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ with q_{ik} substituted by $p_{ik}^{(r+1)}$, as follows

1. First calculate $\boldsymbol{\theta}_{(1)}^{(r+1)}$ as
$$\boldsymbol{\theta}_{(1)}^{(r+1)} = \max Q(\boldsymbol{\theta}_{(1)}|\boldsymbol{\theta}_{(1)}^{(r)}) \tag{3.27}$$
to obtain $(\omega_{k,J}^{(r+1)})_{1\leq k\leq K, 1\leq J\leq N_n+3}$. Finally, substitute $(\omega_{k,J}^{(r+1)})_{1\leq k\leq K, 1\leq J\leq N_n+3}$ into (3.10) to obtain $\pi_k^{(r+1)}(s)$, for $k = 1, 2, \ldots, K$.

2. Finally, calculate $\boldsymbol{\theta}_{(2)}^{(r+1)}$ as

$$\boldsymbol{\theta}_{(2)}^{(r+1)} = \max Q(\boldsymbol{\theta}_{(2)}|\boldsymbol{\theta}_{(2)}^{(r)}) \tag{3.28}$$

to obtain $(\lambda_{k,J}^{(r+1)}, \beta_{k,a}^{(r+1)}, \alpha_{k,b,J}^{(r)}, \eta_{k,c,J}^{(r+1)})_{1\leq k\leq K, 1\leq J\leq N_n+3, 1\leq a\leq D_1, 0\leq b\leq D_2, 1\leq c\leq D_3}$. Use (3.12) to calculate $\mathbf{m}_k^{(r+1)}(s,\mathbf{x},\mathbf{z},\mathbf{t})$ and (3.11) to calculate $\sigma_k^{2(r+1)}(s)$, for $k = 1, 2, \ldots, K$.

Repeat the above E- and M-step until convergence.
Let $\boldsymbol{\theta}^{(R)}$ be the parameter vector at convergence of the above EM algorithm. The first-stage estimator of $\boldsymbol{\theta}$ is given by $\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}^{(R)}$. Substituting the $\hat{\boldsymbol{\theta}}$ into (3.10)-(3.11) and (3.13)-(3.15), we get the first-stage cubic spline estimators, denoted by $\hat{\boldsymbol{\pi}}$, $\hat{\boldsymbol{\sigma}}^2$, $\hat{\boldsymbol{\gamma}}$ and $\hat{\boldsymbol{g}}$, and the estimator $\hat{\boldsymbol{\beta}}$ of the non-parametric functions $\boldsymbol{\pi}$, $\boldsymbol{\sigma}^2$, $\boldsymbol{\gamma}$, $\boldsymbol{g}$ and the parameter $\boldsymbol{\beta}$, respectively.

Second-Stage: The EM algorithm to maximise (3.17)

In the second-stage, we maximise each log-LPL function using the EM algorithm as follows. Given the first-stage estimators $\hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{\gamma}}$ and $\hat{\boldsymbol{g}}$, let $u \in \mathcal{U}$, where $\mathcal{U}$ is the set of local points in the domain of s and define the complete-data log-likelihood version of (3.17) as

$$\ell^c\{\boldsymbol{\theta}(u)\} = \ell^c\{\boldsymbol{\pi}(u)\} + \ell^c\{\boldsymbol{\sigma}^2(u)\}, \tag{3.29}$$

where

$$\begin{aligned}
\ell^c\{\boldsymbol{\pi}(u)\} &= \sum_{i=1}^{n}\sum_{k=1}^{K} q_{ik}\log\big[\pi_k(u)\big]K_h(s_i-u), \\
\ell^c\{\boldsymbol{\sigma}^2(u)\} &= \sum_{i=1}^{n}\sum_{k=1}^{K} q_{ik}\log\mathcal{N}\big(y_i|\hat{\mathbf{m}}_k(s_i,\mathbf{x}_i,\mathbf{z}_i,\mathbf{t}_i),\sigma_k^2(u)\big)K_h(s_i-u)
\end{aligned}$$

and $\boldsymbol{\theta}(u) = (\boldsymbol{\pi}(u), \boldsymbol{\sigma}^2(u))$, with $\boldsymbol{\pi}(u) = (\pi_1(u), \pi_2(u), \ldots, \pi_K(u))$ and $\boldsymbol{\sigma}^2(u) = (\sigma_1^2(u), \sigma_2^2(u), \ldots, \sigma_K^2(u))$, is the vector of local parameters at the local point u. At the $(r+1)^{th}$ iteration of the E-step, we calculate the responsibilities as

$$p_{ik}^{(r+1)}(u) = \frac{\pi_k^{(r)}(u)\mathcal{N}(y_i|\hat{\mathbf{m}}_k(s_i,\mathbf{x}_i,\mathbf{z}_i,\mathbf{t}_i),\sigma_k^{2(r)}(u))}{\sum_{\ell=1}^{K}\pi_\ell^{(r)}(u)\mathcal{N}(y_i|\hat{\mathbf{m}}_\ell(s_i,\mathbf{x}_i,\mathbf{z}_i,\mathbf{t}_i),\sigma_\ell^{2(r)}(u))}. \tag{3.30}$$

At the $(r+1)^{th}$ iteration of the M-step, update $\boldsymbol{\theta}^{(r)}(u)$ for each local point $u \in \mathcal{U}$ as

$$\pi_k^{(r+1)}(u) = \hat{\pi}_{k0}(u), \tag{3.31}$$

where $\hat{\pi}_{k0}(u)$, for $k = 1, 2, \ldots, K$, is the LPL estimator obtained by maximising the expected value of $\ell^c\{\boldsymbol{\pi}(u)\}$, denoted by $Q(\boldsymbol{\pi}(u)|\boldsymbol{\pi}^{(r)}(u))$, as

$$(\hat{\pi}_{k0}(u), \hat{\pi}_{k1}(u), \ldots, \hat{\pi}_{kp}(u)) = \max Q(\boldsymbol{\pi}(u)|\boldsymbol{\pi}^{(r)}(u)) \tag{3.32}$$

and

$$\sigma_k^{2(r+1)}(u) = \hat{\sigma}_{k0}^2(u), \tag{3.33}$$

where $\hat{\sigma}_{k0}^2(u)$, for $k = 1, 2, \ldots, K$, is the LPL estimator obtained by maximising the expected value of $\ell^c\{\boldsymbol{\sigma}^2(u)\}$, denoted by $Q(\boldsymbol{\sigma}^2(u)|\boldsymbol{\sigma}^{2(r)}(u))$, as

$$(\hat{\sigma}_{k0}^2(u), \hat{\sigma}_{k1}^2(u), \ldots, \hat{\sigma}_{kp}^2(u)) = \max Q(\boldsymbol{\sigma}^2(u)|\boldsymbol{\sigma}^{2(r)}(u)). \tag{3.34}$$

Repeat the above E- and M-steps until convergence.
At convergence, we obtain $\tilde{\boldsymbol{\theta}}(s_i)$, for $i = 1, 2, \ldots, n$, by interpolating over $\boldsymbol{\theta}^{(R)}(u)$ for $u \in \mathcal{U}$. Finally, we obtain the second-stage estimators of $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}^2$ as $\tilde{\boldsymbol{\pi}}$ and $\tilde{\boldsymbol{\sigma}}^2$, respectively.

Second-Stage: The EM algorithm to maximise (3.18)

Next, given $(\hat{\beta}_{k,a}, \hat{g}_{k,c}(t_c))_{1\le k\le K, 1\le a\le D_1, 1\le c\le D_3}$, $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\sigma}}^2$ and the current or first-stage estimates of $\boldsymbol{\gamma}_{\cdot,j}(s) = (\gamma_{1,j}(s), \gamma_{2,j}(s), \ldots, \gamma_{K,j}(s))$, for $j \neq b$, let $u \in \mathcal{U}$, where $\mathcal{U}$ is the set of grid points in the domain of the covariate s. Define the complete-data log-likelihood version (3.18) as

$$\ell\{\boldsymbol{\gamma}_{\cdot b}(u)\} = \sum_{i=1}^{n}\sum_{k=1}^{K} q_{ik} \log \mathcal{N}\Big(\hat{y}_{ib}|\gamma_{k,b}(u) z_{ib}, \tilde{\sigma}_k^2(s_i)\Big) K_h(s_i - u), \tag{3.35}$$

where $\boldsymbol{\gamma}_{\cdot b}(u) = (\boldsymbol{\gamma}_{1,b}(u), \boldsymbol{\gamma}_{2,b}(u), \ldots, \boldsymbol{\gamma}_{K,b}(u))$ is a vector of the local parameters at local point $u \in \mathcal{U}$ and $\hat{y}_{ib} = y_i - \mathbf{x}_i^\intercal \hat{\boldsymbol{\beta}}_k + \hat{\boldsymbol{\Gamma}}_k(s_i, \mathbf{z}_{i,-b}) + \hat{\boldsymbol{g}}_k(\mathbf{t}_i)$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, are the pseudo responses.
At the $(r+1)^{th}$ iteration of the E-step, we calculate the responsibilities as

$$p_{ik}^{(r+1)}(u) = \frac{\tilde{\pi}_k(s_i)\mathcal{N}(\hat{y}_{ib}|\gamma_{k,b}^{(r)}(u) z_{ib}, \tilde{\sigma}_k^2(s_i))}{\sum_{\ell=1}^{K} \tilde{\pi}_\ell(s_i)\mathcal{N}(\hat{y}_{ib}|\gamma_{k,b}^{(r)}(u) z_{ib}, \tilde{\sigma}_k^2(s_i))}. \tag{3.36}$$

At the $(r+1)^{th}$ iteration of the M-step, update $\boldsymbol{\gamma}_{\cdot b}^{(r)}(u)$ for each local point $u \in \mathcal{U}$ as

$$\gamma_{k,b}^{(r+1)}(u) = \hat{\gamma}_{k,b,0}(u), \tag{3.37}$$

where $\hat{\gamma}_{k,b,0}(u)$, for $k = 1, 2, \ldots, K$, is the LPL estimator obtained by maximising the expected value of $\ell^c\{\boldsymbol{\gamma}_{\cdot b}(u)\}$, denoted by $Q(\boldsymbol{\gamma}_{\cdot b}(u)|\boldsymbol{\gamma}_{\cdot b}^{(r)}(u))$, as

$$(\hat{\gamma}_{k,b,0}(u), \hat{\gamma}_{k,b,1}(u), \ldots, \hat{\gamma}_{k,b,p}(u)) = \max Q(\boldsymbol{\gamma}_{\cdot b}(u)|\boldsymbol{\gamma}_{\cdot B}^{(r)}(u)). \tag{3.38}$$

Repeat the above E- and M-step until convergence.
At convergence, we obtain $\boldsymbol{\gamma}_{\cdot b}(s_i)$, for $i = 1, 2, \ldots, n$, by interpolating over $\boldsymbol{\gamma}_{\cdot b}^{(R)}(u)$ for $u \in \mathcal{U}$. Let $\tilde{\boldsymbol{\gamma}}_{\cdot b}$ be the second-stage estimator of $\boldsymbol{\gamma}_{\cdot b}$. After setting $\tilde{\boldsymbol{\gamma}}_{\cdot b}$ to be the current estimator of $\boldsymbol{\gamma}_{\cdot b}$, we repeat the above procedure to estimate the second-stage estimator of the rest of the varying coefficient functions $\boldsymbol{\gamma}_{\cdot j}$, for $j \neq b$. Finally, let $\tilde{\boldsymbol{\gamma}}$ be the second-stage estimator of $\boldsymbol{\gamma}$.

Second-Stage: The EM algorithm to maximise (3.21)

Finally, given $(\hat{\beta}_{k,a})_{1\leq k\leq K, 1\leq a\leq D_1}$, $\tilde{\boldsymbol{\gamma}}$, $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\sigma}}^2$ and the current estimates of the functions $g_{k,j}(t_j)$, for $j \neq c$, let $u \in \mathcal{U}_c$, where $\mathcal{U}_c$ is the set of grid points in the domain of the covariate t_c. Define the complete-data log-likelihood version (3.21) as

$$\ell\{\boldsymbol{g}_{\cdot c}(u)\} = \sum_{i=1}^{n}\sum_{k=1}^{K} q_{ik} \log \mathcal{N}\Big(\hat{y}_{ir}|g_{k,c}(u), \tilde{\sigma}_k^2(s_i)\Big) K_h(t_{ic} - u), \tag{3.39}$$

where $\boldsymbol{g}_{\cdot c}(u) = (\boldsymbol{g}_{1,c}(u), \boldsymbol{g}_{2,c}(u), \ldots, \boldsymbol{g}_{K,c}(u))$ is the vector of local parameters at the local point $u \in \mathcal{U}_c$ and $\hat{y}_{ic} = y_i - \mathbf{x}_i^\intercal \hat{\boldsymbol{\beta}}_k - \tilde{\boldsymbol{\Gamma}}_k(s_i, \mathbf{z}_i) - \hat{\boldsymbol{g}}_k(\mathbf{t}_{i,-c})$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, are the pseudo responses.
At the $(r+1)^{th}$ iteration of the E-step, we calculate the responsibilities as

$$p_{ik}^{(r+1)}(u) = \frac{\tilde{\pi}_k(s_i)\mathcal{N}(\hat{y}_{ic}|g_{k,c}^{(r)}(u), \tilde{\sigma}_k^2(s_i))}{\sum_{\ell=1}^{K}\tilde{\pi}_\ell(s_i)\mathcal{N}(\hat{y}_{ic}|g_{k,c}^{(r)}(u), \tilde{\sigma}_k^2(s_i))} \tag{3.40}$$

At the $(r+1)^{th}$ iteration of the M-step, update $\boldsymbol{g}_{\cdot c}^{(r)}(u)$ for each local point $u \in \mathcal{U}_c$ as

$$g_{k,c}^{(r+1)}(u) = \hat{g}_{k,c,0}(u), \tag{3.41}$$

where $\hat{g}_{k,c,0}(u)$, for $k = 1, 2, \ldots, K$, is the LPL estimator obtained by maximising the expected value of $\ell^c\{\boldsymbol{g}_{\cdot c}(u)\}$, denoted by $Q(\boldsymbol{g}_{\cdot c}(u)|\boldsymbol{g}_{\cdot c}^{(r)}(u))$, as

$$(\hat{g}_{k,c,0}(u), \hat{g}_{k,c,1}(u), \ldots, \hat{g}_{k,c,p}(u)) = \max Q(\boldsymbol{g}_{\cdot c}(u)|\boldsymbol{g}_{\cdot c}^{(r)}(u)) \tag{3.42}$$

Repeat the above E- and M-step until convergence.
At convergence, we obtain $\boldsymbol{g}_{\cdot c}(t_{ic})$, for $i = 1, 2, \ldots, n$, by interpolating over $\boldsymbol{g}_{\cdot c}^{(R)}(u)$ for $u \in \mathcal{U}_c$.

Let $\tilde{\boldsymbol{g}}_{\cdot c}$ be the second-stage estimator of $\boldsymbol{g}_{\cdot c}$. After setting the $\tilde{\boldsymbol{g}}_{\cdot c}$ to be the current estimator of $\boldsymbol{g}_{\cdot c}$, we repeat the above procedure to obtain the second-stage estimators of the rest of the additive functions $\boldsymbol{g}_{\cdot j}$, for $j \neq c$. Finally, let $\tilde{\boldsymbol{g}}$ be the second-stage estimator of $\boldsymbol{g}$.

Second-Stage: The EM algorithm to maximise (3.22)

Given the second-stage estimators $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\sigma}}^2$, $\tilde{\boldsymbol{\gamma}}$ and $\tilde{\boldsymbol{g}}$, we can improve the parameter estimate $\hat{\boldsymbol{\beta}}$ by maximising (3.22) using the usual EM algorithm to obtain $\tilde{\boldsymbol{\beta}}$.
We refer to the estimators $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\sigma}}^2$, $\tilde{\boldsymbol{\gamma}}$, $\tilde{\boldsymbol{g}}$ and $\tilde{\boldsymbol{\beta}}$ as the SBLPL estimators. Once again, as in subsection 1.2.1, notice that, at the E-steps (3.30), (3.36) and (3.40), the responsibilities $p_{ik}(u)$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, are calculated at different local points $u \in \mathcal{U}$. This implies that at each local point $u \in \mathcal{U}$, we have an estimate of the latent variable q_{ik} given by $p_{ik}(u)$. These local responsibilities are not guaranteed to match across all local points. In the event of a mismatch we have label-switching which is characterised by discontinuous jumps at the local points where the component labels have switched (see Figure 3.1a). In addition, the non-parametric function estimates are wiggly and non-smooth. Consequently, the estimated functions are not very useful in practice.
In section 3.2, we will consider, in detail, the nature and origin of the label-switching phenomenon studied in this thesis.

3.1.2 One-step backfitting LPL (OSBLPL) estimation procedure

To estimate the SPGMRVPs model (1.17), adopting a similar estimation strategy proposed by Carroll et al. [1997] for efficiently estimating a semi-parametric model, Huang and Yao [2012] proposed one-step backfitting kernel (OSBK) estimation via the EM algorithm. The OSBK estimation procedure is a three-stage estimation procedure. In the first-stage, we estimate both the global (parametric) and local (non-parametric) parameters non-parametrically using local kernel likelihood (LKL) estimation, a special case of LPL estimation with $p = 0$. In the second-stage, given the first-stage estimates of the non-parametric terms, estimate the global parameters using the usual parametric likelihood procedure via the EM algorithm. To improve the first-stage estimate of the non-parametric terms, Huang and Yao [2012] proposed an additional (third) stage in which, given the second-stage estimates of the parametric term, we re-estimate the non-parametric term, hence backfitting.
The OBSK estimation procedure gained popularity for estimating semi-parametric GMRs of the form (1.1). Xiang and Yao [2016] proposed to use the OSBK procedure to estimate the SPGM-NPRs model (1.4). Recently, Xiang and Yao [2020] proposed the OSBK procedure to estimate the semi-parametric mixture of regressions with varying single index models (SPGMRVSIPs). The OSBK procedure is a simple extension of the one-step local-linear (OSLL) procedure of

Carroll et al. [1997] in which an additional stage is included to improve the first-stage estimator of the non-parametric term. The OSLL procedure estimates the non-parametric terms using local-linear estimators (LLE), that is LPL estimators with $p = 1$, hence OSLL. Huang and Yao [2012] gave theoretical and empirical evidence to show that the OSBK estimators are more efficient than the OSK estimators.

In this thesis, we propose the one-step backfitting LPL (OSBLPL) estimation procedure to estimate model (1.1). The OSBLPL procedure is an extension of the OSBK procedure and the OSLL procedure. The OSBLPL estimators combine the efficiency gain from backfitting and the reduction in bias achieved by LPL estimators (see subsection 2.2.2).

The proposed procedure proceeds as follows. In the first-stage, the component regression coefficients, $\boldsymbol{\beta}_k$, for $k = 1, 2, \ldots, K$, in model (1.1) are assumed to be non-parametric functions of the covariate s, denoted $\boldsymbol{\beta}_k(s)$, for $k = 1, 2, \ldots, K$, thus rendering all the estimable terms of the model to be non-parametric. The LPL procedure, via the EM algorithm, is then applied to estimate the resulting model. In the second-stage, given the first-stage estimators of the non-parametric terms, we estimate the parametric term, $\boldsymbol{\beta}_k$, for $k = 1, 2, \ldots, K$, using the usual parametric maximum likelihood estimation via the EM algorithm. In order to improve the first-stage estimators of the non-parametric terms, given the second-stage estimators of the $\boldsymbol{\beta}_k$'s, in the third-stage, we re-estimate the non-parametric terms using the LPL procedure.

Estimation procedure We now give details of each stage of the estimation procedure. Let $\mathbf{x}^* = (\mathbf{x}, \mathbf{z})^\intercal$. In the first-stage, we maximise the log LPL function

$$\ell\{\boldsymbol{\theta}_1(u)\} = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k(u)\mathcal{N}\big(y_i|\mathbf{x}_i^{*\intercal}\boldsymbol{\eta}_k(u) + \boldsymbol{g}_k^{(0)}(\mathbf{t}_i), \sigma_k^2(u)\big)\right] \times K_h(s_i - u), \tag{3.43}$$

where $\boldsymbol{\theta}_1(u) = (\boldsymbol{\pi}(u), \boldsymbol{\sigma}^2(u), \boldsymbol{\eta}(u)) = (\pi_1(u), \pi_2(u), \ldots, \pi_K(u); \sigma_1^2(u), \sigma_2^2(u), \ldots, \sigma_K^2(u);$ $\boldsymbol{\eta}_1(u), \boldsymbol{\eta}_2(u), \ldots, \boldsymbol{\eta}_K(u))$ with $\boldsymbol{\eta}_k(u) = (\boldsymbol{\beta}_k(u), \boldsymbol{\gamma}_k(u))$ is a vector of all the local parameters at grid point u, where $u \in \mathcal{U}$ is in the domain of the covariate s. The term $\boldsymbol{g}_k^{(0)}(\mathbf{t})$ represents the initial estimates of the additive functions defined as $\boldsymbol{g}_k^{(0)}(\mathbf{t}) = \sum_{j=1}^{D_3} g_k^{(0)}(t_j)$.

Let $\hat{\boldsymbol{\theta}}_1(u)$ be the resulting local parameter estimates for $u \in \mathcal{U}$. To obtain $\hat{\boldsymbol{\theta}}_1(s_i)$, for $i = 1, 2, \ldots, n$, we use linear interpolation.

Given the estimate $\hat{\boldsymbol{\theta}}_1(\mathbf{s})$, where $\mathbf{s} = (s_1, s_2, \ldots, s_n)^\intercal$, and $g_k^{(0)}(t_j) : j \neq c$, the additive function

$g_k(t_c)$, is estimated by maximising the log LPL function

$$\ell\{\boldsymbol{g}_1(u)\} = \sum_{i=1}^{n} \log\Bigg[\sum_{k=1}^{K} \hat{\pi}_k(s_i)\mathcal{N}\big(y_i|\mathbf{x}_i^{*\intercal}\hat{\boldsymbol{\eta}}_k(s_i) + \boldsymbol{g}_k(\mathbf{t}_{i,-r}, u), \hat{\sigma}_k^2(s_i)\big)\Bigg] \times K_h(t_{ir} - u) \tag{3.44}$$

where $\boldsymbol{g}_1(u) = (g_1(u), g_2(u), \ldots, g_K(u))$ holds all the local parameters at grid point u, for $u \in \mathcal{U}_c$ in the domain of the covariate t_c. The term $g_k(\mathbf{t}_{i,-c}, u)$ is defined as $g_k(\mathbf{t}_{-c}, u) = g_k(u) + \sum_{j\neq c}^{D_3} \hat{g}_k(t_j)$ with $\mathbf{t}_{-c}$ being the vector of covariates $\mathbf{t}$ excluding the covariate t_c and the $\hat{g}_k(t_j)'s$ are the current estimates of the additive functions $g_k(t_j) : j \neq c$.
Let $\hat{\boldsymbol{g}}_1(u)$, be the resulting local parameter estimates for $u \in \mathcal{U}_c$. To obtain $\hat{\boldsymbol{g}}_1(t_{ic})$, for $i = 1, 2, \ldots, n$, we use linear interpolation. We repeat the above procedure to estimate the rest of the additive functions $g_k(t_j) : j \neq c$ using a backfitting approach.
Let $\hat{\boldsymbol{\theta}}_1 = (\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\sigma}}^2, \hat{\boldsymbol{\eta}}, \hat{\boldsymbol{g}})$ be the resulting first-stage estimators of $\boldsymbol{\theta}_1 = (\boldsymbol{\pi}, \boldsymbol{\sigma}^2, \boldsymbol{\eta}, \boldsymbol{g})$. In the second-stage, given $\hat{\boldsymbol{\theta}}_1$, we estimate the global parameter $\boldsymbol{\beta}$ by maximising the log-likelihood function

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{n} \log\Bigg[\sum_{k=1}^{K} \hat{\pi}_k(s_i)\mathcal{N}\big(y_i|\mathbf{x}_i^{\intercal}\boldsymbol{\beta}_k + \mathbf{z}_i^{\intercal}\hat{\boldsymbol{\gamma}}_k(s_i) + \sum_{c=1}^{D_3} \hat{g}_k(t_{ic}), \hat{\sigma}_k^2(s_i)\big)\Bigg]. \tag{3.45}$$

Let $\tilde{\boldsymbol{\beta}}$ be the resulting second-stage estimator of the global parameter $\boldsymbol{\beta}$. Given $\tilde{\boldsymbol{\beta}}$, to improve the first-stage non-parametric estimates $(\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\sigma}}^2, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{g}})$, we maximise the log LPL function

$$\ell(\boldsymbol{\theta}_2(u)) = \sum_{i=1}^{n} \log\Bigg[\sum_{k=1}^{K} \pi_k(u)\mathcal{N}\big(y_i|\mathbf{x}_i^{\intercal}\tilde{\boldsymbol{\beta}}_k + \mathbf{z}_i^{\intercal}\boldsymbol{\gamma}_k(u) + \hat{g}_k(\mathbf{t}_i), \sigma_k^2(u)\big)\Bigg] \times K_h(s_i - u), \tag{3.46}$$

where $\boldsymbol{\theta}_2(u) = (\boldsymbol{\pi}(u), \boldsymbol{\sigma}^2(u), \boldsymbol{\gamma}(u)) = (\pi_1(u), \ldots, \pi_K(u); \sigma_1^2(u), \ldots, \sigma_K^2(u); \boldsymbol{\gamma}_1(u), \ldots, \boldsymbol{\gamma}_K(u))$ is a vector of all the local parameters at grid point u. The term $\hat{g}_k(\mathbf{t})$ is similarly defined as above. As before, we use linear interpolation to obtain $\tilde{\boldsymbol{\theta}}_2(s_i)$, for $i = 1, 2, \ldots, n$.
Given the estimates $\tilde{\boldsymbol{\theta}}_2(\mathbf{s})$ and $\hat{g}_k(t_j) : j \neq c$, the additive function $g_k(t_c)$, is estimated by maximising the log LPL function

$$\ell\{\boldsymbol{g}_2(u)\} = \sum_{i=1}^{n} \log\Bigg[\sum_{k=1}^{K} \tilde{\pi}_k(s_i)\mathcal{N}\big(y_i|\mathbf{x}_i^{\intercal}\tilde{\boldsymbol{\beta}}_k + \mathbf{z}_i^{\intercal}\tilde{\boldsymbol{\gamma}}_k(u) + g_k(\mathbf{t}_{i,-c}, u), \tilde{\sigma}_k^2(s_i)\big)\Bigg] \times K_h(t_{ic} - u), \tag{3.47}$$

where $\boldsymbol{g}_2(u) = (g_1(u), g_2(u), \ldots, g_K(u))$ holds all the local parameters at grid point u, for $u \in \mathcal{U}_c$ in the domain of the covariate t_c. The definition of the log LPL function (3.47) is similar to the log LPL function (3.44).
Let $(\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\sigma}}^2, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{g}})$ be the resulting improved estimates. We refer to the estimators $(\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\sigma}}^2, \tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{g}})$ as the OSBLPL estimators.

Fitting algorithm We now give details of the multi-stage EM algorithm that can be used to obtain the OSBLPL estimators. As with the SBLPL-EM algorithm, the OSBLPL-EM algorithm may be subject to label-switching.
Let $\{(s_i, \mathbf{x}_i, \mathbf{z}_i, \mathbf{t}_i, y_i, \mathbf{q}_i) : i = 1, 2, \ldots, n\}$ be the complete-data where $\mathbf{q}$ is the latent variable as defined before. Next, let $\ell^c\{\boldsymbol{\theta}_1(u)\}$, $\ell^c\{\boldsymbol{g}_1(u)\}$, $\ell^c(\boldsymbol{\beta})$, $\ell^c\{\boldsymbol{\theta}_2(u)\}$ and $\ell^c\{\boldsymbol{g}_2(u)\}$ denote the corresponding complete-data likelihood functions. Finally, to initialise the following multi-stage EM algorithm, let $\boldsymbol{\theta}^{(0)} = (\boldsymbol{\pi}^{(0)}, \boldsymbol{\sigma}^{2(0)}, \boldsymbol{\beta}^{(0)}, \boldsymbol{\gamma}^{(0)}, \boldsymbol{g}^{(0)})$ be the initial value of the vector of the parametric and non-parametric terms of the model.

First-Stage: The EM algorithm to maximise (3.43)

Given $\boldsymbol{\theta}_1^{(r)}(u)$, for $u \in \mathcal{U}$, from the r^{th} iteration,
E-step (1): at the $(r+1)^{th}$ iteration, we calculate the expected value of $\ell^c\{\boldsymbol{\theta}_1(u)\}$ which reduces to calculating the conditional expected value of the latent variable q_{ik}, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, as

$$p_{ik}^{(r+1)}(u) = \frac{\pi_k^{(r)}(u)\mathcal{N}(y_i|\mathbf{x}_i^{*\intercal}\boldsymbol{\eta}_k^{(r)}(u) + g_k^{(0)}(\mathbf{t}_i), \sigma_k^{2(r)}(u))}{\sum_{\ell=1}^{K} \pi_\ell^{(r)}(u)\mathcal{N}(y_i|\mathbf{x}_i^{*\intercal}\boldsymbol{\eta}_\ell^{(r)}(u) + g_\ell^{(0)}(\mathbf{t}_i), \sigma_\ell^{2(r)}(u))}, \tag{3.48}$$

for $u \in \mathcal{U}$, where $\mathcal{U}$ is the set of local grid points in the domain of the covariate s.
M-step (1): At the $(r+1)^{th}$ iteration, we update $\boldsymbol{\theta}_1^{(r)}(u)$ by maximising the expected value of $\ell^c\{\boldsymbol{\theta}_1(u)\}$, denoted by $Q(\boldsymbol{\theta}_1(u)|\boldsymbol{\theta}_1^{(r)}(u))$, as

$$\boldsymbol{\theta}_1^{(r+1)}(u) = \max Q(\boldsymbol{\theta}_1(u)|\boldsymbol{\theta}_1^{(r)}(u)). \tag{3.49}$$

Repeat the above E- and M-steps until convergence.
Calculate $\boldsymbol{\theta}_1^{(r)}(s_i)$, for $i = 1, 2, \ldots, n$, by linearly interpolating over $\boldsymbol{\theta}_1^{(R)}(u)$, where $\boldsymbol{\theta}_1^{(R)}(u)$ is the estimate of $\boldsymbol{\theta}_1(u)$ at convergence of the above EM algorithm. Let $\hat{\boldsymbol{\theta}}_1(\mathbf{s})$, where $\mathbf{s} = (s_1, s_2, \ldots, s_n)^\intercal$, be the resulting estimate.

First-Stage: The EM algorithm to maximise (3.44)

Given $\hat{\boldsymbol{\theta}}_1(\mathbf{s})$ and $\boldsymbol{g}_1^{(r)}(u)$, for $u \in \mathcal{U}_c$, from the r^{th} iteration,
E-step (2): at the $(r+1)^{th}$ iteration, we calculate the responsibilities as

$$p_{ik}^{(r+1)}(u) = \frac{\hat{\pi}_k(s_i)\mathcal{N}\big(y_i|\mathbf{x}_i^{*\intercal}\hat{\boldsymbol{\eta}}_k(s_i) + g_k^{(r)}(\mathbf{t}_{i,-c}, u), \hat{\sigma}_k^2(s_i)\big)}{\sum_{\ell=1}^{K}\hat{\pi}_\ell(s_i)\mathcal{N}\big(y_i|\mathbf{x}_i^{*\intercal}\hat{\boldsymbol{\eta}}_\ell(s_i) + g_\ell^{(r)}(\mathbf{t}_{i,-c}, u), \hat{\sigma}_\ell^2(s_i)\big)}, \tag{3.50}$$

for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, $u \in \mathcal{U}_c$.
M-step (2): At the $(r+1)^{th}$ iteration, we update $\boldsymbol{g}_1^{(r)}(u)$, for $u \in \mathcal{U}_c$, by maximising the expected value of $\ell^c\{\boldsymbol{g}_1(u)\}$, denoted by $Q(\boldsymbol{g}_1(u)|\boldsymbol{g}_1^{(r)}(u))$, as

$$\boldsymbol{g}_1^{(r+1)}(u) = \max Q(\boldsymbol{g}_1(u)|\boldsymbol{g}_1^{(r)}(u)) \tag{3.51}$$

Repeat the above E- and M-steps until convergence.
Calculate $\hat{\boldsymbol{g}}_1(t_{ic})$, for $i = 1, 2, \ldots, n$, by linearly interpolating over $\boldsymbol{g}_1^{(R)}(u)$, where $\boldsymbol{g}_1^{(R)}(u)$ is the estimate of $\boldsymbol{g}_1(u)$ at convergence. Let $\hat{g}_k(t_{ic})$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, be the resulting estimate.
Using a backfitting approach, repeat the above E- and M-step to estimate the rest of the functions $g_k(t_j)$, for $j \neq c$ and $k = 1, 2, \ldots, K$.

Second-Stage: The EM algorithm to maximise (3.45)

Given $\hat{\boldsymbol{\theta}}_1$ and $\boldsymbol{\beta}^{(r)}$, from the r^{th} iteration, **E-step:** at the $(r+1)^{th}$ iteration, we calculate the responsibilities as

$$p_{ik}^{(r+1)} = \frac{\hat{\pi}_k(s_i)\mathcal{N}\big(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_k^{(r)} + \mathbf{z}_i^\intercal\hat{\boldsymbol{\gamma}}_k(s_i) + \sum_{c=2}^{D_3}\hat{g}_k(t_{ic}), \hat{\sigma}_k^2(s_i)\big)}{\sum_{\ell=1}^{K}\hat{\pi}_\ell(s_i)\mathcal{N}\big(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_\ell^{(r)} + \mathbf{z}_i^\intercal\hat{\boldsymbol{\gamma}}_\ell(s_i) + \sum_{c=2}^{D_3}\hat{g}_\ell(t_{ic}), \hat{\sigma}_\ell^2(s_i)\big)}, \tag{3.52}$$

for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$.
M-step: At the $(r+1)^{th}$ iteration, we update $\boldsymbol{\beta}^{(r)}$, as

$$\boldsymbol{\beta}^{(r+1)} = \max Q(\boldsymbol{\beta}|\boldsymbol{\beta}^{(r)}). \tag{3.53}$$

Repeat the above E- and M-steps until convergence.
Let $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}^{(R)}$, be the resulting estimate of $\boldsymbol{\beta}$ at convergence, where $\boldsymbol{\beta}^{(R)}$ denotes the estimate of $\boldsymbol{\beta}$ at convergence.

Third-Stage: The EM algorithm to maximise (3.46)

Given $\tilde{\boldsymbol{\beta}}$ and $\boldsymbol{\theta}_2^{(r)}(u)$, for all $u \in \mathcal{U}$, from the r^{th} iteration,
E-step (1): at the $(r+1)^{th}$ iteration, we calculate the responsibilities as

$$p_{ik}^{(r+1)}(u) = \frac{\pi_k(u)\mathcal{N}\big(y_i|\mathbf{x}_i^\intercal\tilde{\boldsymbol{\beta}}_k + \mathbf{z}_i^\intercal\boldsymbol{\gamma}_k(u) + \hat{g}_k(\mathbf{t}_i), \sigma_k^2(u)\big)}{\sum_{\ell=1}^K \pi_\ell(u)\mathcal{N}\big(y_i|\mathbf{x}_i^\intercal\tilde{\boldsymbol{\beta}}_\ell + \mathbf{z}_i^\intercal\boldsymbol{\gamma}_\ell(u) + \hat{g}_\ell(\mathbf{t}_i), \sigma_\ell^2(u)\big)}, \tag{3.54}$$

for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$ and $u \in \mathcal{U}$.
M-step (1): At the $(r+1)^{th}$ iteration, we update $\boldsymbol{\theta}_2^{(r)}(u)$ as

$$\boldsymbol{\theta}_2^{(r+1)}(u) = \max Q(\boldsymbol{\theta}_2^{(r)}(u)|\boldsymbol{\theta}_2(u)). \tag{3.55}$$

Repeat the above E- and M-steps until convergence.
Calculate $\boldsymbol{\theta}_2^{(r)}(s_i)$, for $i = 1, 2, \ldots, n$, by linearly interpolating over $\boldsymbol{\theta}_2^{(R)}(u)$, where $\boldsymbol{\theta}_2^{(R)}(u)$ is the estimate of $\boldsymbol{\theta}_2(u)$ at convergence. Let $\tilde{\boldsymbol{\theta}}_2(\mathbf{s})$ be the resulting estimate.

Third-Stage: The EM algorithm to maximise (3.47)

Given $\tilde{\boldsymbol{\theta}}_2(\mathbf{s})$, $\tilde{\boldsymbol{\beta}}$ and $\mathbf{g}_2^{(r)}(u)$, for $u \in \mathcal{U}_c$, from the r^{th} iteration,
E-step (2): at the $(r+1)^{th}$ iteration, we calculate the responsibilities as

$$p_{ik}^{(r+1)}(u) = \frac{\tilde{\pi}_k(s_i)\mathcal{N}(y_i|\mathbf{x}_i^\intercal\tilde{\boldsymbol{\beta}}_k + \mathbf{z}_i^\intercal\tilde{\boldsymbol{\gamma}}_k(u) + g_k(\mathbf{t}_{i,-c}, u), \tilde{\sigma}_k^2(s_i)\big)}{\sum_{\ell=1}^K \tilde{\pi}_\ell(s_i)\mathcal{N}\big(y_i|\mathbf{x}_i^\intercal\tilde{\boldsymbol{\beta}}_\ell + \mathbf{z}_i^\intercal\tilde{\boldsymbol{\gamma}}_\ell(u) + g_\ell(\mathbf{t}_{i,-c}, u), \tilde{\sigma}_\ell^2(s_i)\big)}, \tag{3.56}$$

for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, $u \in \mathcal{U}_c$
M-step (2): At the $(r+1)^{th}$ iteration, we update each $\boldsymbol{g}_2^{(r)}(u)$, as

$$\boldsymbol{g}_2^{(r+1)}(u) = \max Q(\boldsymbol{g}_2(u)|\boldsymbol{g}_2^{(r)}(u)). \tag{3.57}$$

Repeat the above E- and M-steps until convergence.
Calculate $\tilde{\boldsymbol{g}}_2(t_{ic})$, for $i = 1, 2, \ldots, n$, by linearly interpolating over $\boldsymbol{g}_2^{(R)}(u)$, for $u \in \mathcal{U}_c$, where $\boldsymbol{g}_2^{(R)}(u)$ is the estimate of $\boldsymbol{g}_2(u)$ at convergence. Let $\tilde{g}_k(t_{ic})$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, be the resulting estimate. Using a backfitting approach, repeat the above E- and M-step to estimate the rest of the functions $g_k(t_j)$, for $j \neq c$ and $k = 1, 2, \ldots, K$.
Finally, the estimators $(\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\sigma}}^2, \tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{g}})$ are referred to as the OSBLPL estimators.

3.2 Label-switching

In section 1.2, we saw that a naive local-likelihood estimation, via the EM algorithm, of the non-parametric functions of model (1.1) may result in label-switching. In the above-mentioned section, we gave a brief account of this phenomenon. In this section, we provide a comprehensive discussion of label-switching. This section can be viewed as a continuation of section 1.2.

3.2.1 A formal definition of the problem

Let $\hat{\mathbf{m}} = (\hat{\boldsymbol{m}}(u_1), \hat{\boldsymbol{m}}(u_2), \ldots, \hat{\boldsymbol{m}}(u_N))$ be a vector of all the local parameter estimates, where $\hat{\boldsymbol{m}}(u) = (\hat{m}_1(u), \hat{m}_2(u), \ldots, \hat{m}_K(u))$, and $\boldsymbol{\phi}_j = (\boldsymbol{\phi}(1), \boldsymbol{\phi}(2), \ldots, \boldsymbol{\phi}(K))$ be the j^{th} permutation of the labels $\{1, 2, \ldots, K\}$ for $j = 1, 2, \ldots, K!$ permutations. For $j = 1$, we get the identity permutation $\boldsymbol{\phi}_1 = (1, 2, \ldots, K)$. For any $u \in \mathcal{U}$, define the corresponding permutation of $\hat{\boldsymbol{m}}(u)$ as

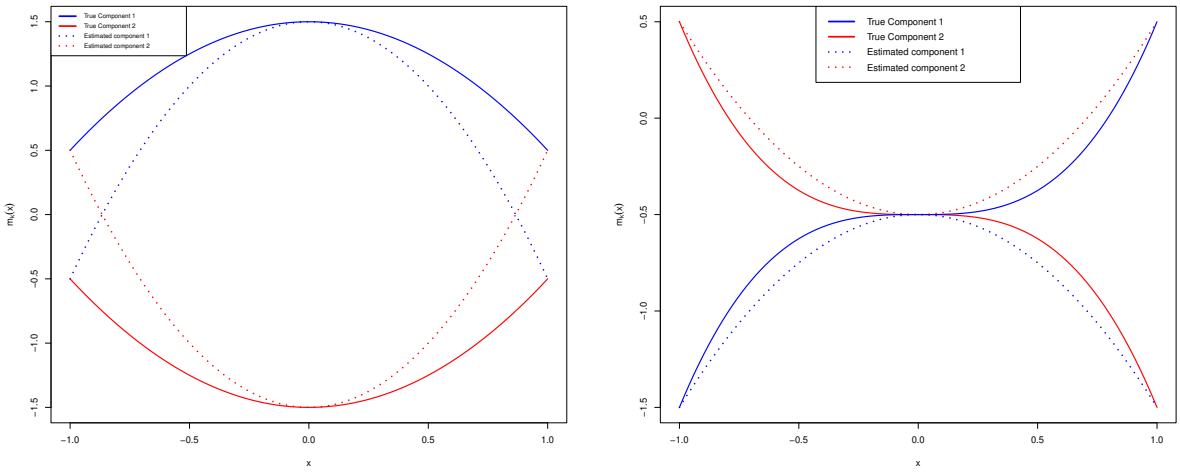
$$\hat{\boldsymbol{m}}^{\boldsymbol{\phi}_j}(u) = (\hat{m}_{\boldsymbol{\phi}(1)}(u), \hat{m}_{\boldsymbol{\phi}(2)}(u), \ldots, \hat{m}_{\boldsymbol{\phi}(K)}(u)). \tag{3.58}$$

Due to the misalignment of the local responsibilities at different local points, the labels on the local parameter estimates can assume any one of the $K!$ possible labelling. As before, consider any two distinct local grid points u_1 and u_2, where there is a misalignment in the corresponding local responsibilities. The respective vectors of the local parameter estimates will be given by $\hat{\boldsymbol{m}}^{\boldsymbol{\phi}_j}(u_1)$ and $\hat{\boldsymbol{m}}^{\boldsymbol{\phi}_{j^*}}(u_2)$, where $j \neq j^* \in \{1, 2, \ldots, K!\}$. We refer to this problem as the label-switching problem. The root of this problem comes from the fact that for any given local point u

$$\ell[\boldsymbol{\theta}^{\phi_j}(u)] = \ell[\boldsymbol{\theta}(u)], \tag{3.59}$$

for any permutation $j = 1, 2, \ldots, K!$ (see Stephens [2000] and Yao [2012]).

Another way to view the label-switching problem is in a model-based clustering context. To simplify the explanation, consider the $K = 2$ component case. Recall that p_{ik} gives the probability that the i^{th} observation belongs to the k^{th} component. In model-based clustering, the components of the mixture model are the clusters. The i^{th} observation is assigned to the k^{th} cluster (component) if $p_{ik} > 0.5$. Consider again two distinct local grid points u_1 and u_2, if $p_{ik}(u_1) > p_{ik}(u_2)$, then the same observation is assigned to two different clusters at two different local grid points. This is in essence a swapping of the labels attached to the mixture components at grid point u_1 and u_2. Translating it back to the problem of estimating the local parameters $\mathbf{m}(u)$, for $K = 2$, this implies that the local parameters at grid point u_1 and u_2 are given by $\hat{\mathbf{m}}(u_1) = (\hat{m}_1(u_1), \hat{m}_2(u_1))^\intercal$ and $\hat{\mathbf{m}}(u_2) = (\hat{m}_2(u_2), \hat{m}_1(u_2))^\intercal$, respectively. If this misalignment occurs at most of the local grid points, the resulting component regression functions



(a) (b)

Figure 3.1: Label switching problem: (a) A $K = 2$ component case showing the true component regression functions (solid curves). The dotted curves are the fitted component regression functions at three local grid points $-1, 0$ and 1 which shows that there was a switch at grid points -1 and 1. (b) A $K = 2$ component case where the true component regression functions (solid curves) are intersecting and the estimated component regression functions (dotted curves) have a switch at grid point 1.

will exhibit discontinuous jumps as shown by the dotted curves in Figure 3.1a. Moreover, these functions will be wiggly and non-smooth. For example, see the simulation results in chapter 4 and chapter 5.

Figure 3.1a shows a simple example of a $K = 2$ component mixture of non-parametric regressions where the regression function of one component is consistently above that of the other component (given by the solid black curves). Consider maximising the local-likelihood functions at three local points $u = -1, 0$ and 1. There are $(2!)^3 = 8$ possible configurations of the component regression functions when we join the local parameter estimates at the local points. Figure 3.1a gives two such configurations (the true configuration given by the solid curves and the label-switched configuration given by the dotted curves). Note that only two of these configurations will give the correct component regression functions. In the first configuration, the labels on all the local parameter estimates are given by the permutation $\boldsymbol{\phi}_1 = (1, 2)$. Whereas, in the second configuration, the labels on all the local parameters are given by the permutation $\boldsymbol{\phi}_2 = (2, 1)$. As a result, there is $0.75(= 1 - 2/8)$ probability that the naive algorithm (subsection 1.2.1) will result in label-switching. This probability is approximately equal to $1(= 1 - 2/(K!)^3)$ for $K > 2$. Thus, the naive algorithm does not work. Next, we attempt to justify why the local responsibilities should be aligned across the local points.

As discussed in subsection 2.1.1, in order to estimate a mixture model as well as use it in prac-

tice, the mixture model must be identifiable. Briefly, any two mixture models will be considered equal if and only if their corresponding parameters are equal. Mathematically, there is a one-to-one mapping between the parameters and the model. The following argument is based on this notion of identifiability. Note that the set of local responsibilities, say $\{p_{ik}(u) : i = 1, 2, \ldots, n\}$, is a posterior probability distribution of the latent variable q. Suppose that $g(q|t, y, \boldsymbol{\theta})$, the theoretical posterior distribution of q is of interest. Then $g(q|x, y, \boldsymbol{\theta})$ can be incorporated as part of the parameter space. If the underlying mixture model is identifiable, then any two posterior distributions $g_1(q|x, y, \boldsymbol{\theta})$ and $g_2(q|x, y, \boldsymbol{\theta})$ are equal. It follows that, for a given random sample $\{(s_i, y_i) : i = 1, 2, \ldots, n\}$, any two estimates of g, say $\{p_{ik}(u_1) : i = 1, 2, \ldots, n\}$ and $\{p_{ik}(u_2) : i = 1, 2, \ldots, n\}$, must be equal.

3.2.2 Similarity to the Bayesian label-switching

The label-switching problem studied in this thesis is similar to the one encountered when estimating Bayesian parametric mixture models using Markov Chain Monte Carlo (MCMC) procedures to sample from the mixture posterior distribution (see Stephens [2000]). The labels attached to the mixture components may switch across the different sample points during MCMC sampling. On the other hand, the label-switching phenomenon we are concerned with in this thesis arises when estimating a mixture model using maximum likelihood via the EM algorithm, where some of the parameters are non-parametric functions of the covariates. To estimate the non-parametric functions, we have to define a local-likelihood function for each local grid point on the domain of its covariate. If we maximise each local-likelihood function independent of the others, the labels attached to the mixture components may switch across the different local grid points during the iterative EM estimation process.

3.2.3 The origin of the problem

In order to obtain reliable estimates of the component non-parametric functions, we need to address the label-switching problem. A simple approach, usually employed to address label-switching in Bayesian mixture modelling, is to make use of an order constraint on the local parameters at each local point. For instance, the constraint

$$m_1(u) < m_2(u) < \cdots < m_K(u) \tag{3.60}$$

can be applied for $u \in \mathcal{U}$. However, this only works if there are no intersections between the component non-parametric functions (see Figure 3.1a). For more complex mixture structures, as shown in Figure 3.1b, the order constraint approach will not work. Thus, for such mixture structures, we need to develop effective approaches to address the label-switching problem.

As already mentioned in section 1.2, the form of label-switching problem studied in this thesis was first mentioned by Huang [2009] and subsequently by Huang and Yao [2012] and Huang et al. [2013]. The authors proposed a modified EM algorithm, referred to as the effective EM algorithm, to address the problem. The effective EM algorithm replaces the local E-steps with a single (global) E-step. That is, $p_{ik}(u_1) = p_{ik}(u_2)$, $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$. In other words, the responsibilities are independent of the local points. In essence, the sets of local responsibilities $\{p_{ik}(u) : i = 1, 2, \ldots, n; k = 1, 2, \ldots, K\}$, for $u \in \mathcal{U}$, are replaced by a single (global) set of responsibilities $\{p_{ik} : i = 1, 2, \ldots, n; k = 1, 2, \ldots, K\}$. This implies that all the local M-steps make use of the same common set of responsibilities.
The effective EM algorithm is summarised below, followed by a brief discussion of its rationale and some aspects of the algorithm.

The effective EM algorithm

On the E-step, the global responsibilities are calculated as follows

$$p_{ik}^{(r+1)} = \frac{\pi_k^{(r)}(s_i)\mathcal{N}\big(y_i|m_k^{(r)}(s_i), \sigma_k^{2(r)}(s_i)\big)}{\sum_{\ell=1}^{K} \pi_k^{(r)}(s_i)\mathcal{N}\big(y_i|m_k^{(r)}(s_i), \sigma_k^{2(r)}(s_i)\big)}. \tag{3.61}$$

Substitute q_{ik} in the complete-data local-likelihood $\ell^c\{\boldsymbol{\theta}(u)\}$ (1.24) with $p_{ik}^{(r+1)}$ (3.61) to obtain

$$Q\{\boldsymbol{\theta}(u)|\boldsymbol{\theta}^{(r)}(u)\} = \sum_{i=1}^{n}\sum_{k=1}^{K} p_{ik}^{(r+1)}\Big[\log\{\pi_k(u)\} + \log\mathcal{N}\big(y_i|m_k(u), \sigma_k^2(u)\big)\Big] \times K_h(s_i - u). \tag{3.62}$$

On the M-step, for $u \in \mathcal{U}$, we maximise (3.62), to obtain

$$\pi_k^{(r+1)}(u) = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)} K_h(s_i - u)}{\sum_{i=1}^{n} K_h(s_i - u)}, \tag{3.63}$$

$$m_k^{(r+1)}(u) = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)} K_h(s_i - u) y_i}{\sum_{i=1}^{n} p_{ik}^{(r+1)} K_h(s_i - u)}, \tag{3.64}$$

$$\sigma_k^{2(r+1)}(u) = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)} K_h(s_i - u)[y_i - m_k^{(r+1)}(u)]^2}{\sum_{i=1}^{n} p_{ik}^{(r+1)} K_h(s_i - u)}. \tag{3.65}$$

Obtain $\pi_k^{(r+1)}(s_i)$, $\sigma_k^{2(r+1)}(s_i)$ and $m_k^{(r+1)}(s_i)$, for $i = 1, 2, \ldots, n$, by interpolating over $\pi_k^{(r+1)}(u)$, $\sigma_k^{2(r+1)}(u)$ and $m_k^{(r+1)}(u)$ for all $u \in \mathcal{U}$, respectively. Using $\pi_k^{(r+1)}(s_i)$, $\sigma_k^{2(r+1)}(s_i)$ and $m_k^{(r+1)}(s_i)$, for $i = 1, 2, \ldots, n$, return to the global E-step and repeat until convergence.

The effective algorithm is based on the following simple idea. In order to ensure or guarantee that the local responsibilities are aligned, we must explicitly define a common set of responsibilities and enforce it at each local M-step. This algorithm is easy to implement and computationally efficient, mainly due to the fact that the complete-data local-likelihood functions (1.24) can be maximised simultaneously. Moreover, even though the algorithm is not maximising the local-likelihood functions $\ell\{\boldsymbol{\theta}(u)\}$ (1.23) but their complete-data versions $\ell^c\{\boldsymbol{\theta}(u)\}$ (1.24), it still preserves the desirable ascent property, although in an asymptotic sense, as shown in Huang et al. [2013].

For practical and theoretical reasons, the authors made use of local-constant estimators (LCEs) (LPL estimators with $p = 0$) to estimate the non-parametric terms. The practical reason is that the LPL estimators of the mixing proportion functions $\pi_k(s)$ do not have closed form expressions for $p > 0$. Besides, even if the closed form expressions were available, the estimated values of $\pi_k(s)$ are not guaranteed to satisfy the condition $0 < \pi_k(s) < 1$ for $p > 0$. This also applies to the component variance functions $\sigma_k^2(s)$. The LPL estimator, with $p > 0$, of $\sigma_k^2(s)$ is not guaranteed to satisfy the constraint $0 < \sigma_k^2(s) < \infty$ for all s. However, this can be remedied by applying an appropriate transformation to force the values $\pi_k(s)$ and $\sigma_k^2(s)$ on to the interval $(0, 1)$ and $(0, \infty)$, respectively. The theoretical reason is that they allow for a much more convenient way to study the asymptotic properties of the estimators. As argued in section 2.2.2, in order to maintain an appropriate balance between the bias and variance of an LPL estimator, the degree of the polynomial p must be chosen carefully. Moreover, as mentioned, in practice there is a preference for $p = 1$, which results in local-linear estimators (LLEs). In chapter 5 of this thesis, we present examples in which only the CRFs $m_k(s)$ are estimated using LLEs. These examples demonstrate the superior performance of the LLEs compared with the LCEs. Future research should explore the extension of LLEs and LPL estimators, in general, for estimating $\pi_k(s)$ and $\sigma_k^2(s)$.

Chapter 4

Objective-based approach to label-switching

From subsection 3.2.3, we saw that the effective EM-type algorithm of Huang et al. [2013] does not directly maximise the local-likelihood functions $\ell\{\boldsymbol{\theta}(u)\}$ (1.23) but their complete-data versions $\ell^c\{\boldsymbol{\theta}(u)\}$ (1.24). In this chapter, we propose an alternative, simple but effective, method to address label-switching by directly maximising the local-likelihood functions $\ell\{\boldsymbol{\theta}(u)\}$. The proposed method is a two-stage estimation strategy. To aid the reader's comprehension, the following discussion is based on the NPGMNRs (1.3). However, the method discussed in this chapter can be easily extended to the general model (1.1) and all its special cases as shown in section 4.2.

4.1 A description of the approach

Let $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$ be a set of N local points in the domain of the covariate s. Note that, at any given local point $u \in \mathcal{U}$, model (1.3) is a GMM (2.1). In the *first-stage* of the proposed method, we separately maximise each $\ell\{\boldsymbol{\theta}(u)\}$, for $u \in \mathcal{U}$, via the EM algorithm thus estimating the local GMMs. Among other things, the estimated local GMM at each u has its corresponding set of local responsibilities $\{p_{ik}(u) : i = 1, 2, \ldots, n; k = 1, 2, \ldots, K\}$. In the *second-stage*, based on an appropriate objective function, we choose one of the local GMMs estimated in the first-stage and use its responsibilities to simultaneously maximise the expected value of each complete-data local-likelihood function $\ell^c\{\boldsymbol{\theta}(u)\}$. Note that the posterior probabilities used to maximise each $Q\{\boldsymbol{\theta}(u)|\boldsymbol{\theta}^{(r)}(u)\}$ in the second stage are the same across all local points. Thus, the proposed estimation strategy is also able to avoid the label-switching problem.

The proposed approach can be seen as a modified version of the effective EM-type algorithm to incorporate the local information in the estimation by directly maximising the local-likelihood functions $\ell\{\boldsymbol{\theta}(u)\}$.

The proposed estimation strategy proceeds as follows. In the first-stage, for each local point $u \in \mathcal{U}$, we maximise $\ell\{\boldsymbol{\theta}(u)\}$. Let $\hat{\boldsymbol{\Theta}}^{(r+1)}(u) = \{(\hat{\boldsymbol{\theta}}^{(r+1)}(u), \hat{\mathbf{p}}^{(r+1)}(u)\}$, where $\hat{\mathbf{p}}^{(r+1)}(u) = \{\hat{p}_{ik}^{(r+1)}(u) : i = 1, 2, \ldots, n; k = 1, 2, \ldots, K\}$, be the set of local parameter estimates $\hat{\boldsymbol{\theta}}(u)$ and the posterior probabilities $\hat{\mathbf{p}}(u)$, jointly referred to as local estimates, obtained at local point u.

Let $f : \hat{\boldsymbol{\Theta}}(u) \to \mathbb{R}$ be a convex objective function (see page 56 of Bishop [2006]) and $\hat{\boldsymbol{\Theta}}(u^*)$, for $u^* \in \mathcal{U}$, be the set of local estimates that minimises $f\{\hat{\boldsymbol{\Theta}}(u)\}$ over all the local points $u \in \mathcal{U}$.

In the second-stage, we simultaneously maximise each $\ell^c\{\boldsymbol{\theta}(u)\}$, hence $Q\{\boldsymbol{\theta}(u)|\boldsymbol{\theta}^{(r)}(u)\}$, using the posterior probabilities $\hat{\mathbf{p}}(u^*)$. That is, maximise

$$Q\{\boldsymbol{\theta}(u)|\boldsymbol{\theta}^{(r)}(u)\} = \sum_{i=1}^{n}\sum_{k=1}^{K} \hat{p}_{ik}\Big[\log\{\pi_k(u)\} + \log\mathcal{N}\big(y_i|m_k(u), \sigma_k^2(u)\big)\Big] \times K_h(s_i - u), \tag{4.1}$$

where $\hat{p}_{ik} \equiv \hat{p}_{ik}(u) = \hat{p}_{ik}(u^*)$, for all $u \in \mathcal{U}$. Thus, we obtain $\hat{\boldsymbol{\theta}}^{(r+1)}(u)$ for all $u \in \mathcal{U}$ using the same responsibilities $\hat{\mathbf{p}}(u^*)$. We use the above obtained local parameter estimates as the new local parameter estimates and return to the first-stage. Repeat the above two stages until convergence.

We refer to the above two-stage estimation strategy as the *objective-based approach*. This estimation strategy can be summarised as follows:

The objective-based approach

Stage 1:

(a) For each $u \in \mathcal{U}$, maximise (1.24) to obtain the local estimates $\hat{\boldsymbol{\Theta}}(u)$.

(b) Let $\hat{\boldsymbol{\Theta}}(u^*)$, for $u^* \in \mathcal{U}$, be the local estimates such that

$$\hat{\boldsymbol{\Theta}}(u^*) = \min_{u\in\mathcal{U}} f\{\hat{\boldsymbol{\Theta}}(u)\} \tag{4.2}$$

Stage 2: Simultaneously maximise (4.1), over all $u \in \mathcal{U}$, using the responsibilities $\hat{\mathbf{p}}(u^*) \in \hat{\boldsymbol{\Theta}}(u^*)$, to obtain $\hat{\boldsymbol{\theta}}(u)$ for all $u \in \mathcal{U}$.

Repeat Stage 1 and 2 until convergence.

4.1.1 Rationale of the proposed estimation strategy

In subsection 3.2.1, it was mentioned that the local responsibilities $\hat{\mathbf{p}}(u)$ have to be aligned across all the N local points. We assume this to be the case and choose one set of local responsibilities, among all N sets from the first-stage, as the common responsibilities used to simultaneously maximise each $\ell^c\{\boldsymbol{\theta}(u)\}$ in the second-stage.

The proposed estimation strategy is similar to the approach used to address label switching in Bayesian mixture modelling. In Bayesian mixture modelling, one of the permutations of the component labels is chosen and applied across all the MCMC samples. See Jasra et al. [2005] for more details.

4.1.2 Choice of objective function f

As mentioned in section 1.2, one of the consequences of label-switching is wiggly and non-smooth, hence rough, estimates of the non-parametric functions. Thus, the natural objective function is the one that measures the roughness of a given function. The corresponding optimisation problem is to choose a function with the smallest roughness measure among all available functional estimates. Towards that end, let $\hat{g}_k^u : \hat{\boldsymbol{\Theta}}(u) \to \mathbb{R}^n$ be a functional estimate of any k^{th} component non-parametric function $g_k(s)$. This can be $\pi_k(s)$, $\sigma_k^2(s)$ or $m_k(s)$.

In the regression context, an intuitively appealing way to measure the roughness of, say the regression function $m(s)$, is to calculate its integrated squared second derivative, see Green and Silverman [1994]

$$f\{m(s)\} = \int \{m''(s)\}^2 ds. \tag{4.3}$$

The function (4.3) is a roughness or penalty function of a twice-differentiable function $m(s)$. It is used in penalised regression modelling (or smoothing splines) as part of the penalty term, see Green and Silverman [1994] for an introduction to penalised regression modelling, in particular, and penalised modelling, in general.

As a measure of roughness, it is location invariant. This implies that if two functions differ by a constant or linear term they will have the same roughness. Another appealing property of the roughness measure (4.3), which is also of practical importance, is that it is computationally advantageous as we show below. For more details about the roughness function (4.3) and roughness functions, in general, see Green and Silverman [1994] and Good and Gaskins [1971], respectively.

As mentioned in the foregoing paragraph, given any twice-differentiable function $m(s)$, we can easily calculate its roughness using $f\{m(s)\}$. The analytical expression of the right side of (4.3)

is given by

$$f\{m(s)\} \equiv \int \{m^{''}(s)\}^2 ds = \mathbf{m}^{\intercal}\mathbf{K}\mathbf{m}, \tag{4.4}$$

where $\mathbf{m} = (m(s_1), m(s_2), \ldots, m(s_n))$ and $\mathbf{K}$ is an $n \times n$ basis matrix that is dependent on the covariate s but independent of the response y. See Green and Silverman [1994] for more details on the derivation of $\mathbf{K}$.
We will make use of (4.4) as our objective function. For practical purposes, we define $f\{\cdot\}$ as

$$f\{\hat{g}^u(s)\} = \sum_{k=1}^{K} \hat{\boldsymbol{g}}_k^{\intercal}(u)\mathbf{K}\hat{\boldsymbol{g}}_k(u) \tag{4.5}$$

or

$$f\{\hat{g}^u(s)\} = \max_k \hat{\boldsymbol{g}}_k^{\intercal}(u)\mathbf{K}\hat{\boldsymbol{g}}_k(u), \tag{4.6}$$

where $\hat{\boldsymbol{g}}_k(u) = (\hat{g}_k(s_1), \hat{g}_k(s_2), \ldots, \hat{g}_k(s_n))$.
Of course other objective functions can be used, we chose the roughness functional (4.3) because it is a global, intuitively appealing and computationally advantageous objective function. Surely there are other objective functions that satisfy some or all of these properties. The investigation of other objective functions may be an interesting study for future research.

4.2 Essays: Objective-based approach

In this section, we demonstrate the effectiveness and practical utility of the proposed objective-based approach to address label-switching. The presentation of this section is in the form of essays. Each essay demonstrates the estimation of a given model using the proposed estimation approach. The first essay (subsection 4.2.1) estimates model (1.3) and the second essay (subsection 4.2.2) estimates model (1.8). The models used in our essays were chosen for illustrative purposes.

4.2.1 Non-parametric Gaussian mixtures of regressions [1] (NPGMNRs)

As mentioned in chapter 1, the NPGMNRs (1.3) is a flexible version of the GMLRs model. Unlike the GMLRs model, model (1.3) does not suffer from mis-specification bias. Moreover, the model can be used to verify a parametric form or discover a new form without making any prior assumptions. Thus, model (1.3) is at least as applicable in practice as the GMLRs model.

[1] Some of the results presented in this essay have appeared in a journal article Skhosana et al. [2022]

Algorithm 2 Fitting NPGMNRs (1.3) using the objective-based approach

Stage 1: For $u \in \mathcal{U}$, we obtain the local responsibilities $\hat{\mathbf{p}}(u)$.
Stage 2:

(a) For $u \in \mathcal{U}$, let $\hat{p}_{ik} = \hat{p}_{ik}(u)$, for all $u \in \mathcal{U}$, and maximise (4.1) over all $u \in \mathcal{U}$ to obtain the non-parametric functions $\hat{\boldsymbol{\Theta}}(u)$.

(b) Among the $\hat{\boldsymbol{\Theta}}(u)$, for $u \in \mathcal{U}$, let $\hat{\boldsymbol{\Theta}}(u^*)$, for $u^* \in \mathcal{U}$, be the non-parametric estimates satisfying (4.2). Finally, the resulting two-stage estimates are given by $\hat{\boldsymbol{\pi}}(u^*)$, $\hat{\mathbf{m}}(u^*)$ and $\hat{\boldsymbol{\sigma}}^2(u^*)$.

Stage 3: We improve the two-stage estimates $\hat{\boldsymbol{\pi}}(u^*)$, $\hat{\mathbf{m}}(u^*)$ and $\hat{\boldsymbol{\sigma}}^2(u^*)$ by using them to initialise the effective EM-type algorithm. Let $\tilde{\boldsymbol{\pi}}$, $\tilde{\mathbf{m}}$ and $\tilde{\boldsymbol{\sigma}}^2$ be the resulting estimates.

In this essay, we demonstrate the proposed objective-based estimation strategy for estimating model (1.3). We use simulated data and real data to show the effectiveness and practical usefulness, respectively, of the proposed approach in addressing label-switching.
The structure of the essay is a follows. We first present specific details of the estimation procedure for the model understudy. Next, we present a simulation study and then an analysis of real data to assess the effectiveness of the proposed approach in addressing label-switching and demonstrate the practical utility of the proposed approach, respectively. Finally, we conclude the essay.

Estimation procedure

Note that the implementation of the proposed estimation approach here incorporates an additional step. To improve the two-stage estimates, we propose to initialise the effective EM algorithm using the two-stage estimates $\hat{\boldsymbol{\pi}}(u^*)$, $\hat{\mathbf{m}}(u^*)$ and $\hat{\boldsymbol{\sigma}}^2(u^*)$.
Let $\tilde{\boldsymbol{\pi}}$, $\tilde{\mathbf{m}}$ and $\tilde{\boldsymbol{\sigma}}^2$ be the resulting estimates from running the effective EM algorithm starting from the two-stage estimates. The above estimation procedure is summarised in Algorithm 2.

Simulations

In this section, we perform intensive Monte Carlo simulations to demonstrate the finite sample performance of the proposed objective-based estimation procedure (henceforth, OB-EM algorithm) for fitting NPGMNRs while correcting label-switching and producing sensible estimates of the non-parametric functions. The code for the algorithm was written in the *R* programming language (R Core Team [2023]).

Choosing the Bandwidth and Number of Components To choose the bandwidth h, we make use of the multi-fold cross-validation approach as defined in Huang et al. [2013]. For the number of components, we use the BIC information criterion defined as follows

$$-2\ell + log(n) \times df, \tag{4.7}$$

where ℓ is the observed maximum log-likelihood value at convergence of the EM algorithm, $log(n)$ is a penalty term and df is the degrees of freedom measures by the complexity of the model (see Huang et al. [2013] for more details). Because the bandwidth, h, and number of components, K, are interdependent, for our simulations and application, we make use of the following approach to choose these tuning parameters:

(1) For each $k = 1, 2, \ldots, Kmax$, find the best bandwidth using the cross-validation approach, where $Kmax$ is the largest number of components to consider.

(2) For each of the models in (1) based on the best bandwidth, choose as a final model the one that minimises the BIC

Performance Measures To evaluate the goodness of the fitted non-parametric functions, based on the proposed OB-EM estimates, we make use of the following measure

Root of the Average Squared Errors (RASE)

$$\text{RASE}^2(f) = \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}\gamma_{ik}\left[f_k(s_i) - \hat{f}_k(s_i)\right]^2, \tag{4.8}$$

where f_k is a non-parametric function of the k^{th} component and $\hat{f}_k$ is its estimate and γ_{ik} is the responsibility obtained at convergence. The function f_k can be either $m_k(\cdot)$, $\pi_k(\cdot)$ or $\sigma_k^2(\cdot)$

Initialising the Fitting Algorithm We will make use of the following strategy to initialise the fitting algorithm:

(1) For each $p = 2, 3, \ldots, 5$, we estimate 20 $p^{th}-$degree polynomial GMLRs models.

(2) Choose the model that minimises the BIC in (1) to initialise the model.

Simulation Studies We now present the simulation results that we performed in order to assess the performance of the proposed OB-EM algorithm for estimating model (1.3). For each of our numerical experiments, we generated 500 data sets of sizes $n = 200$, 400 and 800. The

covariate s is generated from a uniform distribution on the unit interval $(0, 1)$. We made use of $N = 100$ local points chosen uniformly on the range of s. We made use of the Epanechnikov kernel function.

Example 1 First, we demonstrate the effectiveness of the proposed approach (OB-EM) in addressing label-switching compared to the naïve EM algorithm (naiveEM). The data used for this example, were generated from model (1.3) given in Table 4.1 where $a = 3$.

For a sample of sise $n = 400$, Figure 4.1 shows four of the 500 fitted CRFs obtained using the

Table 4.1: NPGMNRs model generating the data.

	k	
	1	2
$\pi_k(s)$	$\exp(0.5s)/\{1 + \exp(0.5s)\}$	$1 - \pi_1(s)$
$m_k(s)$	$a - \sin(2\pi s)$	$\cos(3\pi s)$
$\sigma_k^2(s)$	$0.6\exp(0.5s)$	$0.5\exp(-0.2s)$

OB-EM algorithm (left-column) and the naiveEM algorithm (right-column). For illustrative purposes, the four fitted models were chosen randomly. As can be seen from the figure, the fitted CRFs based on the naiveEM are wiggly, non-smooth and exhibit discontinuous jumps which is characteristic of label-switching. Due to the discontinuities and lack of stability, as evidenced by the wiggliness of the fitted functions, the estimates based on the naiveEM are unreliable and thus not very useful in practice. On the other hand, the estimates based on the OB-EM appear to be reasonably smooth, hence stable, and not sensitive to label-switching, hence reliable and practically useful.

For illustrative purposes, we showed the performance of the two estimation procedures when estimating only the CRFs. However, the above results are applicable also to the mixing proportion and variance functions. To demonstrate this last point and show further evidence of the performance of the two estimation procedures, Table 4.2 gives the average (and standard deviation) of the performance measures for the 500 samples of sises $n = 200$, 400 and 800. As can be seen from the table, the OB-EM algorithm outperforms the naiveEM algorithm.

Example 2 Second, given the success of the proposed approach in addressing label-switching, we now assess its performance in comparison to the effective EM-type algorithm (EffectiveEM). We considered data generated from model (1.3) for different scenarios. The models we considered are given in Table 4.1. The constant a controls the degree of separation between the CRFs, where $a = 2$ represents poor separation and $a = 3$ represents well separated components. The different scenarios are shown in Figure 4.2. The simulation results are given

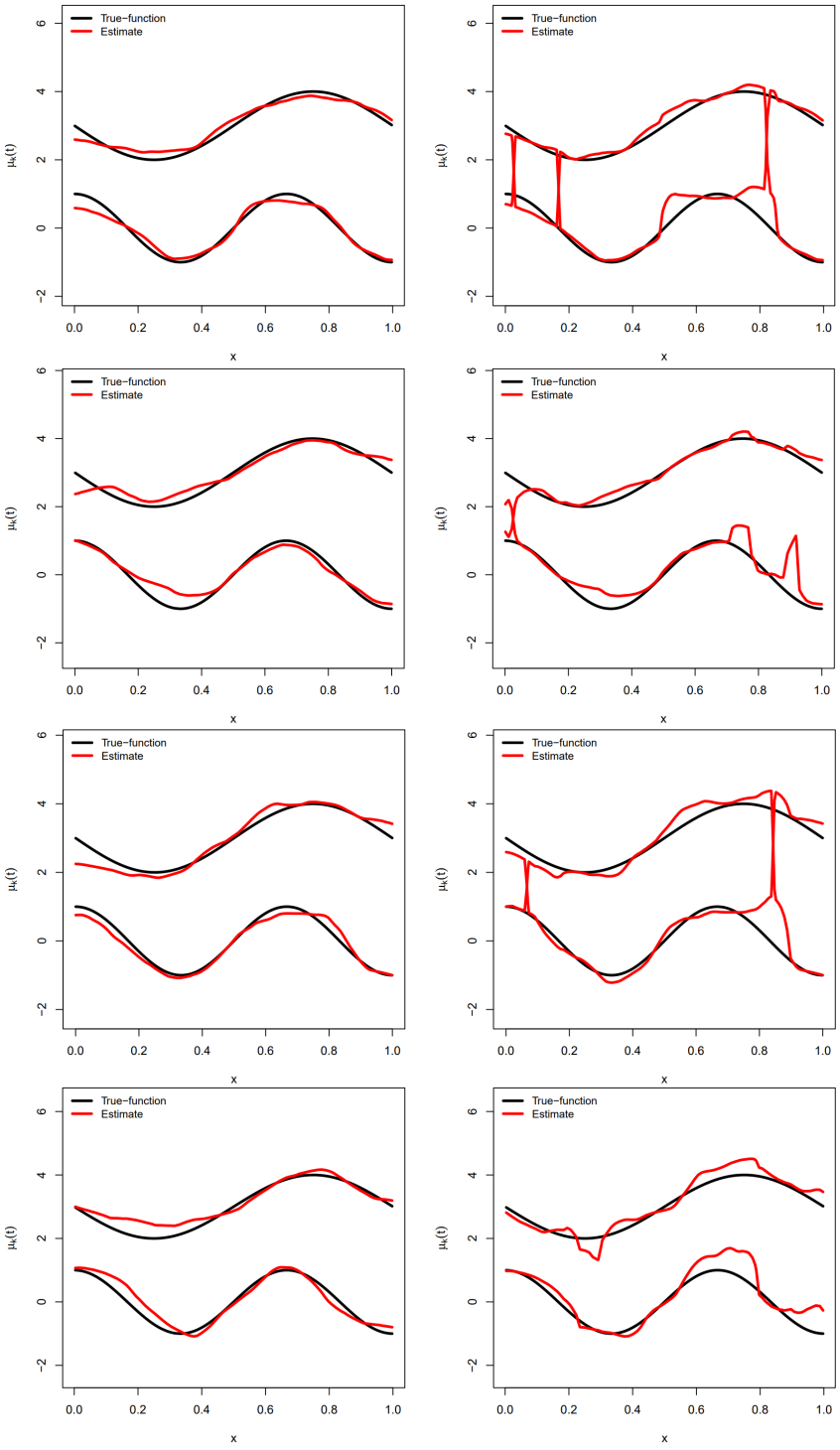


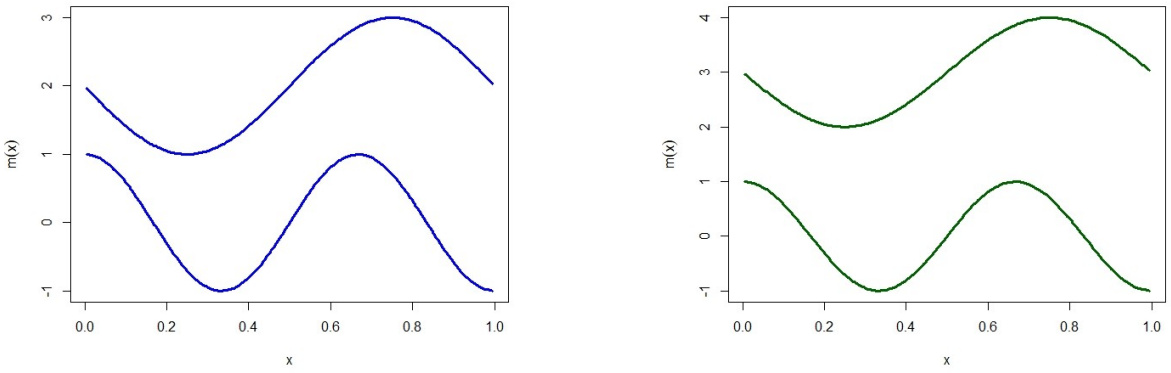
Figure 4.1: True (black curves) and fitted (red curves) CRFs from four randomly chosen estimates obtained via the OB-EM algorithm (**left-column**) and the naiveEM algorithm (**right-column**) for sample sise $n = 400$.

Table 4.2: Average (and standard deviation) of the performance measures for 500 samples.

	RASE(m)	RASE(π)	RASE(σ^2)
		$n = 200$	
OB-EM	0.639 (0.165)	0.231 (0.073)	0.523 (0.174)
NaiveEM	1.758 (1.381)	1.003 (0.001)	0.884 (0.342)
		$n = 400$	
OB-EM	0.476 (0.119)	0.188 (0.061)	0.411 (0.001)
NaiveEM	1.341 (1.419)	1.003 (0.001)	0.642 (0.228)
		$n = 800$	
OB-EM	0.351 (0.082)	0.152 (0.045)	0.329 (0.066)
NaiveEM	0.807 (1.162)	1.003 (0.001)	0.0462 (0.139)

in Table 4.3. The table gives the average and standard deviation of RASE(m), RASE(π) and RASE(σ^2) over the 500 simulations. We can see from the table that in all the scenarios, the OB-EM algorithm gives good estimates of the non-parametric functions. Moreover, for these examples, the OB-EM algorithm outperforms the EffectiveEM algorithm. This might be due to the difficulty of choosing an appropriate initial state. Moreover, this is likely to be a challenge for the EffectiveEM. algorithm since for this algorithm the global responsibilities are more dependent on the initial state compared to the OB-EM algorithm. This can be seen by noting that, at the first few iterations before it settles, the OB-EM algorithm can choose a different set of local responsibilities (as the global responsibilities) than the ones chosen in the previous iterations. This is akin to the stochastic EM (SEM) algorithm (Celeux et al. [1996]). During the first few iterations of the SEM algorithm, there is a non-zero probability of moving from a "superior" local maximum to an inferior local maximum before the algorithm settles. Thus, the OB-EM algorithm is less sensitive to its initial state. This highlights the advantage of taking into account the local responsibilities to obtain the global responsibilities. Finally, to measure the stability and accuracy of the estimates obtained from the OB-EM algorithm, we use the following bootstrap conditional procedure. We approximate the point-wise standard errors as well as the confidence intervals for the model non-parametric functions as follows. For a given s_0 we use the estimated model to generate the corresponding $y^* \sim \sum_{k=1}^{K} \hat{\pi}(s_0)\mathcal{N}(\hat{m}_k(s_0), \hat{\sigma}_k^2(s_0))$; this way we generate the bootstrap sample denoted by $\{(s_i, y_i^*) : i = 1, 2, \ldots, n\}$. We generate $B = 1000$ such samples and fit the model on each sample and average the results to approximate the point-wise standard errors and confidence intervals.

This demonstration is based on the scenario with $a = 2$ for the model in Table 4.1. The results are shown in Figure 4.3 for the CRFs. The local points $u = 0.1, 0.2, \ldots, 0.9$ were used. The plots give the point-wise standard deviations of the estimates over 500 samples which represents the true standard errors (SD) at the local points, as labelled on the graph. The results show



(a) (b)

Figure 4.2: Plots of the component regression functions for the three scenarios of the two-component NPGMNRs model in Table 4.1

Table 4.3: Average (and standard deviation) of the performance measures for 500 samples.

	$a = 2$			$a = 3$		
	RASE(m)	RASE(π)	RASE(σ^2)	RASE(m)	RASE(π)	RASE(σ^2)
			$n = 200$			
OB-EM	0.831	0.331	0.569	0.639	0.231	0.523
	(0.203)	(0.096)	(0.164)	(0.165)	(0.073)	(0.174)
EffectiveEM	1.526	0.330	0.653	1.470	0.248	0.644
	(0.930)	(0.097)	(0.225)	(1.419)	(0.083)	(0.294)
			$n = 400$			
OB-EM	0.642	0.287	0.467	0.476	0.188	0.411
	(0.137)	(0.073)	(0.108)	(0.119)	(0.061)	(0.096)
EffectiveEM	1.750	0.309	0.610	1.178	0.207	0.520
	(1.109)	(0.086)	(0.202)	(1.397)	(0.074)	(0.249)
			$n = 800$			
OB-EM	0.484	0.235	0.389	0.351	0.152	0.329
	(0.107)	(0.060)	(0.080)	(0.082)	(0.045)	(0.066)
EffectiveEM	1.845	0.257	0.565	0.758	0.165	0.387
	(1.152)	(0.075)	(0.190)	(1.165)	(0.063)	(0.188)

slight over- and under-estimations; however, the procedure works well as it shows that the SD are within two standard errors of the estimated point-wise bootstrap standard errors (SE). This can be observed on the plot which shows that all the SDs are within the approximate 95% point-wise bootstrap confidence intervals. The bootstrap procedure works similarly for both the variance and mixing proportion functions.

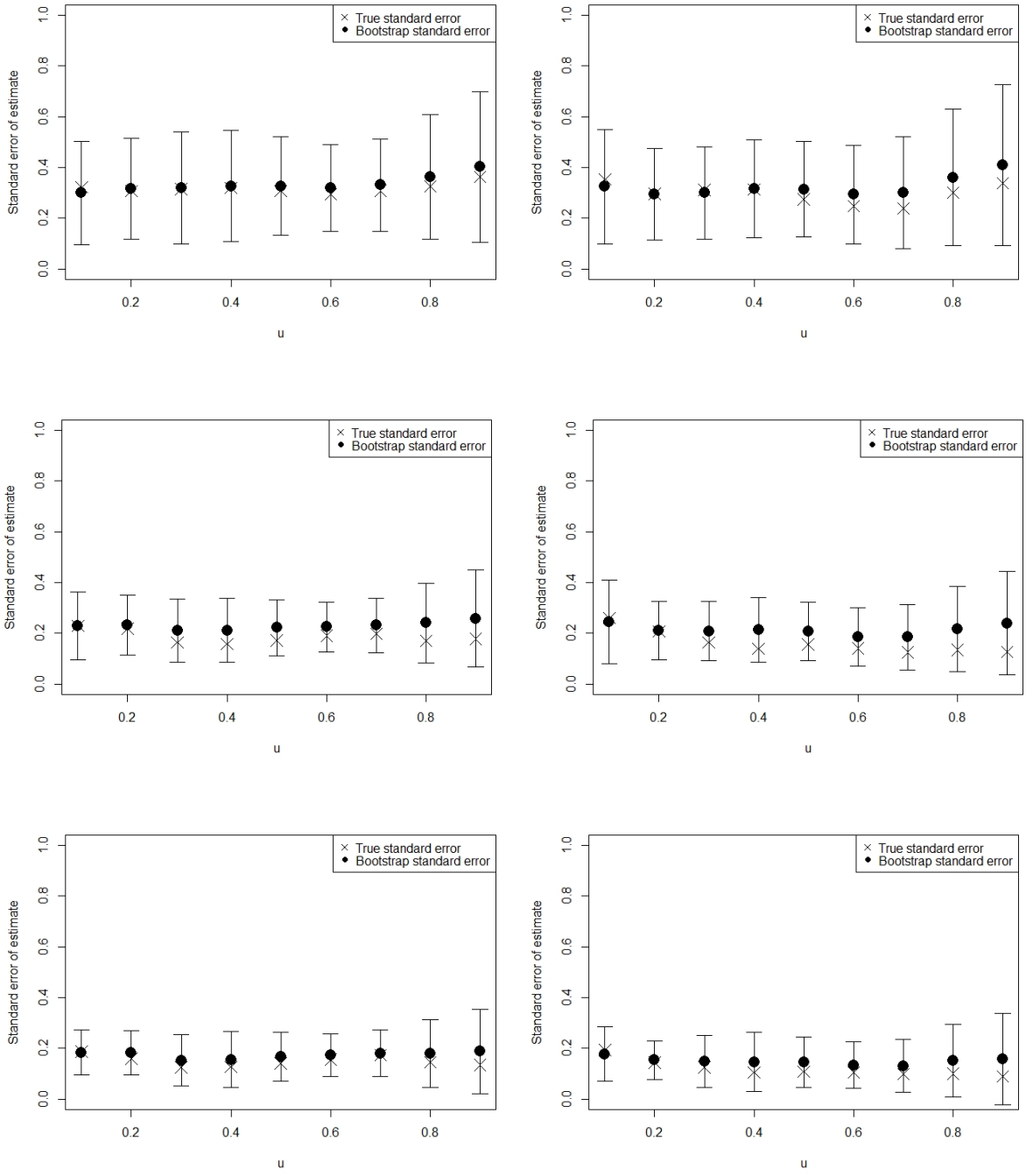


Figure 4.3: Bootstrap standard errors: plots of the estimated point-wise bootstrap standard errors at the local points (shown by the bullet) for the estimated first CRF (left panel) and second CRF (right panel) for sample sises $n = 200$ (top panel), $n = 400$ (middle panel) and $n = 800$ (bottom panel). The error bars represent the approximate 95% point-wise bootstrap confidence intervals at the local points. We also plot the point-wise standard errors (shown by the cross) obtained as the standard deviation of the 500 estimates.

Application

We now demonstrate the practical utility of the proposed estimation strategy on real data. The data consists of per capita CO_2 emissions (in metric tons) and per capita gross domestic product (GDP) (in US$ on a log base scale) for a cross-section of 145 countries for the year 1992. The data were extracted from the *World Development Indicators* database of the World Bank Group. The data are plotted in Figure 4.4a. Each data point on the figure is labelled by the corresponding country's code, for example ZAF is South Africa and CZE is Cezch Republic. A similar dataset, with 28 countries for the year 1996, was analysed by Hurn et al. [2003]. They fitted a $K = 2$ component GMLRs model (2.19) of CO_2 per capita (y) on GNP per capita (s) consisting of two groups of countries. They further mentioned that the identification of these groups "... may help to clarify on which development path they are embarking".
In this analysis, we make no assumption about the functional form of the CRFs and fit a $K = 1, 2, \ldots, 5$ component NPGMNRs model (1.3) to the data set plotted in Figure 4.4a. We also present the results obtained for a GMLRs model fitted on the current data. We choose as the best model the one that minimises the BIC. The resulting BIC values are presented in Table 4.4. We can clearly see that the BIC favours a two-component ($K = 2$) NPGMNRs model. Thus, the $K = 2$ component NPGMNRs model provides the best explanation for this data. Figure 4.4b plots the estimated model. The components were identified by hard classification using the largest responsibility for each data point. We can see from the figure that, for the year 1992, for one group of countries (shown in red), which includes the United Arab Emirates (ARE), a higher income per capita corresponded with a higher quantity of CO_2 emitted per capita. On the other hand, for the other group of countries (shown in blue), which includes Switzerland (CHE), a higher income per capita corresponded with a lower quantity of CO_2 emitted per capita. An interesting point to note about the fitted CRFs in Figure 4.4b is that

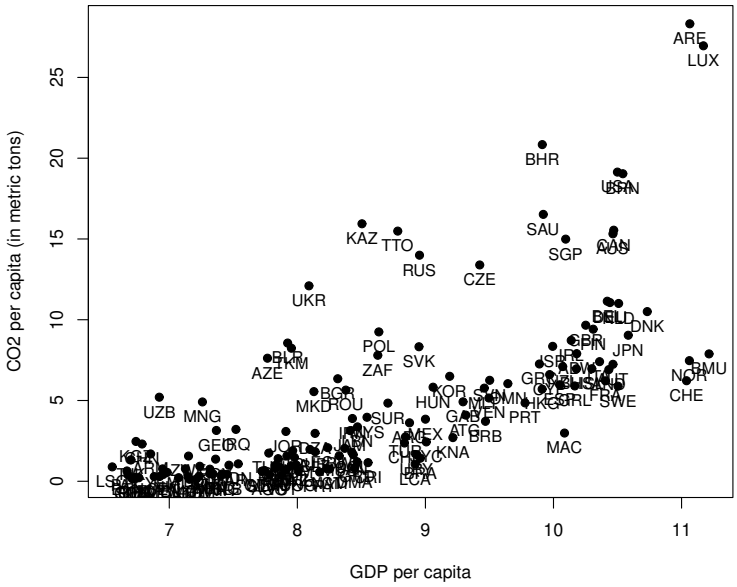
Model	K	h	BIC
NPGMNRs	1	0.945	747.2678
	2	**0.945**	**695.4279**
	3	0.945	746.0681
	4	0.945	809.6640
	5	0.945	869.1760
GMLRs	1		810.1633
	2		704.6483
	3		706.5871
	4		718.1488
	5		728.8317

Table 4.4: BIC values for the fitted NPGMNRs model on the climate data.

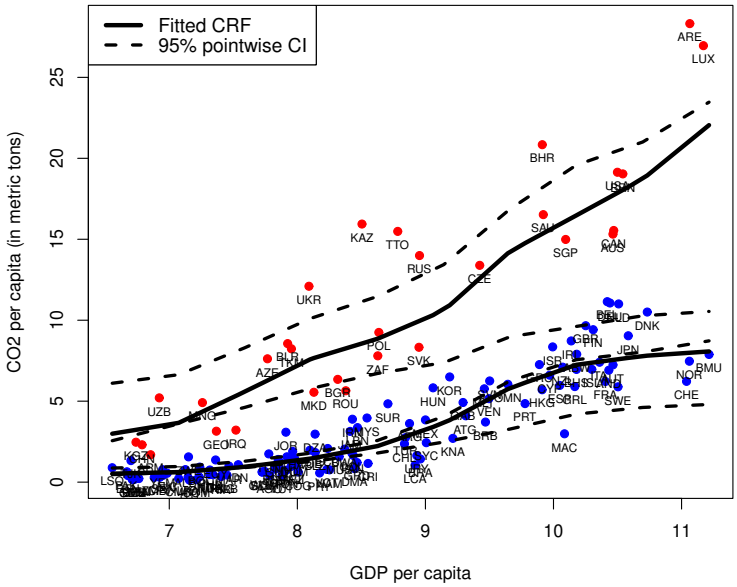
they are in agreement with the environmental Kuznets curve (EKC) hypothesis (Dinda [2004]). The hypothesis states that, as a country becomes industrialised, its carbon emissions increase faster than its income. This environmental degradation continues up until a certain level of income. Beyond this level of income, there is a reduction in carbon emissions. Thus, the EKC hypothesis postulates an inverted-U shaped relationship between environmental degradation (such as carbon emissions) and income. Assuming that all countries follow the same EKC (see Dinda [2004]). It follows that, for a cross-section of countries representing different income groups, it should be observed that poor countries are yet to be industrialised and thus are at the initial stage of the EKC, some developing countries are in the process of industrialisation and thus are at or approaching the peak emission levels and finally developed countries are beyond the peak. Evidence of this is easily seen in Figure 4.4b. For one group of countries (shown in blue points), which includes a lot of high income countries, the peak emission level is reached, whereas for the other group (shown in red points), which has mainly low to middle income countries, by 1992 standards, a peak is yet to be reached.

Discussion and Conclusion

In this essay, we demonstrated the performance of the objective-based approach (OB-EM) to estimate NPGMNRs (1.3) using a simulation study and a real data problem. For illustrative purposes, we considered only two-component NPGMNRs models. However, the proposed approach is applicable for any number of mixture components. The main results are as follows. First, the proposed approach was shown to be successful in addressing label-switching. Second, a comparison of the OB-EM algorithm with the EffectiveEM algorithm reveals some interesting points: (1) for poorly separated mixture components, the OB-EM shows a better performance, whereas for well separated components the performance of the two approaches is similar; (2) the proposed approach is less sensitive to its initial state.
For our real data example, we considered the relationship between CO2 emissions (as the response) and national income (as the covariate) for a cross-section of 145 countries. The practical usefulness of the OB-EM algorithm was demonstrated through its ability to identify two latent components corresponding to two different developmental paths pursued by the countries under consideration.



(a)



(b)

Figure 4.4: Application data and fitted model: (a) scatter plot of the data and (b) fitted $K = 2$ component NPGMNRs model using the proposed algorithm. The dotted curves give the point-wise 95% bootstrap confidence intervals obtained using 1000 bootstrap samples.

4.2.2 Semi-parametric Gaussian mixtures of partially linear models[2] (SPGM-PLMs)

In practical linear regression modelling, it is quite rare to find that all the covariates are linearly related to the response variable. The typical scenario is that some covariates can be fairly assumed to be linearly related to the response variable whereas the functional relationship between the response and each of the other covariates is unknown, hence non-parametric. In such scenarios, the appropriate regression function is a sum of a parametric (linear part) and a non-parametric term (made up of a sum of non-parametric univariate functions known as additive functions). In this form, the model is flexible and still interpretable. A simple case arises when there is only one additive function, resulting in a partial linear model (2.68).
In practice, the structure of the population understudy is unobserved. For simplicity, it is usually assumed to be homogeneous. However, in the real world, there are hidden interactions and associations between different variables some of which divide the population into non-overlapping subpopulations. This is a phenomenon common in many fields such as politics, psychology and economics, among many others (see Fruhwirth-Schnatter et al. [2019]). Thus, it is more often appropriate to assume that the structure of the population heterogeneous, made up of subpopulations of unknown sise and composition. In such cases, the appropriate model is a mixture of partial linear models (1.8). Even in cases where the population is homogeneous, model (1.8) will provide a flexible and usually robust approach to the data analysis. For instance, in the presence of outliers, model (1.8) with $K = 2$ components

$$f(y|\mathbf{X} = \mathbf{x}, T = t) = \pi_1 \mathcal{N}(y|m(\mathbf{x}, t), \sigma_1^2) + \pi_2 \mathcal{N}(y|m(\mathbf{x}, t), \sigma_2^2) \tag{4.9}$$

can be used to account for these outliers by assuming that a proportion π_2 of the observations are outliers. This usually implies that $\sigma_2^2 >> \sigma_1^2$, see subsection 8.2.4 of Frühwirth-Schnatter [2006].
In this essay, we demonstrate the proposed objective-based approach for estimating model (1.8). The format and structure of this essay is the same as the previous essay.

Estimation procedure

Consider a random sample $\{(\mathbf{x}_i, t_i, y_i) : i = 1, 2, \ldots, n\}$ from model (1.8). Let $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$ be a set of N local points in the domain of t.
Note that, at any given local point $u \in \mathcal{U}$, model (1.8) is a GMLRs (2.19), where $\beta_{k,0} = g_k(u)$ is the intercept term of the k^{th} regression component. Thus, the log of the local-likelihood

[2]The results presented in this essay have appeared in a journal article:

function is

$$\ell\{\boldsymbol{\theta}(u)\} = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k(u)\mathcal{N}\big(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_k(u), \sigma_k^2(u)\big)\right] K_h(t_i - u), \tag{4.10}$$

where $\boldsymbol{\theta}(u) = (\boldsymbol{\pi}(u), \boldsymbol{\beta}(u), \boldsymbol{\sigma}^2(u))$, $\mathbf{x} = (x_0, x_1, \ldots, x_p)^\intercal$, with $x_0 = 1$, and $\boldsymbol{\beta}_k(u) = (\beta_{k,0}(u), \beta_{k,1}(u), \ldots, \beta_{k,p}(u))$. The other elements of $\boldsymbol{\theta}(u)$ are as defined in the foregoing essay.

The implementation of the proposed approach for estimating model (1.8) is an extension of the one outlined in the foregoing essay and summarised in Algorithm 2. Due to the presence of both parametric and non-parametric terms, we add an additional stage to obtain appropriate (global) estimates of the parametric term. Thus, the proposed approach here is a three-stage estimation procedure.

In the first-stage, for $u \in \mathcal{U}$, we estimate $\boldsymbol{\theta}(u)$ by maximising (4.10). If we use the LCEs as our local parameter estimates, this is the same as estimating a GMLRs model. Thus, the EM estimation equations are the same as those derived in section 2.1.4.

At the $(r+1)^{th}$ iteration of the E-step, we calculate the local responsibilities as

$$p_{ik}^{(r+1)}(u) = \frac{\pi_k^{(r)}(u)\mathcal{N}\big(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_k^{(r)}(u), \sigma_k^{2(r)}(u)\big)}{\sum_{\ell=1}^{K} \pi_k^{(r)}(u)\mathcal{N}\big(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_k^{(r)}(u), \sigma_k^{2(r)}(u)\big)}. \tag{4.11}$$

At the M-step, we update the local parameters $\boldsymbol{\beta}(u)$ and $\boldsymbol{\sigma}^2(u)$, respectively, using

$$\boldsymbol{\beta}_k^{(r)}(u) = \left(\sum_{i=1}^{n} w_{ik}^{(r)}(u)\mathbf{x}_i\mathbf{x}_i^\intercal\right)^{-1}\left(\sum_{i=1}^{n} w_{ik}^{(r)}(u)\mathbf{x}_i y_i\right), \tag{4.12}$$

$$\sigma_k^{2(r)}(u) = \frac{\sum_{i=1}^{n} w_{ik}^{(r)}(u)(y_i - \mathbf{x}_i^\intercal\boldsymbol{\beta}_k^{(r)}(u))^2}{\sum_{i=1}^{n} w_{ik}^{(r)}(u)}, \tag{4.13}$$

where $w_{ik}^{(r)}(u) = p_{ik}^{(r)}(u)K_h(t_i - u)$. To update $\boldsymbol{\pi}(u)$, we use (1.26).

Let $\hat{\mathbf{p}}(u)$ be the set of estimated local responsibilities at convergence of the local EM algorithm. In the second-stage, for $u \in \mathcal{U}$, set $\hat{\mathbf{p}}(u)$, as the common responsibilities and simultaneously maximise

$$Q\{\boldsymbol{\theta}(u)|\boldsymbol{\theta}(u)\} = \sum_{i=1}^{n}\sum_{k=1}^{K} \hat{p}_{ik}\Big[\log\{\pi_k(u)\} + \log\mathcal{N}\big(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_k(u), \sigma_k^2(u)\big)\Big] \times K_h(t_i - u) \tag{4.14}$$

over all $u \in \mathcal{U}$ as in the effective algorithm, where $\hat{p}_{ik} = \hat{p}_{ik}(u)$ for all $u \in \mathcal{U}$.
Let $\hat{\boldsymbol{\Theta}}(u) = (\hat{\boldsymbol{\pi}}(u), \hat{\boldsymbol{\beta}}(u), \hat{\boldsymbol{\sigma}}^2(u))$, where $\hat{\boldsymbol{\pi}}(u) = (\hat{\pi}_k(t_i))_{ik}$ and $\hat{\boldsymbol{\sigma}}^2(u) = (\hat{\sigma}_k^2(t_i))_{ik}$ are $n \times K$ matrices and $\hat{\boldsymbol{\beta}}(u) = (\hat{\boldsymbol{\beta}}_k(t_i))_{ik}$ is an $n(D_1 + 1) \times K$ matrix, be the estimated non-parametric functions obtained using the local responsibilities $\hat{\mathbf{p}}(u)$ as the common responsibilities across all local points $\mathcal{U}$. We repeat this for $u \in \mathcal{U}$ to obtain the estimated non-parametric functions $\hat{\boldsymbol{\Theta}}(u)$ for all $u \in \mathcal{U}$.
Recall that $\beta_{k,0}(u) = g_k(u)$, it follows that $\hat{g}_k(t_i) = \hat{\beta}_{k,0}(t_i)$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$.
Let $\hat{\boldsymbol{\Theta}}(u^*)$, for $u^* \in \mathcal{U}$, be the estimated non-parametric functions such that

$$\max_k \hat{\boldsymbol{g}}^\intercal(u^*)\mathbf{K}\hat{\boldsymbol{g}}(u^*) \tag{4.15}$$

is a minimum over all $u \in \mathcal{U}$, where $\hat{\boldsymbol{g}}(u^*) = (\hat{g}_k(t_i))_{ik}$ is an $n \times K$ matrix.
Finally, we choose $\hat{\boldsymbol{\Theta}}(u^*)$ to be the two-stage non-parametric estimates of $\boldsymbol{\pi}$, $\boldsymbol{\beta}$, $\boldsymbol{g}$ and $\boldsymbol{\sigma}^2$.
As already noted, the parameters $\boldsymbol{\pi}$, $\boldsymbol{\beta}$ and $\boldsymbol{\sigma}^2$ are global. However, their two-stage estimates are non-parametric. The efficiency of the two-stage estimates can be improved by estimating the global parameters globally. We therefore propose a third-stage, in which, given the two-stage estimates $\hat{\boldsymbol{g}}$, we maximise the global log-likelihood function

$$\ell\{\boldsymbol{\pi}, \boldsymbol{\beta}, \boldsymbol{\sigma}^2\} = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k \mathcal{N}(y_i^* | \mathbf{x}_i^\intercal \boldsymbol{\beta}_k, \sigma_k^2)\right] \tag{4.16}$$

with respect to $\boldsymbol{\pi}$, $\boldsymbol{\beta}$ and $\boldsymbol{\sigma}^2$, where $y_i^* = y_i - \hat{g}_k(t_i)$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$.
The maximisation of (4.16) follows the usual parametric maximum likelihood via the EM algorithm discussed in section 2.1.4. Let $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$ be the resulting three-stage estimates from maximising (4.16)
We can also improve the two-stage non-parametric estimates $\hat{\boldsymbol{g}}$. To do this, given the three-stage estimates $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$, we re-estimate $\boldsymbol{g}$ by maximising the log of the local-likelihood function

$$\ell\{\boldsymbol{g}(u)\} = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \tilde{\pi}_k(u) \mathcal{N}\left(y_i^* | g_k(u), \tilde{\sigma}_k^2(u)\right)\right] K_h(t_i - u), \tag{4.17}$$

where $y_i^* = y_i - \mathbf{x}_i^\intercal \tilde{\boldsymbol{\beta}}_k$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$.
Let $\tilde{\boldsymbol{g}}$ be the resulting estimate obtained by maximising (4.17).
Finally, to estimate $\boldsymbol{\pi}$, $\boldsymbol{\beta}$, $\boldsymbol{g}$ and $\boldsymbol{\sigma}^2$, we propose the estimates $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\beta}}$, $\tilde{\boldsymbol{g}}$ and $\tilde{\boldsymbol{\sigma}}^2$. We refer to the above three-stage estimation procedure as the objective-based backfitted profile likelihood EM (OB-PL-EM) estimation procedure. This estimation procedure is summarised in Algorithm 3

Algorithm 3 Fitting SPGMPLMs (1.8) using the objective-based backfitted profile likelihood EM (OB-PL-EM)

Stage 1: Same as in Algorithm 2
Stage 2: Same as in Algorithm 2
Stage 3: Given the two-stage estimate $\hat{\boldsymbol{g}}$, maximise (4.16) to obtain the three-stage estimates $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$. We can improve the two-stage estimates $\hat{\boldsymbol{g}}$ by maximising (4.17) to obtain $\tilde{\boldsymbol{g}}$.

Simulations

In this section, we perform an extensive simulation study to demonstrate the finite sample performance of the proposed objective based estimation procedure (henceforth, OB-PL-EM algorithm) for fitting the SPGMPLMs in addressing label-switching and producing sensible estimates of the parametric and non-parametric terms of the model. The algorithm was written on the *R* programming language and all the numerical analysis are performed on this platform.

Choosing the Bandwidth h To estimate the non-parametric function $g_k(\cdot)$, we need to choose an appropriate value for the smoothing parameter, h. In practice, this is usually data-dependent based on the cross-validation (CV) or generalised CV (GCV). For estimating model (1.8), Wu and Liu [2017] proposed a multi-fold CV approach to choose h. For the $K = 1$ case, Speckman [1988] proposed a GCV approach to choose h and provided theoretical evidence to support its application. As for its simplicity, we also propose a GCV method to choose h for estimating model (1.8). GCV provides a data-based estimate of h in order to minimise the following unobservable average squared error (ASE):

$$\text{ASE}(h) = \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}[m_k(\mathbf{x}_i, t_i) - \hat{m}_k(\mathbf{x}_i, t_i)]^2, \tag{4.18}$$

where $m_k(\cdot,\cdot)$ and $\hat{m}_k(\cdot,\cdot)$ are the regression function and its estimator for the k^{th} component. In matrix notation, $\hat{m}_k(\cdot,\cdot)$ can be expressed as

$$\begin{aligned}\hat{\mathbf{M}}_k &= \mathbf{X}\hat{\boldsymbol{\beta}}_k + \hat{\mathbf{g}}_{\boldsymbol{\beta}_k} \\ &= \mathbf{X}\hat{\boldsymbol{\beta}}_k + \mathbf{S}_k(\mathbf{y} - \mathbf{X}\hat{\boldsymbol{\beta}}_k) \\ &= \mathbf{A}_k\mathbf{y},\end{aligned} \tag{4.19}$$

where $\mathbf{A}_k = \mathbf{S}_k + \tilde{\mathbf{X}}(\tilde{\mathbf{X}}^\intercal R_k\tilde{\mathbf{X}})^{-1}\tilde{\mathbf{X}}\mathbf{R}_k(\mathbf{I} - \mathbf{S}_k)$ is a linear smoother matrix (see Buja et al. [1989] for more details on linear smoothers). We here define the GCV function as

$$\text{GCV}(h) = \frac{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{(1 - \text{df}/n)^2}, \tag{4.20}$$

where

$$\text{df} = \sum_{k=1}^{K} \text{trace}(\mathbf{A}_k) \tag{4.21}$$

$$\hat{y}_i = \sum_{k=1}^{K} \gamma_{ik}\{\mathbf{x}_i\boldsymbol{\beta}_k + g_{\boldsymbol{\beta}_k}(t_i)\} \tag{4.22}$$

denote the degrees of freedom and the i^{th} fitted value, respectively. In analogy with parametric regression, df represents the effective number of parameters used to estimate the regression function. The GCV criteria selects the bandwidth that minimises GCV(h).

Performance Assessment To assess the performance of the estimator $\hat{g}_k(\cdot)$, we make use of the root of the average squared errors (RASE):

$$\text{RASE}^2(\mathbf{g}_{\boldsymbol{\beta}}) = \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}[g_k(t_i) - \hat{g}_k(t_i)]^2. \tag{4.23}$$

For the parametric estimators, we made use of the ASE:

$$\text{ASE}(\theta) = (\hat{\theta} - \theta)^2, \tag{4.24}$$

where θ is the parameter to be estimated and $\hat{\theta}$ is its estimate.

Initialisation strategy To initialise the algorithm, we made use of a regression spline-based estimator (R-spline-EM). To construct this estimator, we first parameterise each non-parametric function $g_k(t)$ using a set of piecewise polynomial functions, henceforth referred to as basis functions, joined at a set of points, or knots, in the domain of the function (see Wu and Zhang [2006] and James et al. [2021] for more details) as

$$g_k(t) = \sum_{j=1}^{J+Q} \eta_{jk} B_j(t), \tag{4.25}$$

where J is the number of internal knots, Q is the order of the polynomial functions, the $B_j(t)$'s are the basis functions and the η_{jk}'s are the coefficients for the k^{th} component. Substituting (4.25) into the SPGMPLMs (1.8) yields a GMLRs (2.19). Thus, the estimation procedure outlined in section 2.1.4 is applicable.

We make use of the cubic ($Q = 4$) B-spline basis functions with $J = 3$ internal knots chosen to be the 1^{st}, 2^{nd} and 3^{rd} quartiles of the covariate t.

In order to improve the stability of the model estimate and alleviate the issue of the dependence

on the initial solution, we made use of the following initialisation strategy: Fit a mixture of regression splines for a 100 times from random starts and choose as the initial solution the model with the smallest BIC. The `bs` function from the splines R package was used within the R-spline-EM function to compute the basis functions.

Simulation Study Throughout our simulations, we consider a $K = 2$ component mixture environment and a univariate $\mathbf{x}$ (that is, $p = 1$), denoted x. The two covariates, x and t, are generated from a uniform distribution on the interval $(0, 1)$. We generated 500 samples of sises $n = 200$, 400 and 800. We made use of the Epanechnikov kernel function and $N = 100$ grid points. The set of grid points $\mathcal{U}$ was chosen uniformly from the domain of the covariate t.

Example 1 The first aim of this simulation study is to illustrate the effectiveness of the proposed OB-PL-EM algorithm in addressing the label-switching problem. To get a better illustration of the performance of the proposed OB-PL-EM, we considered the following bandwidths: $\frac{2}{3}h_{GCV}$, h_{GCV} and $\frac{3}{2}h_{GCV}$ corresponding to under-smoothing (US), appropriate smoothing (AS) and over-smoothing (OS), respectively, where h_{GCV} denotes the bandwidth selected by the GCV method. This approach will also assist in assessing the sensitivity of the estimation procedure on the bandwidth. The data used in this example were generated from model (1.8) using the $K = 2$ component setting given in Table 4.5.

Table 4.5: The $K = 2$ component SPMPLMs.

k	**1**	**2**
$m_k(x,t)$	$\beta_1 x + g_1(t)$	$\beta_2 x + g_2(t)$
$g_k(t)$	$-\exp(2t)$	$\exp(2t)$
σ_k^2	0.2	0.6
π_k	0.35	0.65
β_k	-1	1

For a sample of sise $n = 400$ and based on the optimal bandwidth h_{GCV}, Figure 4.5 gives the component non-parametric functions of four of the 500 fitted models obtained using the OB-PL-EM algorithm (left-column) and the naiveEM algorithm (right-column). The fitted models were chosen to highlight some of the challenges experienced by the naiveEM algorithm when estimating the non-parametric function $g_k(\cdot)$, namely, the tendency to produce non-smooth estimates and its sensitivity to label-switching. The intention is to show that the OB-PL-EM algorithm can address some of these challenges. As can be seen from the figure, the estimates based on the naiveEM are wiggly, non-smooth and subject to label-switching. Thus, the naiveEM estimates are unreliable and not useful in practice. In contrast, the estimates based on the OB-PL-EM algorithm exhibit no erratic behaviour, are reasonably smooth and

Table 4.6: Average (and standard deviations) of the RASE($\boldsymbol{g}$) over 500 samples.

	h	200	400	800
OB-PL-EM	US	0.243 (0.346)	0.159 (0.041)	0.121 (0.034)
	AS	0.233 (0.377)	0.153 (0.040)	0.119 (0.032)
	OS	0.256 (0.352)	0.181 (0.043)	0.145 (0.032)
naiveEM	US	0.332 (0.297)	0.219 (0.225)	0.127 (0.093)
	AS	0.343 (0.321)	0.203 (0.197)	0.123 (0.100)
	OS	0.379 (0.338)	0.237 (0.220)	0.148 (0.082)

are less sensitive to label-switching, hence reliable and practically dependable.
For further evidence of the effectiveness of the OB-PL-EM algorithm in addressing some of the challenges of the naiveEM algorithm in estimating the component non-parametric functions, Table 4.6 gives the average (and standard deviation) of the RASE(g) over the 500 samples of sizes $n = 200$, 400 and 800.

Example 2 The second aim of this simulation study is to illustrate the performance of the proposed OB-PL-EM algorithm compared to the profile-likelihood EM (PL-EM) algorithm of Wu and Liu [2017].
The data used in this example are the same as those used in Example 1 generated by the model in Table (1.8).
For the same reason as in Example 1, we considered the following bandwidths: $\frac{2}{3}h_{GCV}$, h_{GCV} and $\frac{3}{2}h_{GCV}$ corresponding to under-smoothing (US), appropriate smoothing (AS) and over-smoothing (OS), respectively, where h_{GCV} denotes the bandwidth selected by the GCV method. Tables 4.7–4.9 reports the averages and standard deviations of the performance measures over the 500 samples. The results show that the performance of the proposed OB-PL-EM algorithm is similar to the PL-EM algorithm under all three bandwidths. However, the OB-PL-EM performs slightly better than the PL-EM algorithm.
To assess the accuracy of the fitted non-parametric functions using the OB-PL-EM algorithm compared to the PL-EM algorithm, we make use of the same bootstrap procedure as in subsection 4.2.1.
Based on the fitted model $\hat{\pi}_1\mathcal{N}(y|x\hat{\beta}_1 + \hat{g}_1(t), \hat{\sigma_1^2}) + \hat{\pi}_2\mathcal{N}(y|x\hat{\beta}_2 + \hat{g}_2(t), \hat{\sigma_2^2})$, for each (x_i, t_i), generate y_i^* for all $i = 1, 2, \ldots, n$. Let $\{(y_i^*, x_i, t_i) : i = 1, 2, \ldots, n\}$ be the bootstrap sample obtained in the above manner. We repeated this process 200 times.
For a typical sample of size $n = 400$ generated from the model in Table 4.5 and based on the optimal bandwidth h_{GCV}, figures 4.6 (a) and (b) presents the fitted non-parametric functions based on the OB-PL-EM and the PL-EM, respectively. Included in the figures are the 95% point-wise bootstrap confidence intervals (CI). The resulting estimates are virtually the same.

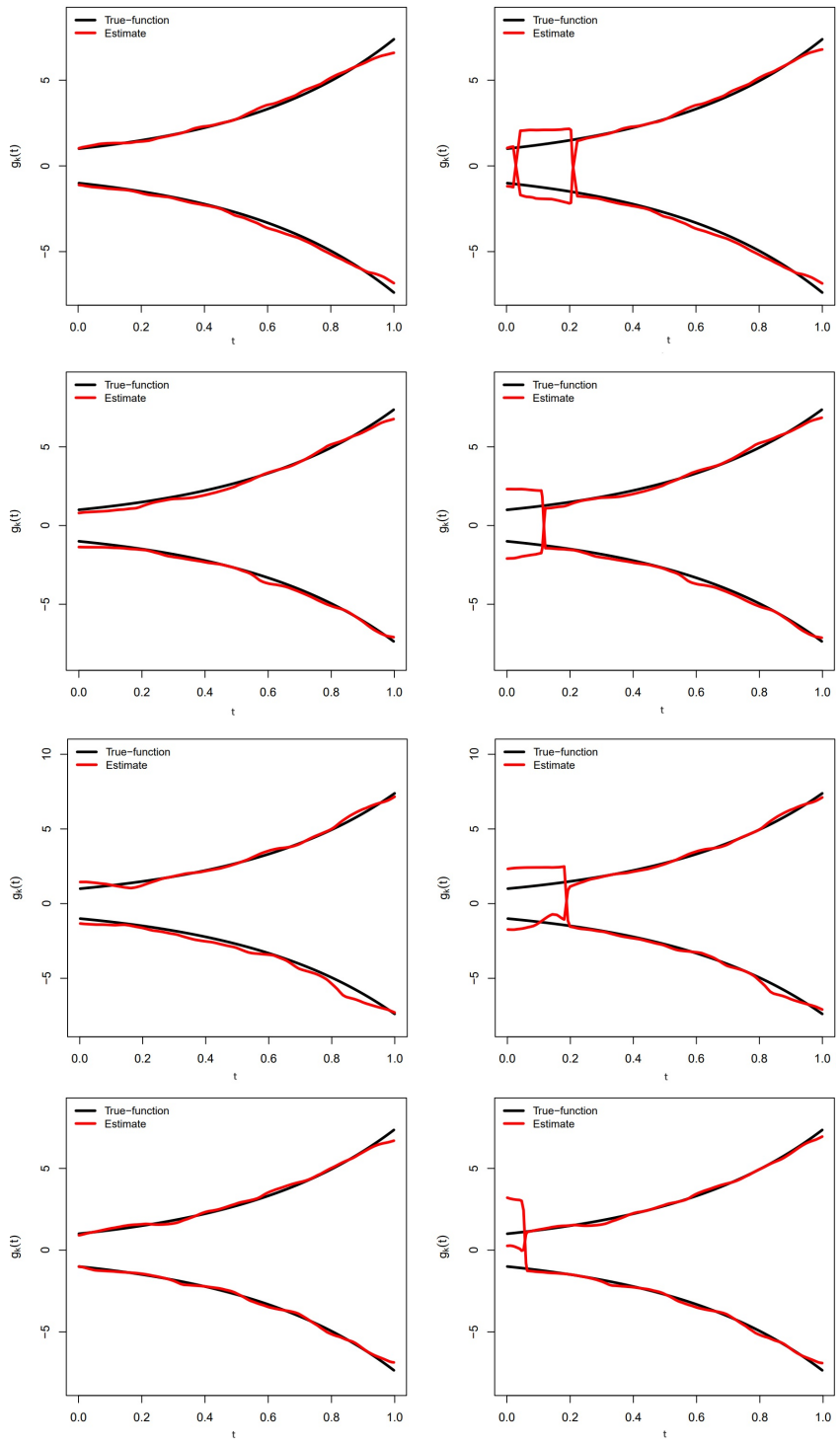


Figure 4.5: True (black curves) and fitted (red curves) CRFs from four randomly chosen estimates obtained via the OB-PL-EM algorithm (**left-column**) and the naiveEM algorithm (**right-column**) for sample size $n = 400$.

Table 4.7: Averages (and standard deviations) of the performance measures over 500 samples of size $n = 200$.

	h	ASE(β_1)	ASE(β_2)	ASE(π_1)	ASE(σ_1^2)	ASE(σ_2^2)	RASE($\boldsymbol{g}$)
OB-PL-EM	US	0.022	0.046	0.001	0.036	0.011	0.205
		(0.059)	(0.138)	(0.002)	(0.760)	(0.028)	(0.161)
	AS	0.017	0.043	0.001	0.001	0.006	0.190
		(0.044)	(0.145)	(0.002)	(0.002)	(0.010)	(0.054)
	OS	0.021	0.074	0.001	0.001	0.016	0.218
		(0.048)	(0.166)	(0.002)	(0.002)	(0.066)	(0.059)
PL-EM	US	0.046	0.055	0.001	0.002	0.010	0.218
		(0.063)	(0.081)	(0.002)	(0.002)	(0.012)	(0.126)
	AS	0.045	0.057	0.001	0.002	0.007	0.204
		(0.059)	(0.077)	(0.002)	(0.002)	(0.009)	(0.056)
	OS	0.052	0.069	0.001	0.001	0.006	0.229
		(0.075)	(0.105)	(0.002)	(0.002)	(0.007)	(0.064)

Application

In this section, we demonstrate the practical usefulness of the proposed objective-based estimation procedure (OB-PL-EM) using two real data examples. The data were obtained from

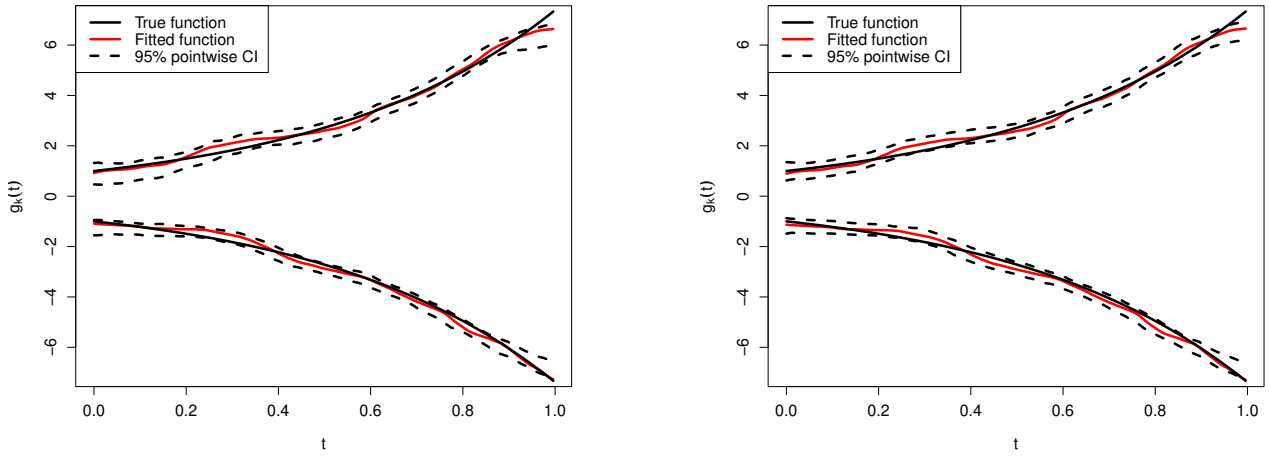


Figure 4.6: Fitted non-parametric component functions (red solid curve) for a typical sample of size $n = 400$ based on OB-PL-EM algorithm (**a**) and the PL-EM algorithm (**b**). The black solid curve gives the true component function. The dotted lines give the 95% bootstrap confidence intervals.

Table 4.8: Averages (and standard deviations) of the performance measures over 500 samples of size $n = 400$.

	h	ASE(β_1)	ASE(β_2)	ASE(π_1)	ASE(σ_1^2)	ASE(σ_2^2)	RASE($\boldsymbol{g}$)
OB-PL-EM	US	0.01	0.026	0.001	0.001	0.004	0.146
		(0.024)	(0.072)	(0.001)	(0.001)	(0.008)	(0.038)
	AS	0.009	0.030	0.001	0.001	0.004	0.142
		(0.026)	(0.085)	(0.001)	(0.001)	(0.009)	(0.039)
	OS	0.011	0.037	0.001	0.001	0.006	0.171
		(0.026)	(0.064)	(0.001)	(0.001)	(0.014)	(0.038)
PL-EM	US	0.019	0.028	0.001	0.001	0.004	0.151
		(0.025)	(0.038)	(0.001)	(0.001)	(0.004)	(0.035)
	AS	0.020	0.031	0.001	0.001	0.003	0.148
		(0.030)	(0.043)	(0.001)	(0.001)	(0.005)	(0.037)
	OS	0.025	0.033	0.001	0.001	0.003	0.174
		(0.035)	(0.042)	(0.001)	(0.001)	(0.004)	(0.041)

the Our World In Data database. The data comprises CO_2 emissions per capita (CO_2), oil consumption per capita (EnergyUse), the number of people living in urban areas (Urbanisation), real GDP per capita (GDP-per-capita) and the percentage share of primary energy attributable to renewable energy sources (RenewEnergyShare) for the period 1990 to 2019 for 7-8 OECD countries.

CO_2 data 1 For our first application, we considered the impact of EnergyUse and Urbanisation on CO_2. After pre-processing the data, we produced scatter plots of the data; see Figure 4.7. A two-regime (component) structure is clearly evident in the figure. Moreover, there appears to be a non-linear relationship between CO_2 and Urbanisation within each regime. In the first regime, including countries such as Australia, per capita CO_2 emissions increase sharply up to a point and then they gradually decrease.

In the second regime, including countries such as Denmark, per capita CO_2 emissions increased gradually up to a point, followed by a sharp decline. This naive interpretation of the data suggests that the relationship between CO_2 and Urbanisation in the two regimes exhibits a form of the environmental Kuznets curve (EKC) (see Dinda [2004]). This could be naively explained as implying that the increased number of people in the urban areas (which is where corporate and industrial activities take place) translates into an increase in skilled human capital. This in turn leads to an increase in productivity. For the data in Figure 4.7, we propose to fit the following model:

$$\pi_1 \mathcal{N}(y|x\beta_1 + g_1(t), \sigma_1^2\} + \pi_2 \mathcal{N}\{y|x\beta_2 + g_2(t), \sigma_2^2), \tag{4.26}$$

Table 4.9: Averages (and standard deviations) of the performance measures over 500 samples of size $n = 800$.

	h	ASE(β_1)	ASE(β_2)	ASE(π_1)	ASE(σ_1^2)	ASE(σ_2^2)	RASE($\boldsymbol{g}$)
OB-PL-EM	US	0.008	0.016	0.001	0.001	0.002	0.114
		(0.019)	(0.043)	(0.001)	(0.001)	(0.003)	(0.033)
	AS	0.006	0.018	0.001	0.001	0.002	0.111
		(0.015)	(0.051)	(0.001)	(0.001)	(0.004)	(0.030)
	OS	0.005	0.034	0.001	0.001	0.005	0.138
		(0.012)	(0.090)	(0.001)	(0.001)	(0.021)	(0.030)
PL-EM	US	0.009	0.015	0.001	0.001	0.002	0.111
		(0.013)	(0.022)	(0.001)	(0.001)	(0.002)	(0.023)
	AS	0.010	0.015	0.001	0.001	0.002	0.111
		(0.013)	(0.021)	(0.001)	(0.001)	(0.002)	(0.024)
	OS	0.011	0.017	0.001	0.001	0.001	0.136
		(0.015)	(0.023)	(0.001)	(0.002)	(0.002)	(0.027)

where y, x and t are CO_2, Energyuse and Urbanisation, respectively. Figure 4.8a plots the GCV(h) over a range of bandwidths, and the minimum GCV(h) occurs at 0.4925. Figure 4.8b shows the estimated non-parametric functions based on the proposed OB-PL-EM algorithm. Included in Figure 4.8b are plots of the 95% point-wise confidence intervals obtained via bootstrapping.

We then checked whether the data can be explained by a simple mixture of linear regressions model. More specifically, we checked whether the non-parametric function $g_k(t)$ has a linear

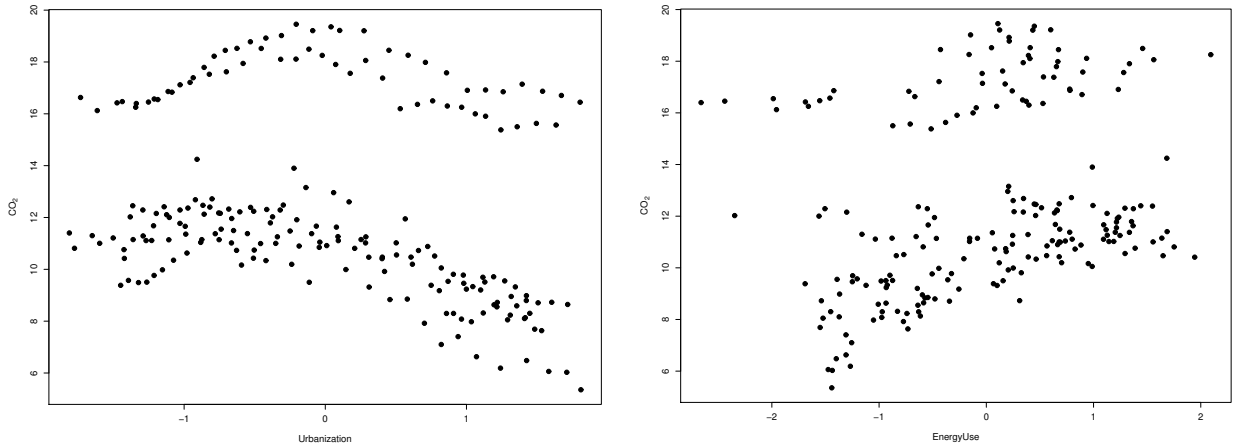
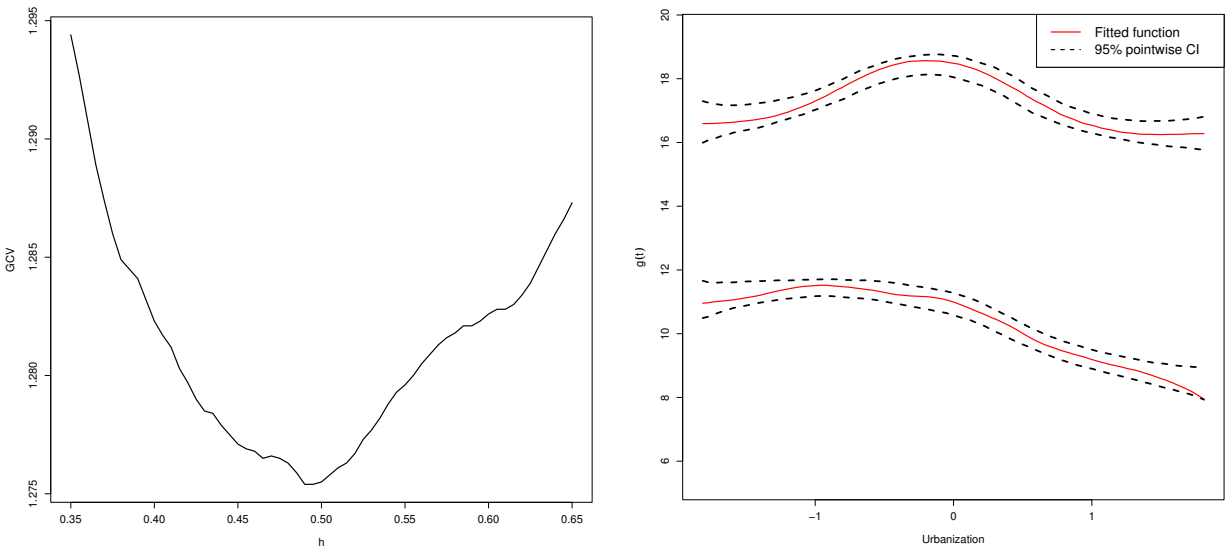


Figure 4.7: Scatter plots of climate data 1.



(a) (b)

Figure 4.8: Fitted model (4.26): (**a**) The GCV plot over a range of bandwidths with a minimum at $h = 0.4925$. (**b**) estimated non-parametric functions (red solid lines) $\hat{g}_k(t) : k = 1, 2$ based on the OB-PL-EM procedure. The dashed lines are the 95% point-wise confidence intervals.

structure. Mathematically, we wanted to test the following hypotheses:

$$\begin{aligned} \mathrm{H}_0 &: g_k(t) = \alpha_{0k} + \alpha_{1k}(t) \qquad \text{for } k = 1, 2 \\ \mathrm{H}_a &: g_k(t) \text{ is a smooth function.} \end{aligned}$$

To test these hypotheses, we made use of the bootstrap specification test proposed by Wu and Liu [2017]. Define the test statistic as

$$T_n = \sum_{i=1}^{n} (\hat{y}_i - \tilde{y}_i)^2, \tag{4.27}$$

where $\hat{y}$ and $\tilde{y}$ are the fitted values (defined as in Equation (4.22)) obtained from fitting the model under the null and alternative hypotheses, respectively. To ensure that our test results would not be sensitive to the choice of the bandwidth, we performed the test using the bandwidths $\frac{2}{3}h_{GCV}$, h_{GCV} and $\frac{3}{2}h_{GCV}$, where $h_{GCV} = 0.4925$. Respectively, the observed test statistics were 48.5473, 48.4349 and 48.3420 with p-values 0.01, 0.022 and 0.02. At a 5% level of significance, we can reject the null hypothesis. Therefore, model (4.26) provides an adequate fit for these data. Table 4.10 gives the estimated parameters of the fitted model (4.26). Included in the table are the 95% confidence intervals obtained via the bootstrap procedure outlined above.

The first thing to note is that the estimated slope parameters are positive, as expected (see Figure 4.7). The second thing to note is that the 95% confidence interval for the slope parameter of the first component includes a zero. This implies that the parameter may not be significant.

To obtain further evidence in support of this conclusion, we conducted the following hypotheses test:

$$\mathrm{H}_0 : \beta_1 = 0$$
$$\mathrm{H}_a : \beta_1 \neq 0$$

Under the null hypothesis, we fit the following reduced model:

$$\pi_1 \mathcal{N}(y|g_1(t), \sigma_1^2) + \pi_2 \mathcal{N}(y|x\beta_2 + g_2(t), \sigma_2^2) \tag{4.28}$$

and under the alternative hypothesis we fit model (4.26). The test statistic was similarly defined as in (4.27). The observed test statistic is 0.0015 with a p-value of 0.426. Thus, we cannot reject the null hypothesis. We can therefore conclude that the reduced model (4.28) provides an adequate fit for the data. The estimated value of β_2 is 0.4337, which is virtually the same as the one in Table 4.10. The same applies for the rest of the parameter estimates. The fitted non-parametric functions are similar to those obtained for the fitted model (4.26), and therefore, they are omitted.

Table 4.10: Parameter estimates of the fitted model (4.26) and the corresponding 95% bootstrap confidence intervals.

	$\boldsymbol{\beta_1}$		$\boldsymbol{\beta_2}$		$\boldsymbol{\pi_1}$		$\boldsymbol{\sigma_1^2}$		$\boldsymbol{\sigma_2^2}$	
	0.126		0.433		0.286		0.288		0.827	
95% CI	−0.043	0.333	0.243	0.627	0.229	0.343	0.178	0.396	0.619	0.984

CO_2 data 2 For our second application, we consider the impact of GDP-per-capita and RenewEnergyShare on CO_2. After pre-processing the data, we produced scatter plots of the data in Figure 4.9. From the figure, we can see that there are at least two components. Moreover, it seems that the relationship between CO_2 and GDP-per-capita is non-linear. For this data, we propose to fit the SPGMPLMs. We first obtained the number of components using the BIC as in Wu and Liu [2017]. We fit the SPGMPLMs for $K = 2$, 3, 4 and 5 and choose the model with the smallest BIC. The BIC scores are 939.1803, 967.2592, 1133.067 and 1220.576, respectively. We therefore fit a $K = 2$ component SPGMPLMs. We use y, t and x to denote CO_2, GDP-per-capita and RenewEnergyShare, respectively. Figure 4.10a shows the GCV

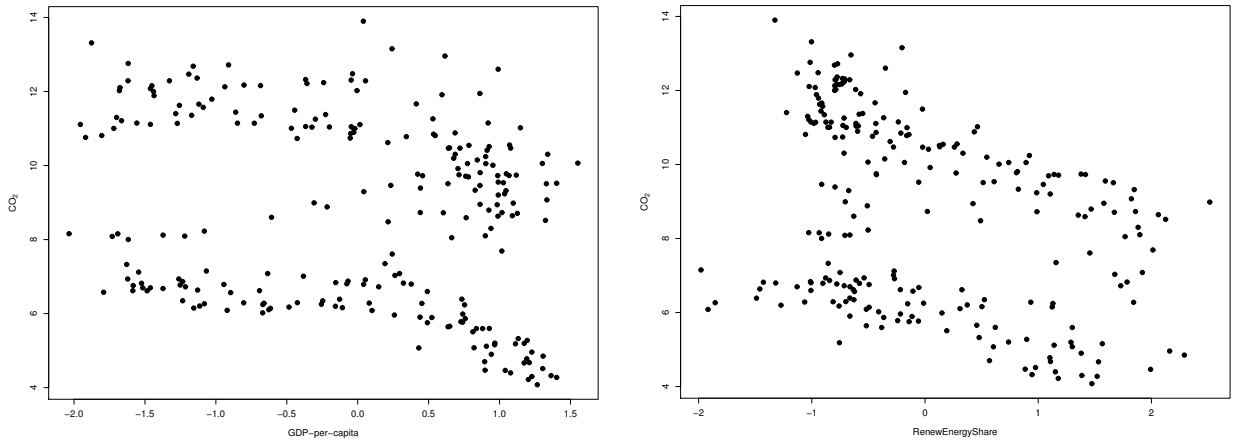


Figure 4.9: Scatter plots of the climate data 2.

plot with the minimum GCV occurring at $h = 0.475$. Figures 4.10b and c gives the fitted non-parametric functions for the two components, respectively, obtained using the OB-PL-EM estimation procedure.

As in the previous application, we conducted a hypothesis test to check whether the non-parametric functions $g_k(t)$ have linear structure. Using (4.27), the observed test statistics are 0.3457, 0.3407 and 0.3273 with p-values 0.02, 0.004 and 0.004, respectively. As before, we used a range of bandwidths to ensure that our results are not sensitive to the choice of bandwidth. The hypothesis test results show that, at a 5% level of significance, we can reject the null hypothesis that the non-parametric functions have a linear structure. It follows that the $K = 2$ component SPGMPLMs is adequate for this data.

Discussion and Conclusion

This essay was concerned with estimating semi-parametric Gaussian mixtures of partially linear models (SPGMPLMs) (1.8) using the proposed objective-based backfitted profile likelihood EM (OB-PL-EM) algorithm in order to demonstrate the effectiveness and practical utility of the algorithm in addressing label-switching and producing useful model estimates, respectively. Following an intensive Monte Carlo simulation study, the following observations were made. First, the naïve EM algorithm was shown to be sensitive to label-switching, resulting in wiggly and non-smooth estimates of the non-parametric functions and thus practically unreliable. Second, the proposed OB-PL-EM algorithm overcomes all the challenges faced by the naïve EM algorithm. The OB-PL-EM algorithm is less sensitive to label-switching, results in reasonably smooth and thus stable estimates of the non-parametric functions. Third, in terms of goodness-of-fit or accuracy of the estimated non-parametric function, the estimate produced by

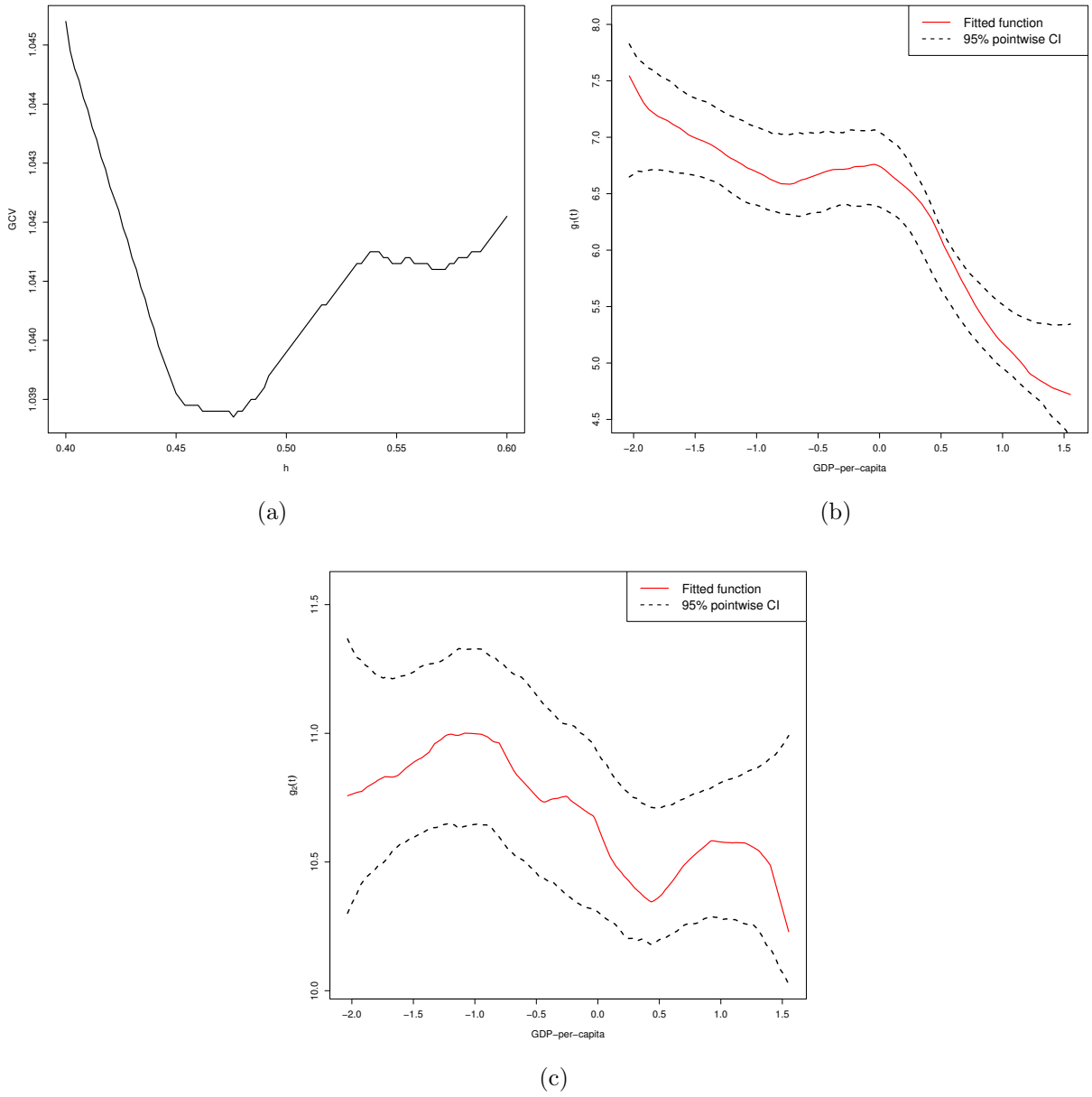


Figure 4.10: Fitted model: (**a**) The GCV plot over a range of bandwidths with a minimum at $h = 0.475$. (**b**) Estimated non-parametric function (red solid lines) $\hat{g}_1(t)$ and (**c**) estimated non-parametric function (red solid lines) $\hat{g}_2(t)$. The dashed lines are the 95% point-wise confidence intervals.

the proposed OB-PL-EM algorithm is, on average, better than that obtained using the naïve EM algorithm. Fourth, the proposed OB-PL-EM algorithm performs as well as a competitive fitting algorithm (PL-EM algorithm Wu and Liu [2017]) for addressing label-switching.

In our real data analysis, we considered two datasets on the impact of (1) the increased consumption of fossil fuel for energy and a growing urban population; and (2) the increased consumption of clean energy coupled with a growing economy on carbon emissions for a panel of OECD countries. The following interesting observations were made. First, the relationship between carbon emissions and urbanisation exhibits the well-known Environmental Kuznets Curve (EKC) hypothesis. That is, a growing urban population leads to more carbon emissions up until a certain level of the urban population. Beyond this point, a further increase in the urban population leads to a decrease in carbon emissions. Second, the relationship between carbon emissions and national income (measured by GDP) differs between the two groups (components) of countries discovered by the fitted model. In the first group, as GDP increases, the fitted curve shows a slow decrease followed by a slight increase then lastly a sharp decrease in CO_2 emissions characterised by a clear general downward trend. In the second group, as GDP increases, there is a slow increase followed by a decrease then another brief sharp increase followed by a sharp decrease in CO_2 emissions. The fitted curve for the second group exhibits a two peak EKC.

4.3 Conclusion

In this chapter, we proposed a novel estimation procedure to address label-switching when estimating any model of the form (1.1). The proposed approach is based on the idea of using the same responsibilities (global responsibilities) to maximise each local-likelihood function at each local M-step of the EM algorithm. The proposed approach proceeds in two stages. In the first-stage, we estimate all the local responsibilities at each local grid point. In the second-stage, based on an appropriate objective function, we choose one set of local responsibilities, among those estimated in the first-stage, as the global responsibilities. Finally, we replace the local responsibilities at each local grid point by the global responsibilities and then proceed to the M-step of the EM algorithm.

The effectiveness and practical usefulness of the proposed method was demonstrated using intensive Monte Carlo simulations and real data analysis, respectively. First, compared with the naïve estimation procedure (see subsection 1.2.1), the proposed approach is less sensitive to label-switching and produces reasonably smooth estimates of the non-parametric functions that are in line with expectations. Second, the proposed method performs at least as well as the competitive estimation procedure.

For illustrative purposes, the efficacy of the proposed method was demonstrated for the case of estimating two special cases of the general model (1.1). However, the method is applicable for estimating any model of the form (1.1).

Chapter 5

Model-based approach to label-switching

In section 1.2 and, in more detail, section 3.2, we saw that label-switching takes place when estimating the local parameters separately using local-likelihood estimation via the EM algorithm. Moreover, note that each local-likelihood function is a likelihood function of a GMM (2.1) in the case of model (1.3) or GMLRs (2.19) in the case of the general model (1.1). This implies that, locally, model (1.1) is a GMLRs.
In this chapter we propose another novel approach to address label-switching by reformulating model (1.1) as a mixture of GMLRs. Estimating the mixture of GMLRs model is, in effect, equivalent to simultaneously estimating the parameters of each local GMLRs and hence all the non-parametric functions of model (1.1).

5.1 A description of the approach

Note that, for $D_3 > 1$ in (1.1), the local GMLRs is not identifiable. Therefore, the following discussion is based on the following reduced version of model (1.1)

$$f(y|\mathbf{X}=\mathbf{x},\mathbf{Z}=\mathbf{z},T=t)=\sum_{k=1}^{K}\pi_k(t)\mathcal{N}(y|m_k(\mathbf{x},\mathbf{z},t),\sigma_k^2(t)), \tag{5.1}$$

where $\mathbf{x} \in \mathbb{R}^{D_1}$ and $\mathbf{z} \in \mathbb{R}^{D_2}$ are the vectors of covariates and $t \in \mathbb{R}$ is a scalar. The CRFs are given by

$$\begin{aligned} m_k(\mathbf{x}, \mathbf{z}, t) &= \sum_{a=1}^{D_1} \beta_{k,a} x_a + \sum_{b=1}^{D_2} \gamma_{k,b}(t) z_b + g_k(t), \\ &= \mathbf{x}^\intercal \boldsymbol{\beta}_k + \mathbf{z}^\intercal \gamma_k(t) + g_k(t). \end{aligned} \tag{5.2}$$

Note that each $\boldsymbol{\beta}_k$ does not include the intercept term, $\beta_{k,0}$. Moreover, each $\boldsymbol{\gamma}_k(t)$ does not include the intercept function, $\gamma_{k,0}(t)$. In this form, model (5.1) is a semi-parametric Gaussian mixture of varying-coefficients partially linear models (SPGMVCPLMs).
In what follows, we propose an estimation strategy to achieve the following objectives:

1. simultaneously estimate the local parameters in order to address label switching; and
2. select the optimal set of local grid points;

Towards achieving these objectives, we reformulate the problem as an incomplete-data problem. Consider a random sample $\{(t_i, \mathbf{x}_i, \mathbf{z}_i, y_i) : i = 1, 2, \ldots, n\}$ from model (5.1). We assume that each component regression parameter $\beta_k : k = 1, 2, \ldots, K$ is a non-parametric function of t and let $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$ be a set of N local points in the domain of the covariate t. Then, at each local point $u \in \mathcal{U}$, model (5.1) is a GMLRs model (2.19)

$$f_u(y|\mathbf{X} = \mathbf{x}) = \sum_{k=1}^{K} \pi_k(u) \mathcal{N}\Big(y|\mathbf{x}^{*\intercal} \boldsymbol{\eta}_k(u), \sigma_k^2(u)\Big), \tag{5.3}$$

where $\mathbf{x}^* = (1, \mathbf{x}, \mathbf{z})^\intercal$ and $\boldsymbol{\eta}_k(u) = (g_k(u), \boldsymbol{\beta}_k(u), \boldsymbol{\gamma}_k(u))^\intercal$.
One of these local GMLRs can be viewed as a conditional distribution of y, given $\mathbf{X} = \mathbf{x}$ and $\mathbf{Z} = \mathbf{z}$, that generated the n pairs $\{(\mathbf{x}_i, \mathbf{z}_i, y_i)\}_{i=1}^n$. Since we do not observe the identity of this local GMLRs model, conditional on $\mathbf{X} = \mathbf{x}$ and $\mathbf{Z} = \mathbf{z}$, y follows a mixture of these local GMLRs

$$\begin{aligned} f(y|\mathbf{X} = \mathbf{x}) &= \sum_{j=1}^{N} \lambda_j f_{u_j}(y|\mathbf{X} = \mathbf{x}) \\ &= \sum_{j=1}^{N} \lambda_j \left[\sum_{k=1}^{K} \pi_k(u_j) \mathcal{N}\Big(y|\mathbf{x}^{*\intercal} \boldsymbol{\eta}_k(u_j), \sigma_k^2(u_j)\Big) \right] \\ &= \sum_{j=1}^{N} \sum_{k=1}^{K} \lambda_j \pi_k(u_j) \mathcal{N}\Big(y|\mathbf{x}^{*\intercal} \boldsymbol{\eta}_k(u_j), \sigma_k^2(u_j)\Big), \end{aligned} \tag{5.4}$$

where $\lambda_j > 0$ (satisfying $\sum_{j=1}^{N} \lambda_j = 1$) is the mixing proportion, probability or weight. This parameter can be interpreted in various ways. As a mixing proportion, λ_j can be viewed as the relative number of data points that were generated by the j^{th} local GMLRs model. As a mixing probability, λ_j can be interpreted as the probability that a given data point, say $(\mathbf{x}_i, y_i)$, was generated by the j^{th} local GMLRs model. Thus, the larger the value of λ_j, the more data will be associated with the j^{th} local GMLRs model. Alternatively, the larger the value of λ_j, the more likely that a given data point was generated by the local model $f_{u_j}(y|\mathbf{X} = \mathbf{x})$. As a mixing weight, λ_j can be viewed as specifying the relative importance of the j^{th} local GMLRs model. The larger the weight, the greater the contribution made by the local model to the the overall model. Stated differently, a local model with a small weight ($\lambda_t \approx 0$) is indicative of a sparse local region. This is the case where a neighbourhood of a local point has few to no data points. This in turn implies that the local model has little to no information about the data and consequently about the overall model. Thus, the use of the corresponding local point is of little value to the overall fit of the model. This local point can therefore be removed because it may lead to an unreliable local estimate (Loader [1999]).

From the previous discussion, the benefits of the weights $(\lambda_1, \lambda_2, \ldots, \lambda_N)$ become apparent. They can be used in various innovative ways as we discuss below.

To estimate model (5.4), we first need to specify the set of local grid points $\mathcal{U}$. We can follow convention and use the observed covariate values or a set of equally-spaced values from the domain of the covariate. Alternatively, we can use the weights as follows: we begin by setting $\mathcal{U}$ as all the observed covariate values. Next, we modify the EM algorithm by introducing a step between the E- and M- step that determines all the weights that are below a certain threshold, say λ_0. Recall that the weights correspond with the local grid points. Thus, all the local grid points whose corresponding weights are below the threshold are removed and the algorithm continues with the remaining local grid points. We repeat the steps of this modified EM algorithm until convergence. The advantage of this approach is that it finds both the number and location of the grid points.

Another benefit of the weights is in suggesting an alternative approach to address label-switching. As mentioned before, estimating model (5.4) is equivalent to simultaneously estimating all the local parameters thus addressing label-switching. Moreover, the estimation can be done using the classical EM algorithm or the modified EM algorithm described above. An alternative strategy to addressing label-switching is to estimate all the local GMLRs and choose the one with the largest weight and use its resulting local responsibilities as the common responsibilities used to maximise each local-likelihood function. In this manner, the proposed approach is, in principle, similar to the objective-based approach proposed in chapter 4, where the objective function is $\max_j \lambda_j$.

Let $(\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{g}}, \hat{\boldsymbol{\sigma}}^2)$ be the first-stage estimates of the parametric and non-parametric terms $\boldsymbol{\beta}$ and $(\boldsymbol{\pi}, \boldsymbol{\gamma}, \boldsymbol{\sigma}^2, \boldsymbol{g})$, respectively, obtained from estimating (5.4). Note that when defining the mixture of GMLRs (5.4), we assumed that the global parameter $\boldsymbol{\beta}$ was local. The local estimate $\hat{\boldsymbol{\beta}}$, obtained from estimating model (5.4), can be improved by estimating $\boldsymbol{\beta}$ globally. Moreover, using the global estimate of $\boldsymbol{\beta}$, we can also improve the local estimates $(\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\sigma}}^2, \hat{\boldsymbol{g}})$. To achieve this, we propose one-step backfitting estimates $(\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{\sigma}}^2, \tilde{\boldsymbol{g}})$.
Given $(\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\sigma}}^2, \hat{\boldsymbol{g}})$, obtained from fitting model (5.4), let $\tilde{\boldsymbol{\beta}}$ be the estimates of the global parameters obtained by maximising the global log-likelihood function

$$\ell_1(\boldsymbol{\beta}) = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \hat{\pi}_k(t_i)\mathcal{N}(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_k + \mathbf{z}_i^\intercal\hat{\boldsymbol{\gamma}}_k(t_i) + \hat{g}_k(t_i), \hat{\sigma}_k^2(t_i))\right]. \tag{5.5}$$

Given $\tilde{\boldsymbol{\beta}}$, let $(\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{\sigma}}^2, \tilde{\boldsymbol{g}})$ be the estimate of the non-parametric functions $(\boldsymbol{\pi}, \boldsymbol{\gamma}, \boldsymbol{\sigma}^2, \boldsymbol{g})$ obtained by maximising the local log-likelihood function

$$\ell_2\{\boldsymbol{\theta}(u)\} = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{K} \pi_k(u)\mathcal{N}(y_i^*|\mathbf{z}_i^{*\intercal}\boldsymbol{\gamma}_k^*(u), \sigma_k^2(u))\right] \times K_h(t_i - u), \tag{5.6}$$

where $y_i^* = y_i - \mathbf{x}_i^\intercal\tilde{\boldsymbol{\beta}}_k$, $\mathbf{z}^* = (1, \mathbf{z})^\intercal$ and $\boldsymbol{\theta}(u) = (\boldsymbol{\pi}(u), \boldsymbol{\gamma}^*(u), \boldsymbol{\sigma}^2(u))$, with $\boldsymbol{\gamma}^*(u) = (\boldsymbol{g}(u), \boldsymbol{\gamma}(u))^\intercal$.
In summary, the proposed estimation procedure proceeds in two stages. In the first stage, we obtain $(\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{g}}, \hat{\boldsymbol{\sigma}}^2)$. Thereafter, in the second stage, we obtain $(\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{\sigma}}^2)$.
Note that since the set of local points $\mathcal{U}$ is determined by the range $\mathcal{T}$ of the covariate t, model (5.4) represents a reformulation of model (5.1). To simplify the notation, model (5.1) can be written as

$$f(y|\mathbf{X} = \mathbf{x}, T = t) = \sum_{j=1}^{N} \lambda_j \sum_{k=1}^{K} \pi_{j,k}\mathcal{N}\Big(y|\mathbf{x}^{*\intercal}\boldsymbol{\eta}_{j,k}, \sigma_{j,k}^2\Big), \tag{5.7}$$

where $\pi_{j,k} = \pi_k(u_j)$, $\boldsymbol{\eta}_{j,k} = \boldsymbol{\eta}_k(u_j)$ and $\sigma_{j,k}^2 = \sigma_k^2(u_j)$, for $j = 1, 2, \ldots, N$.
We now give details of the proposed estimation procedure to estimate model (5.7). Consider a random sample $\{(t_i, \mathbf{x}_i, y_i) : i = 1, 2, \ldots, n\}$ from model (5.7). The corresponding log-likelihood function is

$$\ell_0(\boldsymbol{\lambda}, \boldsymbol{\theta}) = \sum_{i=1}^{n} \log\left[\sum_{j=1}^{N}\sum_{k=1}^{K} \lambda_j\pi_{j,k}\mathcal{N}\Big(y|\mathbf{x}_i^{*\intercal}\boldsymbol{\eta}_{j,k}, \sigma_{j,k}^2\Big)\right], \tag{5.8}$$

where $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_N)$ and $\boldsymbol{\theta} = (\boldsymbol{\theta}(u_1), \boldsymbol{\theta}(u_2), \ldots, \boldsymbol{\theta}(u_N))$ with $\boldsymbol{\theta}(u_j) = (\boldsymbol{\pi}_{j\cdot}, \boldsymbol{\eta}_{j\cdot}, \boldsymbol{\sigma}^2_{j\cdot})$, $\boldsymbol{\pi}_{j\cdot} = (\pi_{j,1}, \pi_{j,2}, \ldots, \pi_{j,K})$, $\boldsymbol{\eta}_{j\cdot} = (\boldsymbol{\eta}_{j,1}, \boldsymbol{\eta}_{j,2}, \ldots, \boldsymbol{\eta}_{j,K})$ and $\boldsymbol{\sigma}^2_{j\cdot} = (\sigma^2_{j,1}, \sigma^2_{j,2}, \ldots, \sigma^2_{j,K})$, for $j = 1, 2, \ldots, N$.

We propose a modified Expectation Conditional Maximisation (ECM-) type algorithm (see section 2.1.2 of chapter 1) to maximise (5.8). Note that we now have two latent variables. The first latent variable serves as an indicator variable for the identity of the local model that generated a given data point. For each data point, we define this second latent variable as $\mathbf{v}_i = (v_{i1}, v_{i2}, \ldots, v_{iN})$ where $v_{ij} = 1$ if the i^{th} data point belongs or was generated by the j^{th} local model and 0 otherwise. The second latent variable is $\mathbf{z}_{ij}$. This serves as an indicator variable for the identity of the Gaussian component, from the j^{th} local model, that generated a given data point. Thus, $\mathbf{z}_{ij} = (z_{ij1}, z_{ij2}, \ldots, z_{ijK})$, where $z_{ijk} = 1$ if the i^{th} data point was generated by the k^{th} component from the j^{th} local mixture model. Given the completed-data $\{(t_i, \mathbf{x}_i, y_i, \mathbf{z}_{ij}, \mathbf{v}_i) : i = 1, 2, \ldots, n\}$, the corresponding (complete-data) log-likelihood is

$$\ell_0^c(\boldsymbol{\lambda}, \boldsymbol{\theta}) \quad = \quad \ell_0^{1c}(\boldsymbol{\lambda}) + \ell_0^{2c}(\boldsymbol{\theta}), \tag{5.9}$$

where

$$\begin{aligned}
\ell_0^{1c}(\boldsymbol{\lambda}) &= \sum_{i=1}^{n}\sum_{j=1}^{N} v_{ij} \log\lambda_j, \\
\ell_0^{2c}(\boldsymbol{\theta}) &= \sum_{i=1}^{n}\sum_{j=1}^{N}\sum_{k=1}^{K} v_{ij} z_{ijk} \Big[\log\pi_{j,k} + \log\mathcal{N}\big(y_i | \mathbf{x}_i^{*\intercal}\boldsymbol{\eta}_{j,k}, \sigma^2_{j,k}\big)\Big],
\end{aligned}$$

Let $\mathcal{T} = \{j | \lambda_j > \lambda_0\}$ be the set of all indices of the local models where the weights λ_j's are greater than some constant $0 < \lambda_0 < 1$. The constant λ_0 is a threshold that specifies a level beyond which a local point can be considered to have more influence on the estimation of the non-parametric functions, as discussed above. The threshold λ_0 is a free parameter (hyperparameter) that can be chosen either subjectively, by the data analyst, or objectively based on the observed data.

At the $(r+1)^{th}$ iteration of the E-step, we calculate the conditional expected value of $\ell_0^{1c}(\boldsymbol{\lambda})$ and $\ell_0^{2c}(\boldsymbol{\theta})$, denoted by $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$ and $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$, respectively, based on the conditional distribution of $\mathbf{v}_i$ and thereafter based on the conditional distribution of $\mathbf{z}_i$, given $\mathbf{v}_i$. This corresponds to estimating the latent variables v_{ij} and z_{ijk} using $\mathbb{E}[v_{ij}|t_i, \mathbf{x}_i, y_i, \lambda_j^{(r)}, \boldsymbol{\theta}^{(r)}(u_j)]$ and $\mathbb{E}[z_{ijk}|t_i, \mathbf{x}_i, y_i, \mathbf{v}_i, \boldsymbol{\theta}^{(r)}(u_j)]$, respectively. The conditional expectations are calculated as,

respectively,

$$\hat{v}_{ij}^{(r+1)} = \frac{\lambda_j^{(r)} \sum_{k=1}^{K} \lambda_{j,k}^{(r)} \mathcal{N}\left(y_i | \mathbf{x}_i^{*\intercal} \boldsymbol{\eta}_{j,k}^{(r)}, \sigma_{j,k}^{2(r)}\right)}{\sum_{\ell \in \mathcal{T}^{(r)}} \lambda_\ell^{(r)} \sum_{k=1}^{K} \lambda_{j,k}^{(r)} \mathcal{N}\left(y_i | \mathbf{x}_i^{*\intercal} \boldsymbol{\eta}_{j,k}^{(r)}, \sigma_{j,k}^{2(r)}\right)} \tag{5.10}$$

and

$$\hat{z}_{ijk}^{(r+1)} = \frac{\lambda_{j,k}^{(r)} \mathcal{N}\left(y_i | \mathbf{x}_i^{*\intercal} \boldsymbol{\eta}_{j,k}^{(r)}, \sigma_{j,k}^{2(r)}\right)}{\sum_{\ell=1}^{K} \lambda_{j,\ell}^{(r)} \mathcal{N}\left(y_i | \mathbf{x}_i^{*\intercal} \boldsymbol{\eta}_{j,\ell}^{(r)}, \sigma_{j,\ell}^{2(r)}\right)}. \tag{5.11}$$

Note that (5.11) is similar to (2.8), with the difference being that we now have to take into account the value of $\mathbf{v}_i$, for $i = 1, 2, \ldots, n$. Expression (5.10) $\hat{v}_{it}^{(r+1)}$ can be interpreted as the probability that the i^{th} data point was generated by the j^{th} local model. In other words, it represents the responsibility of the j^{th} local model for the i^{th} data point. Given that the i^{th} data point belongs to the j^{th} local model, $\hat{z}_{ijk}^{(r+1)}$ has the same interpretation as $\hat{p}_{ik}(u_j)$.
After replacing v_{ij} with $\hat{v}_{ij}^{(r+1)}$ and z_{ijk} with $\hat{z}_{ijk}^{(r+1)}$ in (5.9), we obtain $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$ and $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$. At the $(r+1)^{th}$ iteration of the first CM-step, we update $\lambda^{(r)}$, by maximising $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$, given $\mathcal{T}^{(r)}$, to obtain

$$\hat{\lambda}_j^{(r+1)} = \frac{\sum_{i=1}^{n} \hat{v}_{ij}^{(r+1)}}{n} \qquad \text{for } j \in \mathcal{T}^{(r)}. \tag{5.12}$$

To update $\mathcal{T}^{(r)}$, let

$$\mathcal{T}^{(r+1)} = \{j | \hat{\lambda}_j^{(r+1)} > \lambda_0\}. \tag{5.13}$$

At the second CM-step, we update $\boldsymbol{\theta}^{(r)}$ by maximising $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$. Note that if we maximise $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ with respect to, say $\pi_{j,k}$, for $j \in \mathcal{T}^{(r+1)}$, the resulting estimated function $\pi_k(t_i)$, for $i = 1, 2, \ldots, n$, may exhibit wild oscillations. This is because, at each local point u_j, the contribution of all the covariate values $\{t_1, t_2, \ldots, t_n\}$, to the expected complete-data log-likelihood function $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$, is equal. Thus, the local parameter estimate, say $\hat{\pi}_{j,k}$, will be sensitive to remote values of the covariate t.
To remedy this, we propose to maximise a kernel weighted version of the expected complete-

data log-likelihood function $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$

$$Q^{w}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}) = \sum_{j\in\mathcal{T}^{(r+1)}}\sum_{i=1}^{n}\sum_{k=1}^{K}\hat{v}_{ij}^{(r+1)}\hat{z}_{ijk}^{(r+1)}K_h(t_i-u_j)\Big[\log\pi_{j,k}+\log\mathcal{N}\big(y_i|\mathbf{x}_i^{*\intercal}\boldsymbol{\eta}_{j,k},\sigma_{j,k}^2\big)\Big] \tag{5.14}$$

where the kernel function $K_h(t_i-u_j)$ is used to provide a weight to t_i relative to the local point u_j. Note that if we choose $K_h(\cdot)$ as the uniform kernel function, the above problem persists. Thus, $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ is implicitly kernel weighted, where the kernel function is uniform.
Maximising (5.14) with respect to $\pi_{j,k}$, we get

$$\pi_{j,k}^{(r+1)} = \frac{\sum_{i=1}^{n}\hat{v}_{ij}^{(r+1)}\hat{z}_{ijk}^{(r+1)}K_h(x_i-u_j)}{\sum_{i=1}^{n}\hat{v}_{it}^{(r+1)}K_h(x_i-u_j)}. \tag{5.15}$$

Maximising (5.14) with respect to $\boldsymbol{\eta}_{j,k}$ and $\sigma_{j,k}^2$, we get

$$\boldsymbol{\eta}_{j,k}^{(r+1)} = \left(\sum_{i=1}^{n}w_{ijk}^{(r+1)}\mathbf{x}_i^*\mathbf{x}_i^{*\intercal}\right)^{-1}\left(\sum_{i=1}^{n}w_{ijk}^{(r+1)}\mathbf{x}_i^*y_i\right), \tag{5.16}$$

$$\sigma_{j,k}^{2(r+1)} = \frac{\sum_{i=1}^{n}w_{ijk}^{(r+1)}\left(y_i-\mathbf{x}_i^{*\intercal}\boldsymbol{\eta}_{j,k}^{(r+1)}\right)^2}{\sum_{i=1}^{n}w_{ijk}^{(r+1)}}, \tag{5.17}$$

where $w_{ijk}^{(r+1)} = \hat{v}_{ij}^{(r+1)}\hat{z}_{ijk}^{(r+1)}K_h(t_i-u_j)$.
We repeat the above E- and CM-steps until convergence.
Let $r=R$ be the iteration index at convergence. To obtain the first-stage estimates $\hat{\pi}_k(t_i)$, $\hat{\boldsymbol{\eta}}_k(t_i)$ and $\hat{\sigma}_k^2(t_i)$, for $i=1,2,\ldots,n$ and $k=1,2,\ldots,K$, we linearly interpolate over $\pi_{j,k}^{(R)}$, $\boldsymbol{\eta}_{j,k}^{(R)}$ and $\sigma_{j,k}^{2(R)}$, respectively, for $j\in\mathcal{T}^{(R)}$.
We refer to the above algorithm as the model-based ECM-type algorithm. Model-based because of its two-level clustering capabilities (at the local and global level) as well as its ability to choose the number and location of the grid points in a principled manner by making use of a probability distribution (model) and ECM algorithm because of its use of conditional maximisation.
Note the following properties of the model-based ECM-type algorithm:

Choice of λ_0: Intuitively, the value of λ_0 should not be too large because it might lead to the choice of an inadequately small (or zero!) number of local points. In the extreme case the algorithm will stop working. On the other hand, if λ_0 is chosen too small, the algorithm may not be effective in choosing the appropriate number and location of

the local grid points. The resulting local neighbourhood will include all the initial local points. However, the algorithm will still be effective in addressing label-switching, which is our main objective in this thesis;

Ascent property: An important and attractive property of the classical EM algorithm is the ascent property. That is, at each iteration $\ell_0^{(r+1)}(\boldsymbol{\lambda}, \boldsymbol{\theta}) \geq \ell_0^{(r)}(\boldsymbol{\lambda}, \boldsymbol{\theta})$. Empirical evidence, based on our simulations, shows that the model-based ECM-type algorithm also has this property. Among others, this property is useful for evaluating the convergence of the algorithm;

Convergence: The convergence of the algorithm can be evaluated in either one of the following ways: (1) Stop the algorithm when the increase in the likelihood from one iteration to the next is below some small pre-specified threshold. (2) Stop the algorithm when the change in the estimated parameters from one iteration to the next is smaller than some small value. For instance, $||\boldsymbol{\lambda}^{(r+1)} - \boldsymbol{\lambda}^{(r)}||_1 < 10^{-5}$ or $||\boldsymbol{\lambda}^{(r+1)} - \boldsymbol{\lambda}^{(r)}||_2 < 10^{-5}$, where $||\cdot||_1$ and $||\cdot||_2$ denotes the L_1 and L_2 norm, respectively, on $\mathbb{R}^{N^{(r+1)}}$. The notation, $N^{(r+1)}$, is used to denote the number of local grid points at the $(r+1)^{th}$ iteration.

In the second-stage, we propose an updated estimate, $\tilde{\boldsymbol{\beta}}$, of the global parameter $\boldsymbol{\beta}$, obtained by maximising the global log-likelihood function (5.5). We make use of this global parameter estimate to improve the estimates of the non-parametric functions $(\boldsymbol{\pi}, \boldsymbol{\gamma}, \boldsymbol{g}, \boldsymbol{\sigma}^2)$. To do this, we propose the estimates $(\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{\sigma}}^2)$ obtained by maximising (5.6). Note that the global parameter estimate $\tilde{\boldsymbol{\beta}}$ is well labelled. This implies that the local log-likelihood functions (5.6) can be maximised separately without being concerned about label switching. Incidentally, note that the local likelihood function (5.6) corresponds to a NPGMVCMs (1.19). Thus, this is the same as estimating model (1.19). Let $(\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{\sigma}}^2)$ be the one-step backfitting estimates of $(\boldsymbol{\pi}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \boldsymbol{g}, \boldsymbol{\sigma}^2)$.

Note the difference between the proposed model-based strategy to address label-switching and the objective-based approach proposed in chapter 4. In the objective-based approach, we address label-switching by first obtaining a global set of responsibilities and then using it to maximise each local-likelihood function to estimate the local parameters. In the model-based approach, all the local parameters are contained in the model (5.4). Therefore, we only need to maximise (5.8) to estimate all the local parameters.

5.2 One-step backfitting algorithm

In this section, we propose an algorithm to carry out the above two-stage estimation procedure.

Stage 0: Initialising the algorithm

Obtain appropriate initial estimates of both the global parameter and the non-parametric functions, denoted $\boldsymbol{\beta}^{(0)}$ and $(\boldsymbol{\pi}^{(0)}, \boldsymbol{\gamma}^{(0)}, \boldsymbol{g}^{(0)}, \boldsymbol{\sigma}^{2(0)})$, respectively. Let $\mathcal{U}$ be the set of N grid points, $\mathcal{T}^{(0)} = \{1, 2, \ldots, N\}$ be the initial set of indices and specify λ_0.

Stage 1: Model-based ECM-type algorithm to maximise (5.8)

Let $\boldsymbol{\lambda}^{(r)}$ and $\boldsymbol{\theta}^{(r)}$ be the parameter estimates obtained at the r^{th} iteration.
E-Step: At the $(r+1)^{th}$ iteration, calculate $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$ and $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ by first estimating $\mathbf{v}_i$ and $\mathbf{z}_i$, for $i = 1, 2, \ldots, n$, using (5.10) and (5.11), respectively.

CM-Step 1: Maximise $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$ to obtain $\boldsymbol{\lambda}^{(r+1)}$ and $\mathcal{T}^{(r+1)}$ using (5.12) and (5.13), respectively.

CM-Step 2: Given $\mathcal{T}^{(r+1)}$, maximise $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ to obtain
$\boldsymbol{\theta}^{(r+1)} = (\pi_{j,k}^{(r+1)}, \boldsymbol{\eta}_{j,k}^{(r+1)}, \sigma_{j,k}^{2(r+1)})_{j \in \mathcal{T}^{(r+1)}, 1 \leq k \leq K}$ using (5.15), (5.16) and (5.17).
Repeat the above E- and CM-steps until convergence.

Stage 2(a): EM algorithm to maximise ℓ_1 in (5.5)

Given $(\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\gamma}}, \hat{\boldsymbol{\sigma}}^2, \hat{\boldsymbol{g}})$ obtained from Stage 1, we obtain the global estimate $\tilde{\boldsymbol{\beta}}$ of the global parameter $\boldsymbol{\beta}$ by maximising ℓ_1 in (5.5) using the usual EM algorithm.

E-Step: At the $(r+1)^{th}$ iteration, calculate the expected value of the latent variable as

$$p_{ik}^{(r+1)} = \frac{\hat{\pi}_k(t_i)\mathcal{N}(y_i^*|\mathbf{x}_i^\intercal \boldsymbol{\beta}_k^{(r)}, \hat{\sigma}_k^2(t_i))}{\sum_{\ell=1}^{K} \hat{\pi}_\ell(t_i)\mathcal{N}(y_i^*|\mathbf{x}_i^\intercal \boldsymbol{\beta}_\ell^{(r)}, \hat{\sigma}_\ell^2(t_i))\})}, \tag{5.18}$$

where $y_i^* = y_i - \mathbf{z}_i^{*\intercal}\hat{\boldsymbol{\gamma}}_k(t_i) - \hat{g}_k(t_i)$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$.
M-Step: We obtain $\boldsymbol{\beta}_k^{(r+1)}$ as

$$\boldsymbol{\beta}_k^{(r+1)} = \left(\sum_{i=1}^{n} p_{ik}^{(r+1)} \mathbf{x}_i \mathbf{x}_i^\intercal \right)^{-1} \left(\sum_{i=1}^{n} p_{ik}^{(r+1)} \mathbf{x}_i y_i^* \right). \tag{5.19}$$

Repeat the above E- and M-step until convergence

Stage 2(b): EM algorithm to maximise ℓ_2 in (5.6)

Given $\tilde{\boldsymbol{\beta}}$ obtained from Stage 2(a), we propose an improved estimate of the component non-parametric functions, denoted by $(\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\gamma}}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{\sigma}}^2)$, obtained by separately maximising the local log-likelihood functions in (5.6) using the usual EM algorithm.

E-Step: At the $(r+1)^{th}$ iteration, calculate the expected value of the latent variable as

$$p_{ik}^{(r+1)}(u) = \frac{\pi_k^{(r)}(u)\mathcal{N}(y_i^*|\mathbf{z}_i^{*\mathsf{T}}\boldsymbol{\gamma}_k^{*(r)}(u), \sigma_k^{2(r)}(u))}{\sum_{\ell=1}^{K}\pi_\ell^{(r)}(u)\mathcal{N}(y_i^*|\mathbf{z}_i^{*\mathsf{T}}\boldsymbol{\gamma}_\ell^{*(r)}(u), \sigma_\ell^{2(r)}(u))}, \tag{5.20}$$

where $y_i^* = y_i - \mathbf{x}_i^{\mathsf{T}}\tilde{\boldsymbol{\beta}}_k$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$.
M-Step: We obtain $\pi_k^{(r+1)}(u)$, for $u \in \mathcal{U}$, using the LPL estimator $\hat{\pi}_{k0}(u)$ obtained as the maximiser of

$$\sum_{i=1}^{n} p_{ik}^{(r+1)}(u)K_h(t_i - u)\log\left\{\sum_{j=0}^{p}\pi_{kj}(u)[t_i - u]^j\right\} \tag{5.21}$$

Note that, for $p > 0$, the LPL estimator of $\pi_k^{(r+1)}(u)$ does not have a closed form expression. Thus, $p = 0$ in (5.21).
Next, we obtain $\boldsymbol{\gamma}_k^{*(r+1)}(u)$ and $\sigma_k^{2(r+1)}(u)$, for $u \in \mathcal{U}$, using the LPL estimators $(\hat{\gamma}_{k,b,0}^*(u))_{1\le b\le D_2}$, and $\hat{\sigma}_{k0}^2(u)$, respectively, obtained as the maximisers of

$$\sum_{i=1}^{n} w_{ik}^{(r+1)}(u)\log\mathcal{N}\left(y_i^*|\sum_{b=1}^{D_2}\left(\sum_{j=0}^{p}\gamma_{k,b,j}^*(u)[t_i - u]^j\right)z_{ib}^*, \sigma_k^{2(r)}(u)\right), \tag{5.22}$$

$$\sum_{i=1}^{n} w_{ik}^{(r+1)}(u)\log\mathcal{N}\left(y_i^*|\mathbf{z}_i^{*\mathsf{T}}\boldsymbol{\gamma}_k^{*(r)}(t_i), \sum_{j=0}^{p}\sigma_{kj}^2(u)[t_i - u]^j\right), \tag{5.23}$$

where $w_{ik}^{(r+1)}(u) = p_{ik}^{(r+1)}(u)K_h(t_i - u)$ and $\boldsymbol{\gamma}_k^{*(r+1)}(t_i)$ is the estimated vector of regression coefficients, for $i = 1, 2, \ldots, n$, obtained from evaluating (5.22). Note that, for $p > 0$, the LPL estimator of $\sigma_k^{2(r+1)}(u)$ is not guaranteed to be positive. However, it can be truncated to the positive interval using an appropriate transformation.
Repeat the above E- and M-step until convergence.
At convergence of the EM algorithm of Stage 2(b), we obtain $\tilde{\pi}_k(t_i)$, $\tilde{\gamma}_k(t_i)$, $\tilde{g}_k(t_i)$ and $\tilde{\sigma}^2(t_i)$, for $t_i \notin \mathcal{U}$, by linearly interpolating over $\pi_k^{(R)}(u)$, $\sigma_k^{2(R)}(u)$, $\boldsymbol{\gamma}_k^{(R)}(u)$ and $g_k^{(R)}(u)$, for all $u \in \mathcal{U}$ and $k = 1, 2, \ldots, K$.

5.3 Essays: Model-based approach

In this section, we demonstrate the effectiveness and practical utility of the proposed model-based approach to address label-switching. The format of the presentation of this section is the same as in section 4.2. However, the essays in this section will be based on different models than those used in section 4.2. The first essay (subsection 5.3.1) estimates model (1.4) and the second essay (subsection 5.3.2) estimates model (1.17).

5.3.1 Semi-parametric Gaussian mixtures of non-parametric regressions (SPGMNRs)

The GMLRs (2.19) is a useful tool for studying the relationship between a response y and a set of covariates $\mathbf{x}$ whenever the underlying population consists of an a priori known number of subpopulations whose size and composition is unknown and unobserved. As already mentioned in section 2.1.4, the model enjoys widespread use in the social and biological sciences, among many other fields. However, the model makes an oft unrealistic assumption by assuming that the CRFs are linear functions of the covariates. As mentioned in chapter 4, in the modern world, associations between different factors or variables deviate significantly from linearity. The practitioner favours the linearity assumption because of its simplicity of structure and the ease of interpreting the effect of each covariate. The SPGMAMs (1.11) retains both these features while relaxing the linearity assumption entirely. The model replaces each parametric function of a covariate $\beta_{k,j}x_j$ with a univariate non-parametric function $g(x_j)$. Thus, model (1.11) is more flexible than the model (2.19). The model (1.11) will assist in discovering the appropriate form of each univariate function. Even if the linearity assumption is appropriate, the model (1.11) will confirm it. Thus, it's generality cannot be understated.
In this essay, we demonstrate the proposed model-based approach for estimating model (1.11). For illustrative purposes, we consider the case of a single covariate, that is $\mathbf{t} \in \mathbb{R}^1$. Thus, the model (1.11) reduces to the model (1.4). We will use simulated data and real data to show the performance and practical utility of the proposed approach.

Estimation procedure

Consider a random sample $\{(t_i, y_i) : i = 1, 2, \ldots, n\}$ from model (1.4). Assume that the parameters π_k and σ_k^2, for $k = 1, 2, \ldots, K$, are non-parametric functions of t and let $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$ be a set of N local points in the domain of the covariate t. Then, locally,

model (1.4) is a GMM (2.1)

$$f_u(y) = \sum_{k=1}^{K} \pi_k(u)\mathcal{N}\Big(y|m_k(u), \sigma_k^2(u)\Big). \tag{5.24}$$

One of these local GMMs can be viewed as a distribution of the response variable y. Since we do not observe the identity of this local GMM, y follows a mixture of these local GMMs

$$\begin{aligned} f(y) &= \sum_{j=1}^{N} \lambda_j f_{u_j}(y) \\ &= \sum_{j=1}^{N} \lambda_j \left[\sum_{k=1}^{K} \pi_k(u_j)\mathcal{N}\Big(y|m_k(u_j), \sigma_k^2(u_j)\Big)\right] \\ &= \sum_{j=1}^{N}\sum_{k=1}^{K} \lambda_j \pi_k(u_j)\mathcal{N}\Big(y|m_k(u_j), \sigma_k^2(u_j)\Big), \end{aligned} \tag{5.25}$$

where $\lambda_j > 0$ (satisfying $\sum_{j=1}^{N} \lambda_j = 1$) is the mixing proportion, probability or weight.
Model (5.25) represents a reformulation of model (1.4). The model can be estimated directly using maximum likelihood via a modified ECM-type algorithm. By estimating model (5.25), we are in effect simultaneously estimating the local parameters, thus avoiding the label-switching problem. To simplify the notation, model (5.25) can be written as

$$f(y|T = t) = \sum_{j=1}^{N} \lambda_j \sum_{k=1}^{K} \pi_{j,k}\mathcal{N}\Big(y|m_{j,k}, \sigma_{j,k}^2\Big), \tag{5.26}$$

where $\pi_{j,k} = \pi_k(u_j)$, $m_{j,k} = m_k(u_j)$ and $\sigma_{j,k}^2 = \sigma_k^2(u_j)$.
The log-likelihood function with respect to model (5.26) is

$$\ell_0(\boldsymbol{\lambda}, \boldsymbol{\theta}) = \sum_{i=1}^{n} \log\left[\sum_{j=1}^{N}\sum_{k=1}^{K} \lambda_j \pi_{j,k}\mathcal{N}\Big(y|m_{j,k}, \sigma_{j,k}^2\Big)\right], \tag{5.27}$$

where $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_N)$ and $\boldsymbol{\theta} = (\boldsymbol{\theta}(u_1), \boldsymbol{\theta}(u_2), \ldots, \boldsymbol{\theta}(u_N))$ with $\boldsymbol{\theta}(u_j) = (\boldsymbol{\pi}_{j\cdot}, \mathbf{m}_{j\cdot}, \boldsymbol{\sigma}_{j\cdot}^2)$, $\boldsymbol{\pi}_{j\cdot} = (\pi_{j,1}, \pi_{j,2}, \ldots, \pi_{j,K})$, $\mathbf{m}_{j\cdot} = (m_{j,1}, m_{j,2}, \ldots, m_{j,K})$ and $\boldsymbol{\sigma}_{j\cdot}^2 = (\sigma_{j,1}^2, \sigma_{j,2}^2, \ldots, \sigma_{j,K}^2)$, for $j = 1, 2, \ldots, N$.
To maximise (5.27), we use a modified ECM-type algorithm as before. At the $(r+1)^{th}$ iteration of the E-step, we calculate the conditional expectations of the complete-data log-likelihoods $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$ and $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ with respect to the conditional distribution of $\mathbf{v}$ and $\mathbf{z}$. This corresponds to estimating the latent variables v_{ij} and z_{ijk} for $i = 1, 2, \ldots, n$, $k = 1, 2, \ldots, K$

and $j \in \mathcal{T}^{(r)}$ using $\mathbb{E}[v_{ij}|t_i, y_i, \lambda_j^{(r)}, \boldsymbol{\theta}^{(r)}(u_j)]$ and $\mathbb{E}[z_{ijk}|t_i, y_i, \mathbf{v}_i, \boldsymbol{\theta}^{(r)}(u_j)]$, respectively. The conditional expectations are calculated, respectively, as

$$\hat{v}_{ij}^{(r+1)} = \frac{\lambda_j^{(r)} \sum_{k=1}^{K} \pi_{j,k}^{(r)} \mathcal{N}\left(y_i | m_{j,k}^{(r)}, \sigma_{j,k}^{2(r)}\right)}{\sum_{\ell \in \mathcal{T}^{(r)}} \lambda_\ell^{(r)} \sum_{k=1}^{K} \pi_{\ell,k}^{(r)} \mathcal{N}\left(y_i | m_{\ell,k}^{(r)}, \sigma_{\ell,k}^{2(r)}\right)} \tag{5.28}$$

and

$$\hat{z}_{ijk}^{(r+1)} = \frac{\pi_{j,k}^{(r)} \mathcal{N}\left(y_i | m_{j,k}^{(r)}, \sigma_{j,k}^{2(r)}\right)}{\sum_{\ell=1}^{K} \pi_{j,\ell}^{(r)} \mathcal{N}\left(y_i | m_{j,\ell}^{(r)}, \sigma_{j,\ell}^{2(r)}\right)}. \tag{5.29}$$

Note that (5.29) is similar to (1.25), with the difference being that we now have to take into account the value of $\mathbf{v}_i$, for $i = 1, 2, \ldots, n$. Expression (5.28) $\hat{v}_{ij}^{(r+1)}$ can be interpreted as the probability that the i^{th} data point was generated by the j^{th} local model. In other words, it represents the responsibility of the j^{th} local model for the i^{th} data point. Given that the i^{th} data point belongs to the j^{th} local model, $\hat{z}_{ijk}^{(r+1)}$ has the same interpretation as $\gamma_{ik}(u)$.

At the first CM-step, on the $(r+1)^{th}$ iteration, we update $\lambda^{(r)}$. The ECM update equation is the same as (5.12). At the second CM-step, we update $\boldsymbol{\theta}^{(r)}$ by maximising $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$, respectively. Note that if we maximise $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$, with respect to, say $m_{j,k}$, for $j \in \mathcal{T}^{(r+1)}$, the resulting estimated function $m_k(t_i)$, for $i = 1, 2, \ldots, n$, may exhibit wild oscillations. This is because, at each local point u_j, the contribution of all the covariate values $\{t_1, t_2, \ldots, t_n\}$ to the likelihood function is equal. Thus, the local parameter estimate, say $\hat{m}_{j,k}$, will be sensitive to remote values of the covariate.

To remedy this, we propose to maximise kernel weighted versions of these complete-data log-likelihood functions

$$Q^w(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)}) = \sum_{j \in \mathcal{T}^{(r+1)}} \sum_{i=1}^{n} \sum_{k=1}^{K} \hat{v}_{ij}^{(r+1)} \hat{z}_{itk}^{(r+1)} K_h(x_i - u_j) \left[\log \pi_{j,k} + \log \mathcal{N}\left(y_i | m_{j,k}, \sigma_{j,k}^2\right)\right]. \tag{5.30}$$

Maximising (5.30) with respect to $\pi_{j,k}$, we get

$$\pi_{j,k}^{(r+1)} = \frac{\sum_{i=1}^{n} \hat{v}_{ij}^{(r+1)} \hat{z}_{ijk}^{(r+1)} K_h(t_i - u_j)}{\sum_{i=1}^{n} \hat{v}_{ij}^{(r+1)} K_h(t_i - u_j)}. \tag{5.31}$$

Maximising (5.30) with respect to $m_{j,k}$ and $\sigma^2_{j,k}$ we get

$$m^{(r+1)}_{j,k} = \frac{\sum_{i=1}^{n} w^{(r+1)}_{ijk} y_i}{\sum_{i=1}^{n} w^{(r+1)}_{ijk}}, \tag{5.32}$$

$$\sigma^{2(r+1)}_{j,k} = \frac{\sum_{i=1}^{n} w^{(r+1)}_{ijk} (y_i - m^{(r+1)}_{j,k})^2}{\sum_{i=1}^{n} w^{(r+1)}_{ijk}}, \tag{5.33}$$

where $w^{(r+1)}_{ijk} = \hat{v}^{(r+1)}_{ij} \hat{z}^{(r+1)}_{ijk} K_h(t_i - u_j)$.
We repeat the above E- and CM-steps until convergence.
Let $\boldsymbol{\theta}^{(R)}$ and $\mathcal{T}^{(R)}$ be the resulting parameter estimates and the set of indices of the local points, respectively, obtained at convergence of the above ECM algorithm. To obtain $(\hat{\pi}_k(t_i), \hat{m}_k(t_i), \hat{\sigma}^2_k(t_i))_{1\leq i\leq n, 1\leq k\leq K}$, we linearly interpolate over $(\pi^{(R)}_{j,k}, m^{(R)}_{j,k}, \sigma^{2(R)}_{j,k})_{1\leq k\leq K, j\in\mathcal{T}^{(R)}}$.
Note that $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}^2$ are global parameters but their first-stage estimators are non-parametric (local). In the second-stage, we improve the first-stage estimates $\hat{\boldsymbol{\pi}}$ and $\hat{\boldsymbol{\sigma}}^2$ by estimating the global parameters ($\boldsymbol{\pi}$ and $\boldsymbol{\sigma}^2$) globally. Given $\hat{\mathbf{m}}$, we propose updated estimates $\tilde{\boldsymbol{\pi}}$ and $\tilde{\boldsymbol{\sigma}}^2$ obtained by maximising the global log-likelihood function

$$\ell_1(\boldsymbol{\pi}, \boldsymbol{\sigma}^2) = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{n} \pi_k \mathcal{N}(y_i | \hat{m}_k(t_i), \sigma^2_k)\right]. \tag{5.34}$$

We can make use of these global parameter estimates ($\tilde{\boldsymbol{\pi}}$ and $\tilde{\boldsymbol{\sigma}}^2$) to improve the first-stage estimates of the CRFs $\hat{\mathbf{m}}$. Given $\tilde{\boldsymbol{\pi}}$ and $\tilde{\boldsymbol{\sigma}}^2$, we propose the estimate $\tilde{\mathbf{m}}$ obtained by maximising the local log-likelihood function, for $u \in \mathcal{U}$,

$$\ell_2[\mathbf{m}(u)] = \sum_{i=1}^{n} \log\left\{\sum_{k=1}^{K} \tilde{\pi}_k \mathcal{N}(y_i | m_k(u), \tilde{\sigma}^2_k)\right\} K_h(t_i - u). \tag{5.35}$$

Note that the global parameter estimates $\tilde{\boldsymbol{\pi}}$ and $\tilde{\boldsymbol{\sigma}}^2$ are well labelled. This implies that each local log-likelihood function (5.35) can be maximised separately without being concerned about label switching. Let $\tilde{\boldsymbol{\theta}} = (\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\sigma}}^2, \tilde{\mathbf{m}})$ be the one-step backfitting estimate of $\boldsymbol{\theta}$.
In summary, the proposed estimation procedure proceeds in two stages. In the first stage, we obtain $\hat{\boldsymbol{\pi}}$, $\hat{\boldsymbol{\sigma}}^2$ and $\hat{\mathbf{m}}$. Thereafter, in the second stage, we obtain $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\sigma}}^2$ and $\tilde{\mathbf{m}}$.
The following is a one-step backfitting algorithm that can be used to carry out the above two-stage estimation procedure.

Stage 0: Initialising the algorithm Obtain appropriate initial estimates of the global parameters and the non-parametric functions, denoted $(\boldsymbol{\pi}^{(0)}, \boldsymbol{\sigma}^{2(0)})$ and $\mathbf{m}^{(0)}$, respectively, by

making use of, say a mixture of regression splines (see Xiang and Yao [2016]). Let $\mathcal{U}$ be the set of N grid points, $\mathcal{T}^{(0)} = \{1, 2, \ldots, N\}$ be the initial set of indices and specify λ_0.

Stage 1: Model-based ECM-type algorithm to maximise ℓ_0 in (5.27) Let $\boldsymbol{\lambda}^{(r)}$ and $\boldsymbol{\theta}^{(r)}$ be the parameter estimates obtained at the r^{th} iteration.
E-Step: At the $(r+1)^{th}$ iteration, calculate the expected complete-data log-likelihoods $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$ and $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ by first estimating v_{ij} and z_{ijk}, for $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, N$ and $k = 1, 2, \ldots, K$, using (5.28) and (5.29), respectively.

CM-Step 1: Maximise $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$ to obtain $\boldsymbol{\lambda}^{(r+1)}$ and $\mathcal{T}^{(r+1)}$ using (5.12) and (5.13), respectively.

CM-Step 2: Given $\mathcal{T}^{(r+1)}$, maximise $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ to obtain $\boldsymbol{\theta}^{(r+1)} = (\pi_{j,k}^{(r+1)}, m_{j,k}^{(r+1)}, \sigma_{j,k}^{2(r+1)})_{j\in\mathcal{T}^{(r+1)}, 1\leq k\leq K}$ using (5.31), (5.32) and (5.33).
Repeat the above E- and CM-steps until convergence.

Stage 2(a): EM algorithm to maximise ℓ_1 in (5.34)

Given $\hat{\mathbf{m}}$ obtained from Stage 1, we obtain the global estimates $\tilde{\boldsymbol{\pi}}$ and $\tilde{\boldsymbol{\sigma}}^2$ of the global parameters $\boldsymbol{\pi}$ and $\boldsymbol{\sigma}^2$, respectively, by maximising ℓ_1 in (5.34) using the usual EM algorithm.

E-Step: At the $(r+1)^{th}$ iteration, calculate the expected value of the latent variable as

$$p_{ik}^{(r+1)} = \frac{\pi_k^{(r)} \mathcal{N}(y_i|\hat{m}_k(t_i), \sigma_k^{2(r)})}{\sum_{\ell=1}^{K} \pi_\ell^{(r)} \mathcal{N}(y_i|\hat{m}_\ell(t_i), \sigma_\ell^{2(r)})} \tag{5.36}$$

M-Step: We obtain $\pi_k^{(r+1)}$ and $\sigma_k^{2(r+1)}$, respectively, using the equations

$$\pi_k^{(r+1)} = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)}}{n}, \tag{5.37}$$

$$\sigma_k^{2(r+1)} = \frac{\sum_{i=1}^{n} p_{ik}^{(r+1)} \big(y_i - \hat{m}_k(t_i)\big)^2}{\sum_{i=1}^{n} p_{ik}^{(r+1)}}. \tag{5.38}$$

Repeat the above E- and M-step until convergence

Stage 2(b): EM algorithm to maximise ℓ_2 in (5.35)

Given $\tilde{\boldsymbol{\pi}}$ and $\tilde{\boldsymbol{\sigma}}^2$ obtained from Stage 2(a), we propose an improved estimate of the component non-parametric functions, denoted by $\tilde{\mathbf{m}}$, obtained by maximising each local log-likelihood

function in (5.35) using the usual EM algorithm.

E-Step: At the $(r+1)^{th}$ iteration, calculate the expected value of the latent variable as

$$p_{ik}^{(r+1)}(u) = \frac{\tilde{\pi}_k \mathcal{N}(y_i | m_k^{(r)}(u), \tilde{\sigma}_k^2)}{\sum_{\ell=1}^{K} \tilde{\pi}_\ell \mathcal{N}(y_i | m_\ell^{(r)}(u), \tilde{\sigma}_\ell^2)} \tag{5.39}$$

M-Step: We obtain $m_k^{(r+1)}(u)$, for $u \in \mathcal{U}$, using the LPL estimator $\hat{m}_{k0}(u)$, obtained as the maximiser of

$$\sum_{i=1}^{n} w_{ik}^{(r+1)}(u) \log \mathcal{N}\left(y_i | \sum_{j=0}^{p} m_{kj}(u)[t_i - u]^j, \tilde{\sigma}_k^2 \right) \tag{5.40}$$

where $w_{ik}^{(r+1)}(u) = p_{ik}^{(r+1)}(u) K_h(t_i - u)$.
Repeat the above E- and M-step until convergence.

At convergence of the EM algorithm of Stage 2(b), we obtain $\tilde{\mathbf{m}} = (\tilde{\mathbf{m}}_1, \tilde{\mathbf{m}}_2, \ldots, \tilde{\mathbf{m}}_K)$, where $\tilde{\mathbf{m}}_k = (\tilde{m}_k(t_1), \tilde{m}_k(t_2), \ldots, \tilde{m}_k(t_n))$ by linear interpolation over $m_k^{(R)}(u)$ for $u \in \mathcal{U}$ and $k = 1, 2, \ldots, K$.

Simulations

In this section, we perform numerical experiments to demonstrate the performance of the proposed method. The purpose of these experiments is two fold. First, we want to demonstrate the effectiveness of the proposed method towards addressing label-switching. Second, we want to evaluate the accuracy of the proposed one-step backfitting estimators and the fully iterative estimators. Moreover, we want to demonstrate the practical suitability of the fitted model based on these estimators. For the remainder of the chapter, we refer to the proposed model-based algorithm as the MB-ECM algorithm. Since for the SPGMNRs model (1.4), the CRFs are the only non-parametric terms in the model, we estimate these functions using the LLEs and compare them with the estimates based on the LCEs. All the programming and subsequent simulations are performed in the R programming language R Core Team [2023].

Choosing the bandwidth h Among other things, local polynomial fitting requires the bandwidth, h. In practice, this component is usually chosen using a data-driven approach such as cross-validation (CV) (see chapter 7 of Hastie et al. [2009] for more details). In this paper, we propose a generalised CV approach (see Wahba [1977] and Craven and Wahba [1979]) for

bandwidth selection.
Let $\hat{\mathbf{y}}_k = (\hat{y}_{1k}, \ldots, \hat{y}_{nk})^\intercal$ be the vector of fitted values, where $\hat{y}_{ik} = \tilde{m}_k(x_i)$ is the one-step backfitting estimate of $m_k(x_i)$. This can be expressed as

$$\hat{\mathbf{y}}_k = \mathbf{S}_{hk}\mathbf{y}, \quad \text{for } k = 1, 2, \ldots, K, \tag{5.41}$$

where $\mathbf{y} = (y_1, y_2, \ldots, y_n)$ and $\mathbf{S}_{hk}$ is known as the smoother matrix, see Buja et al. [1989] for more details. The first subscript shows that the smoother matrix depends on the bandwidth h, among others. We propose the following GCV error

$$\text{GCV}(h) = \sum_{k=1}^{K} \frac{(\mathbf{y} - \hat{\mathbf{y}}_k)^\intercal \mathbf{W}_k (\mathbf{y} - \hat{\mathbf{y}}_k)/n_k}{(1 - \text{df}_k/n_k)^2} \tag{5.42}$$

$$= \sum_{k=1}^{K} \frac{\text{ASE}_k}{(1 - \text{df}_k/n)^2}, \tag{5.43}$$

where $\text{ASE}_k = (\mathbf{y} - \hat{\mathbf{y}}_k)^\intercal \mathbf{W}_k (\mathbf{y} - \hat{\mathbf{y}}_k)/n_k$, with $n_k = \sum_{i=1}^{n} \hat{\gamma}_{ik}$, is the average squared error (ASE) of the fitted k^{th} CRF, $\mathbf{W}_k = \text{diag}(\hat{\gamma}_{1k}, \hat{\gamma}_{2k}, \ldots, \hat{\gamma}_{nk})$ is the diagonal matrix of the responsibilities of the k^{th} component obtained based on the one-step backfitting estimates $\tilde{\boldsymbol{\theta}} = (\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\sigma}}^2, \tilde{\mathbf{m}})$ and

$$\text{df}_k = \text{trace}(\mathbf{S}_{hk}) = \sum_{i=1}^{n} s_{ii}, \tag{5.44}$$

where s_{ii}, for $i = 1, 2, \ldots, n$, are the diagonal entries of the smoother matrix $\mathbf{S}_{hk}$. Expression (5.44) denotes the degrees of freedom of the k^{th} component. From subsection 2.2.1, we know that df_k quantifies the complexity of the k^{th} fitted CRF. This concept is very useful for comparing local polynomial estimates of different degrees. We will demonstrate this in our simulation study.

Performance measures To evaluate the goodness of the overall fitted model and the fitted parameters and CRFs, based on the proposed one-step backfitting estimates, we make use of the following measures.
As above, let $\hat{y}_{ik}$ and $\hat{\gamma}_{ik}$, for $i = 1, 2, \ldots, n$ and $k = 1, 2, \ldots, K$, be the fitted values and responsibilities, respectively, based on the one-step backfitting estimates $\tilde{\theta}$. Moreover, let $f_{\boldsymbol{\theta}}$ and $F_{\boldsymbol{\theta}}$ be the true conditional probability distribution (1.4) and the corresponding cumulative conditional probability distribution, respectively. Let $\hat{f}_{\tilde{\boldsymbol{\theta}}}$ and $\hat{F}_{\tilde{\boldsymbol{\theta}}}$ be the respective estimates.

Root average squared error (RASE) The root average squared error (RASE) is the most used measure to assess the adequacy of a fitted model. For model (1.4), we define the

total RASE as

$$\text{RASE}^2(y) = \sum_{k=1}^{K} \text{ASE}_k, \tag{5.45}$$

where $\text{ASE}_k = \frac{1}{n_k}\sum_{i=1}^{n} \hat{\gamma}_{ik}(y_i - \hat{y}_{ik})^2$ with $n_k = \sum_{i=1}^{n} \hat{\gamma}_{ik}$.
We also use the RASE to measure the accuracy of the fitted CRFs $\tilde{\mathbf{m}}$ and $\hat{f}_{\tilde{\boldsymbol{\theta}}}$, respectively, as

$$\text{RASE}^2(\mathbf{m}) = \frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}\left[\tilde{m}_k(x_i) - m_k(x_i)\right]^2, \tag{5.46}$$

$$\text{RASE}^2(f_{\boldsymbol{\theta}}) = \frac{1}{n}\sum_{i=1}^{n}\left[\hat{f}_{\tilde{\boldsymbol{\theta}}}(y_i|x_i) - f_{\boldsymbol{\theta}}(y_i|x_i)\right]^2. \tag{5.47}$$

Finally, to evaluate the goodness of the estimated component parameters $\tilde{\boldsymbol{\phi}}$, we make use of

$$\text{ASE}(\boldsymbol{\phi}_k) = (\tilde{\phi}_k - \phi_k)^2, \tag{5.48}$$

where ϕ_k can be either π_k or σ_k^2.

Adjusted Rand Index (ARI) In order to evaluate the clustering ability of the fitted model, we make use of the adjusted rand index (ARI; Hubert and Arabie [1985]).

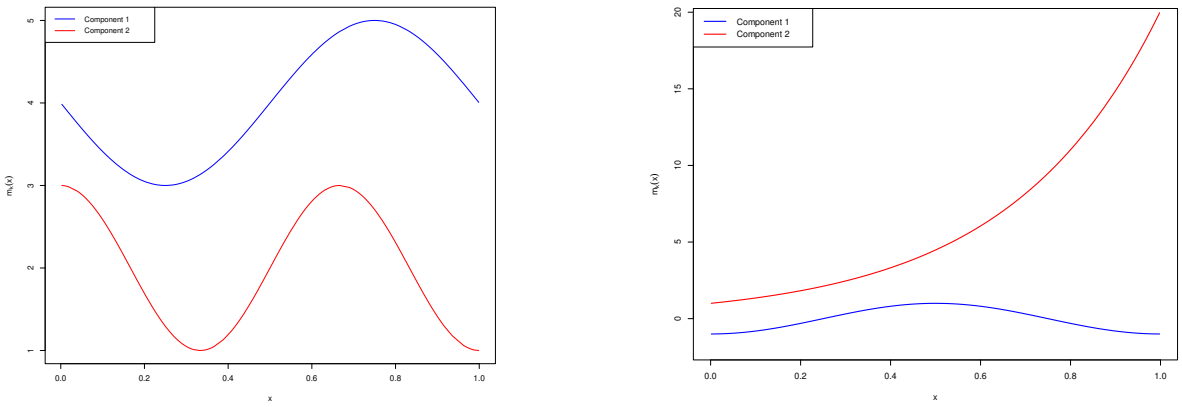
Kolmogorov-Smirnov (KS) statistic We use the Kolmogorov-Smirnov (KS) statistic to assess the goodness of the fit $\hat{F}_{\tilde{\boldsymbol{\theta}}}$ as

$$\text{KS} = \max_i |F_{\boldsymbol{\theta}}(y_i|x_i) - \hat{F}_{\tilde{\boldsymbol{\theta}}}(y_i|x_i)| \qquad \text{for } i = 1, 2, \ldots, n. \tag{5.49}$$

The smaller the value of KS, the better the goodness of the fitted distribution.

Initialisation strategy Following Xiang and Yao [2016], we make use of a mixture of regression splines (MRS) to initialise the proposed estimation procedure. To estimate the MRS, we make use of the `bs` and `ns` functions from the R package `splines`. The knots are chosen as the quartiles of x.

Simulation studies For each of our numerical experiments, we generate 500 data sets of sizes $n = 200, 400$ and 800. We make use of $N = 100$ local points chosen uniformly on the range of x. In all our simulations, the covariate x is generated from a uniform distribution on the interval $(0, 1)$. We make use of the Gaussian kernel function.



(a) (b)

Figure 5.1: CRFs for the model in (a) Table 5.1 and (b) Table 5.4

Example 1: Evaluating the performance of the proposed methods towards addressing label-switching The purpose of our first experiment is to demonstrate that the proposed methods are less sensitive to label-switching and produce reliable estimators. We consider data generated from a $K = 2$ component SPGMNRs given in Table 5.1. The CRFs,

Table 5.1: Data generating model

k	1	2
π_k	0.65	0.35
$m_k(x)$	$4 - \sin(2x\pi)$	$2 + \cos(3x\pi)$
σ_k^2	0.09	0.16

$m_k(x)'$s, in Table 5.1 are given in Figure 5.1a. We fit model (1.4) for $K = 2$ on the generated data using LPL estimators obtained via the naïve EM algorithm (naiveEM) (see section 1.2) and the proposed MB-ECM algorithm. Recall that in stage 1 of the naiveEM, among others, the mixing proportions are assumed to be non-parametric. However, the LLEs of the mixing proportion functions have no closed form expressions. Consequently, this demonstration is based on the results obtained using the LCEs.
Using the GCV approach, the bandwidths for the LCE were chosen as 0.04, 0.03, 0.02 for the sample sizes $n = 200$, 400 and 800, respectively. The results are given in Table 5.2.

Figure 5.2 shows four of the fitted CRFs chosen from among the 500 fitted models for sample size $n = 200$. These fitted CRFs were chosen from the first four, of the 500, fitted models with the largest likelihood based on the naiveEM. As can be seen from the figure, the estimates based on the naiveEM (right-column) are wiggly and non-smooth. The figure also shows that the

Table 5.2: Average (and standard deviations) of the performance measures over the 500 replications based on the LCEs. Bold values indicate the the best performing approach

n	Algorithm	RASE($f_{\boldsymbol{\theta}}$)	RASE($\mathbf{m}$)	KS	ASE(π)	ASE(σ_1^2)	ASE(σ_2^2)	ARI
200	naiveEM	0.136	0.383	0.023	**0.007**	0.009	0.002	0.886
		(0.033)	(0.188)	(0.011)	**(0.021)**	(0.068)	(0.002)	(0.08)
	MB-ECM	**0.125**	**0.213**	**0.018**	0.009	**0.002**	0.002	**0.908**
		(0.028)	**(0.052)**	**(0.008)**	(0.026)	**(0.013)**	(0.002)	**(0.052)**
400	naiveEM	0.102	0.276	0.019	**0.006**	0.003	**0.001**	0.904
		(0.023)	(0.104)	(0.009)	**(0.021)**	(0.037)	**(0.002)**	(0.053)
	MB-ECM	**0.090**	**0.155**	**0.012**	0.011	**0.001**	0.002	**0.918**
		(0.017)	**(0.03)**	**(0.005)**	(0.029)	**(0.002)**	(0.002)	**(0.03)**
800	naiveEM	0.082	0.223	0.014	**0.002**	0.000	**0.000**	0.913
		(0.012)	(0.054)	(0.006)	**(0.012)**	(0.000)	**(0.001)**	(0.02)
	MB-ECM	**0.070**	**0.122**	**0.008**	0.005	0.000	0.001	**0.923**
		(0.01)	**(0.02)**	**(0.004)**	(0.021)	(0.001)	(0.001)	**(0.02)**

estimates based on the proposed MB-ECM algorithm (left-column) are more stable compared to those obtained by the naiveEM. For further evidence of the outperformance of the proposed method compared to the naiveEM, Table 5.2 gives the average and standard deviations of the performance measures over all the 500 replicates. The results from Table 5.2 show that the naiveEM gives better estimates of the mixing proportions compared to the proposed method. However, the overall fit of the model based on the proposed methods is better than the fit based on the naiveEM. These results reaffirm and emphasise that the estimates based on the naiveEM are unstable and consequently unreliable. This remains the case as we increase the sample size. Thus, the naiveEM is not useful in practice. This serves as a motivation for the proposed method. As can been seen from Figure 5.2 and Table 5.2, the proposed method is able to address the challenges faced by the naiveEM because they produce smooth estimates of the non-parametric functions, the resulting estimators are stable and get better as we increase the sample size.

To demonstrate that the performance of the naiveEM is not due to undersmoothing (that is, a very small bandwidth h), we repeated the simulations using twice the optimal bandwidth. The results are given in Table 5.3. As can be seen from the table, the proposed method still outperforms the naive method.

Example 2: Local-constant estimator vs. Local-linear estimator Next, we compare the results obtained using LCEs and LLEs. The data for this experiment is generated from the model in Table 5.4. A plot of the CRFs is given in Figure (5.1b). It is known that the first and second derivatives of the regression function is a multiplicative and additive term,

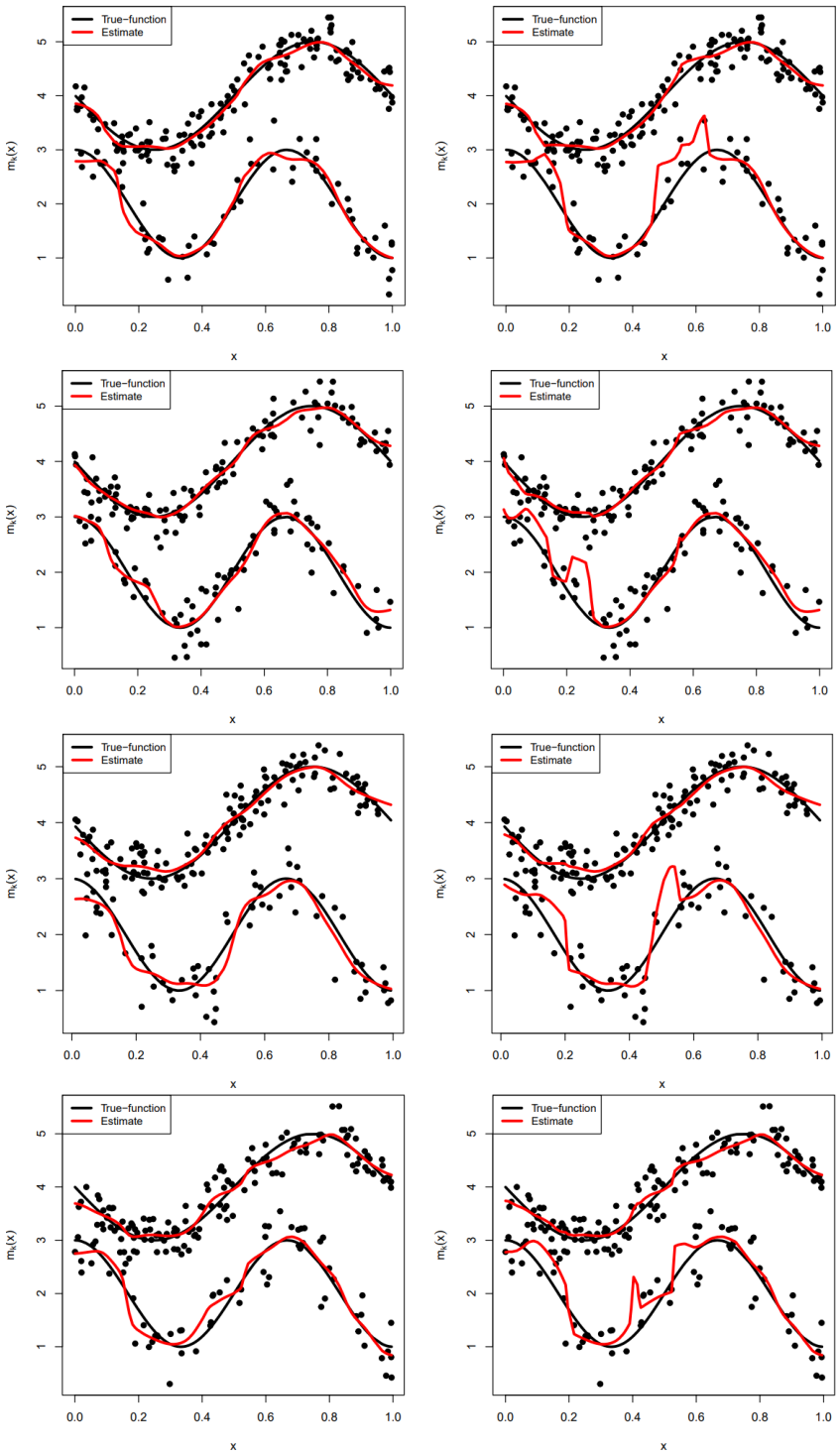


Figure 5.2: True (black curves) and fitted (red curves) CRFs from four estimates based on the LCEs obtained via the MB-ECM algorithm (**left-column**) and the naiveEM algorithm (**right-column**) for sample size $n = 200$. These CRFs were chosen from the first four fitted models ordered using the fitted likelihood values (from largest to smallest).

Table 5.3: Average (and standard deviations) of the performance measures over the 500 replications with an oversmoothing bandwidth h obtained as $2 \times h_{opt}$, where h_{opt} is the optimal bandwidth based on the GCV. Bold values indicate the the best performing approach

n		RASE($f_{\boldsymbol{\theta}}$)	RASE($\mathbf{m}$)	KS	ASE(π)	ASE(σ_1^2)	ASE(σ_2^2)	ARI
200	naiveEM	0.169	0.505	0.038	**0.006**	**0.008**	**0.002**	0.874
		(0.031)	(0.21)	(0.015)	**(0.019)**	**(0.071)**	**(0.002)**	(0.092)
	MB-ECM	**0.151**	**0.341**	**0.023**	0.008	0.012	0.011	**0.876**
		(0.033)	**(0.089)**	**(0.01)**	(0.022)	(0.092)	(0.063)	**(0.105)**
400	naiveEM	0.121	0.345	0.028	**0.006**	0.004	**0.001**	0.900
		(0.025)	(0.118)	(0.011)	**(0.021)**	(0.049)	**(0.002)**	(0.061)
	MB-ECM	**0.101**	**0.207**	**0.014**	0.007	**0.003**	0.002	**0.916**
		(0.019)	**(0.048)**	**(0.006)**	(0.02)	**(0.047)**	(0.004)	**(0.044)**
800	naiveEM	0.083	0.235	0.020	**0.003**	**0.000**	**0.000**	0.915
		(0.014)	(0.06)	(0.007)	**(0.015)**	**(0.001)**	**(0.001)**	(0.020)
	MB-ECM	**0.068**	**0.121**	**0.009**	0.009	0.001	0.001	**0.924**
		(0.01)	**(0.022)**	**(0.004)**	(0.025)	(0.002)	(0.002)	**(0.019)**

respectively, in the theoretical bias of a LCE of the regression function (see Fan [1992]). Thus, the CRF for component 2 was chosen so that its first and second derivatives are large. We are therefore interested in the estimation of the CRFs. Thus, we only report the RASE($\mathbf{m}$). Following Buja et al. [1989], we obtain the bandwidths such that the two estimators have the

Table 5.4: Data generating model

k	1	2
π_k	0.6	0.4
$m_k(x)$	$-\cos(2x\pi)$	$\exp(3x)$
σ_k^2	0.09	0.16

same total degrees of freedom (tdf), $\sum_{k=1}^{K} \text{df}_k$, where df_k is given by (5.44). This is done so that we can be able to compare the results based on the LCE and LLE (see Buja et al. [1989] for more details). Table 5.5 gives the average and standard deviations of the RASE, over all the 500 replicates, using the LCEs and LLEs obtained via the proposed MB-ECM. As can be seen from the table, LLEs perform better than the LCEs for estimating the CRFs. This is not unexpected. As alluded to above, if the true non-parametric function has a large first and second derivative, then the LCEs will be subject to bias (see Fan [1992]).

Example 3: Evaluating the sensitivity of the proposed MB-ECM algorithm on the value of the parameter λ_0 Next, we evaluate the sensitivity of the proposed MB-ECM algorithm on the value of the parameter λ_0. Before presenting any empirical results, intuitively, the value of λ_0 should not be too large because it might lead to the choice of an inadequately

Table 5.5: Average (and standard deviations) of the RASE($\mathbf{m}$) over the 500 replications based on the LCEs and LLEs obtained using the MB-ECM algorithm

Estimator	n		
	200	400	800
LCE	0.344 (0.081)	0.232 (0.045)	0.152 (0.025)
LLE	0.168 (0.041)	0.123 (0.026)	0.093 (0.018)

small (or zero!) number of local points. In the extreme case the algorithm will fail. On the other hand if λ_0 is chosen too small, the algorithm may not be able to separate significant local points from insignificant ones. In this case, it will not be effective in automatically choosing the location and number of local points. The resulting local neighbourhood will include all the initial local points.

For different SPGMNRs models, we evaluate the sensitivity of the fitted model on the value of λ_0. For a sample size $n = 400$, Table 5.6 gives the results of the MB-ECM algorithm for a range of values of λ_0. The value $1 \times 10^{-5} = 0.00001$.

Table 5.6: Evaluating the sensitivity of the MB-ECM algorithm: average (and standard deviations) of the performance measures over the 500 replications based on the local-constant estimators (LCEs) using $n = 400$

Model	λ_0	RASE($f_{\boldsymbol{\theta}}$)	RASE($\mathbf{m}$)	KS	ASE$_\pi$	ASE$_{\sigma_1^2}$	ASE$_{\sigma_2^2}$	ARI
Table 5.1	1×10^{-8}	0.0877	0.1567	0.0116	0.0096	0.0030	0.0014	0.9166
		(0.022)	(0.068)	(0.005)	(0.027)	(0.035)	(0.002)	(0.049)
	1×10^{-6}	0.0876	0.1570	0.0116	0.0095	0.0034	0.0014	0.9167
		(0.022)	(0.070)	(0.005)	(0.027)	(0.040)	(0.002)	(0.050)
	1×10^{-5}	0.0875	0.1577	0.0116	0.0101	0.0036	0.0014	0.9165
		(0.022)	(0.077)	(0.005)	(0.028)	(0.044)	(0.002)	(0.052)
	1×10^{-4}	0.087	0.1575	0.0116	0.0084	0.0039	0.0012	0.9165
		(0.022)	(0.074)	(0.005)	(0.026)	(0.047)	(0.002)	(0.053)
	1×10^{-2}	0.1435	0.2287	0.0144	0.0155	0.0681	0.1015	0.8474
		(0.060)	(0.140)	(0.007)	(0.029)	(0.248)	(0.181)	(0.142)
Table 5.4	1×10^{-8}	0.0772	0.1519	0.0122	0.0317	0.0069	0.0108	0.9692
		(0.015)	(0.030)	(0.005)	(0.019)	(0.005)	(0.006)	(0.018)
	1×10^{-6}	0.0770	0.1519	0.0122	0.0322	0.0072	0.0110	0.9691
		(0.015)	(0.030)	(0.005)	(0.019)	(0.005)	(0.006)	(0.018)
	1×10^{-5}	0.0768	0.1519	0.0122	0.0328	0.0075	0.0111	0.9692
		(0.015)	(0.030)	(0.005)	(0.018)	(0.006)	(0.005)	(0.018)
	1×10^{-4}	0.0765	0.1519	0.0122	0.0317	0.0074	0.0107	0.9693
		(0.014)	(0.030)	(0.005)	(0.019)	(0.006)	(0.006)	(0.017)
	1×10^{-2}	0.1332	0.2710	0.0145	0.0388	0.2999	0.0621	0.9215
		(0.069)	(0.180)	(0.007)	(0.025)	(1.388)	(0.271)	(0.129)

As can be seen from the table, for values of λ_0 at most 1×10^{-4}, the performance of the algorithm is virtually the same. However, when λ_0 is chosen greater than 1×10^{-4}, the performance deteriorates. In terms of choosing the number of local points where the estimation takes place, for $\lambda_0 = 1 \times 10^{-2}$, the algorithm tends to choose $2 - 10$ local points thus resulting in an inadequate fit. On the other hand, for $\lambda_0 = 1 \times 10^{-8}$, the algorithm tends to choose $95 - 100$ local points. This results are consistent with our above intuition. Thus, any value of λ_0 that is not too small (to prevent a large non-local neighbourhoods) and not too large (to prevent empty neighbourhoods) value will suffice. In our simulations and applications, we made use of $\lambda_0 = 1 \times 10^{-5}$.

Application

In this section, we demonstrate the practical usefulness of the proposed method on real data. For real data analysis,

1. we present results based on the proposed MB-ECM algorithm and compare them with the results based on the local EM-type (LEM) algorithm proposed by Xiang and Yao [2016]. As mentioned in section 1.2, the LEM algorithm addresses label-switching using a similar idea as in the effective EM algorithm (see subsection 3.2.3);

2. we initialise each fitting algorithm by making use of the fitted model based on the local-constant estimator;

3. we use the GCV criterion to to select the bandwidth for the local-constant estimator. We then choose a bandwidth for the local-linear estimator such that the total degrees of freedom (tdf) of the two estimators are the same. As before, this renders the fit based on the two estimators comparable;

4. we measure the goodness-of-fit using the RASE and Bayesian information criterion (BIC)

$$\text{BIC} = -2\ell + \text{df} \times \log(n), \tag{5.50}$$

where $\text{df} = \text{tdf} + 2K - 1$ is the overall model degrees of freedom and ℓ is the maximum log-likelihood value. Moreover, we assess the predictive ability of the fitted model using the mean squared prediction error (MPSE). Following Xiang and Yao [2016], we calculate the MSPE via a Monte Carlo cross validation (MCCV) procedure. The MCCV procedure randomly partitions the data into a training set with size $n(1-r)$ and a test set with size nr, where r is the proportion of data in the test set. The model is estimated using the data in the training set and then validated using data in the test set. The procedure is

repeated T times and we take the average of the MSPEs. We use $r = 0.1$ and $T = 200$; and

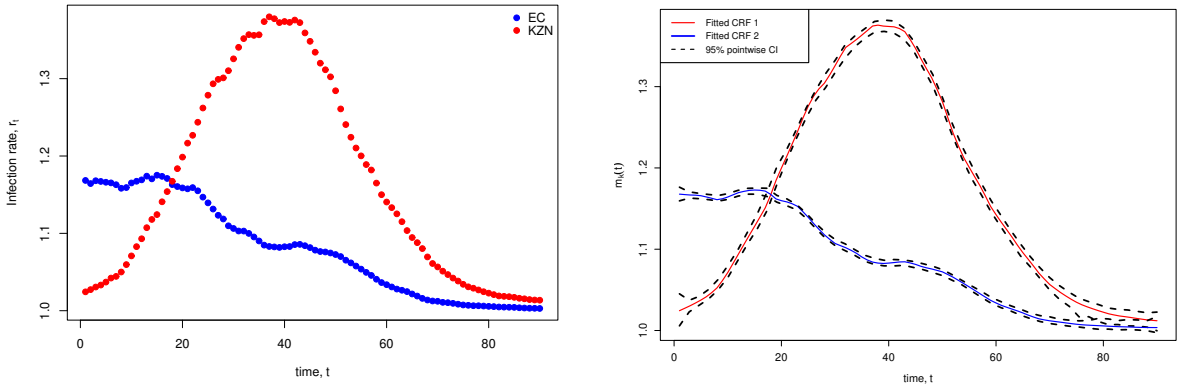
5. lastly, we use a conditional bootstrap approach to calculate the pointwise 95% confidence intervals of the fitted CRFs and the 95% confidence intervals of the component mixing proportions and variances. That is, for a given value of x, we sample the corresponding value of the response, denoted by y^*, from the fitted SPGMNRs model $\sum_{k=1}^{K} \hat{\pi}_k \mathcal{N}(y|\hat{m}_k(x), \hat{\sigma}_k^2)$. We repeat this sampling process n times to get a bootstrap sample $\mathcal{S} = \{(x_i, y_i^*) : i = 1, 2, \ldots, n\}$. We generate B bootstrap samples $\mathcal{S}^{(1)}, \mathcal{S}^{(2)}, \ldots, \mathcal{S}^{(B)}$ in the above manner. We fit the SPGMNRs model (1.4) on each of these bootstrap samples, thus generating a sampling distribution of $\hat{\pi}_k$, $\hat{\sigma}_k^2$ and $\hat{m}_k(x)$. To compute the 95% confidence intervals, we take the 2.5^{th} and 97.5^{th} percentiles of the sampling distributions as the lower and upper limits, respectively, of the interval. We set $B = 200$.

South African Covid-19 data For our first application, we consider the Covid-19 14-day infection rates (r_t) over time (t) in two South African provinces, Kwa-Zulu Natal (KZN) and the Eastern Cape (EC), for the period December 2020 to 15 February 2021. The infection rate (r_t) is calculated as $r_t = \frac{x_t}{x_{t-14}}$, where x_t is the daily cumulative COVID-19 positive cases at time point t. This measure is useful in modelling the spread of the disease over time. This data set was previously used by Millard and Kanfer [2022]. The data was collected from the Data Science for Social Impact COVID-19 data repository.

Figure 5.3a gives a scatter plot of the data along with the identity of the province that generated each data point. The purpose of this application is to demonstrate the effectiveness of the proposed method in addressing label-switching and identify each data point with the province that generated it. Thus, we take province as a latent variable. It is clear from Figure 5.3a that the relationship between the infection rate, r_t, and time, t, is non-linear in each province. Thus, we fit a $K = 2$ component SPGMNRs to the data.

The GCV criterion gave a bandwidth of 1.0249 for the local-constant estimator which corresponds with a tdf of about 71. The bandwidth for the local-linear estimator with about the same tdf is 1.0468. Table 5.7 gives the results of the fitted model obtained using the MB-ECM algorithm and LEM algorithm. Since we know the actual component (province) where each data point belongs to, we also measure the clustering ability of the fitted models using the ARI. For this data, the local-constant estimated model is slightly better than the local-linear estimate, with a small BIC and RASE. However, the predictive ability of the two estimates is virtually the same. The results based on the proposed MB-ECM and the LEM algorithm are virtually the same for this data set.

Figure 5.3b shows the fitted component regression functions (CRFs) using the proposed MB-



(a) (b)

Figure 5.3: SA Covid-19 data: (a) Scatter plot of the data. (b) Fitted CRFs for the SA Covid data using the LLE estimator via the MB-ECM algorithm. Also included are the 95% pointwise bootstrap confidence intervals.

ECM algorithm. We can see that the proposed method was able to detect the "latent" structure.

Table 5.7: SA Covid-19 data: The fitted model using the local-constant estimator (LCE) and local-linear estimator (LLE) via the MB-ECM algorithm and the LEM algorithm

	MB-ECM		LEM	
	LCE	LLE	LCE	LLE
RASE(×10)	0.0237	0.0241	0.0238	0.00241
BIC	-1204.1	-1198	-1212.4	-1207.7
ARI	1	1	1	1
MSPE	0.0002	0.0001	0.0002	0.0001
	(0.0002)	(0.0001)	(0.0002)	(0.0001)

African CO2 data For our next analysis on real data, we consider the relationship between carbon dioxide (CO_2) emissions, a measure of environmental degradation, and gross domestic product (GDP), a measure of the monetary value produced by a country in a given period. Figure 5.4a shows a scatter plot of CO_2 per capita (in metric tons) on GDP per capita (in US$) for a group of 51 African countries in 2014. The countries includes, among others, South Africa (ZAF), Botswana (BWA) and Zimbabwe (ZWE). The data were obtained from the World Bank's World development indicators database (accessed on 10 April 2023). A quick visual inspection of Figure 5.4a reveals two clusters (groups) of countries based on the relationship between CO_2 and GDP. Moreover, this relationship is not linear in either of the two groups. A

mixture of non-parametric regression analysis is apt for this data. Such an analysis can assist us in answering questions such as

- What development path is adopted by each group of countries? Especially, the low GDP countries.
- Which countries, if any, are pursuing economic growth at a high cost to the environment?
- Is a linear relationship between CO2 and GDP appropriate for each group of countries?
- Are there more than two groups of countries?

After standardising the variables, we fit a $K = 2$ component SPGMNRs model to the data on Figure 5.4a in an attempt to answer some of the questions above. The GCV criterion chose a bandwidth of 0.1725 for the LCE which corresponds to a tdf of about 14. To obtain about the same tdf, the bandwidth of the LLE was chosen to be 0.2343. To confirm that there are indeed two groups and the regression relationships are non-linear, we also fitted the SPGMNRs and GMLRs models with $K = 1, 3, 4$ and 5 components and compared them based on the BIC. The SPGMNRs and the GMLRs for $K = 1$ are essentially the non-parametric regression and linear regression models, respectively. These models were fitted using the R functions: `locfit` (Loader [2023]) and `glm`, respectively.
The results (Table 5.8) show that the $K = 2$ component SPGMNRs model is appropriate for this data having the smallest value of the BIC. Thus, we have confirmed that there are indeed two groups of countries. We therefore proceed with the fitted $K = 2$ component SPGMNRs model. Table 5.9 gives the results from the fitted model. It can be seen that the model

Table 5.8: BIC values obtained for the SPGMNRs fitted using the MB-ECM algorithm and the GMLRs model fitted using the EM algorithm. The SPGMNRs and GMLRs with $K = 1$ corresponds with the non-parametric regression model and simple linear regression model, respectively.

K	Model	
	SPGMNRs	GMLRs
1	70.888	100.420
2	**-15.046**	9.793
3	10.598	25.520
4	16.355	32.574
5	26.168	56.975

based on the local-linear estimator is the best as it attains the best overall model goodness-of-fit and good performance on out-of-sample prediction. Moreover, the overall performance of the proposed MB-ECM algorithm is slightly better than that the other LEM for this data set. Based on the proposed local-linear one-step backfitting estimators via the MB-ECM algorithm,

Table 5.9: CO2 data: The fitted model using the LCE and LLE via the MB-ECM algorithm and LEM algorithm

	MB-ECM		LEM	
	LCE	LLE	LCE	LLE
RASE	0.1264	0.1233	0.1213	0.1072
BIC	-34.4426	-39.2662	-16.9129	-28.6801
MSPE	0.1143	0.0954	0.1390	0.1003
	(0.1414)	(0.1256)	(0.16274)	(0.1356)

the mixing proportions and and variances, along with their 95% bootstrap confidence intervals, were obtained as 0.4775 (0.2425 - 0.5054), 0.5225 (0.4946 - 0.7576), 0.0106 (0.0047 - 0.0343) and 0.0053 (0.0010 - 0.0148), respectively. Figure 5.4c and 5.4d gives the fitted CRFs obtained using the proposed LLEs via the MB-ECM algorithm. Included in Figure 5.4 are the 95% pointwise bootstrap confidence intervals. The estimated CRFs based on the LEM are similar and hence they are excluded.

The estimated CRF in Figure 5.4c reveals an interesting phenomenon. CO_2 emissions increase up until a certain level of GDP. Thereafter, beyond this level, they exhibit a slow down in further increases of CO_2 emissions. This is consistent with the well-known environmental Kuznets curve (EKC) hypothesis in environmental economics (see Dinda [2004]). The EKC says that, at the development phase, the value of a country's economy increases at a high cost to the environment due to high carbon emissions from the industrialisation process. Beyond a certain level of growth, this effect is reversed and economic growth leads to lower carbon emissions. This phenomenon hypothesises a non-linear negative parabolic-like relationship between CO2 and GDP. Assuming that all countries follow the same EKC, for a cross-section of countries, the estimated EKC's in Figure 5.4 show countries at different stages of development (Dinda [2004]). Using model-based clustering (see McNicholas [2016]), we can use the fitted model to assign each country to a given group. The results are given in Figure 5.4b. We find that the developmental path given by the curve in Figure 5.4c is made up by countries such as Namibia, Swaziland and Botswana. Countries in which the energy mix is becoming less dominated by fossil fuels. Whereas the developmental path given by the curve in Figure 5.4d is made up by countries such as South Africa, Morocco and Egypt. Countries in which the energy mix is still heavily dominated by fossil fuels.

Discussion and Conclusion

This essay was concerned with estimating semi-parametric Gaussian mixtures of non-parametric regressions (SPGMNRs) (1.4) using the proposed model-based one-step backfitting procedure in order to demonstrate the effectiveness and practical utility of the estimation procedure in

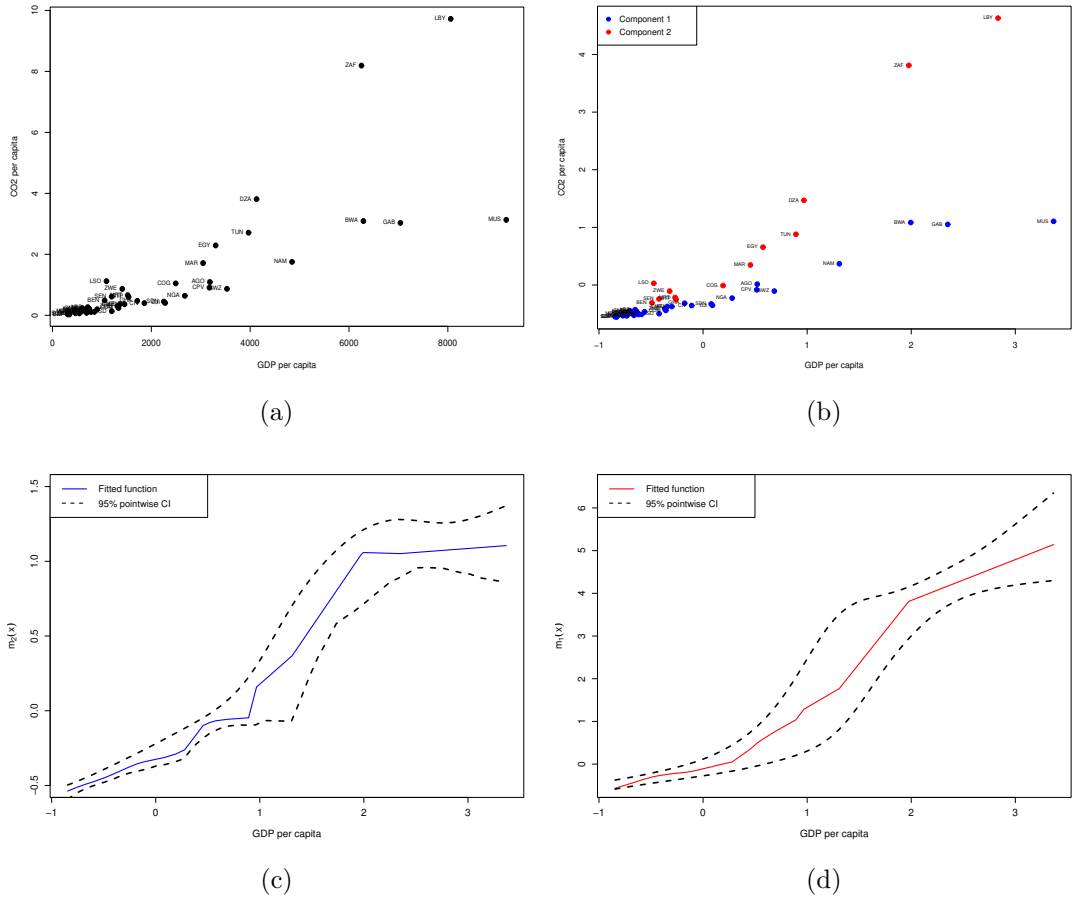


Figure 5.4: (a) Scatter plot of the CO2 data. Fitted $K = 2$ component SPGMNRs model obtained using the LLE via the MB-ECM algorithm. (b) hard clustered data based on the fitted model. (c) and (d) Fitted CRF for component 1 and component 2, respectively.

addressing label-switching and producing useful model estimates, respectively.

We used intensive Monte Carlo simulations to, first, show that the naive estimation procedure (naiveEM) (see section 1.2) may lead to wiggly, non-smooth and ultimately practically unreliable estimates of the non-parametric CRFs. Thus, the naiveEM, is unstable and sensitive to label-switching. On the other hand, we showed that the proposed model-based procedure (MB-ECM) overcomes the challenges of the naiveEM. The MB-ECM is stable, less sensitive to label-switching and tended to result in model estimates that were consistent with theory. Moreover, the MB-ECM, was shown to always outperform the naiveEM in model fit accuracy and clustering capability as measured by the KS statistic and ARI, respectively.

Next, we showed that the local-linear estimator (LLE) is superior to the local-constant estimator (LCE) as an estimator of the non-parametric CRFs. Lastly, under different scenarios,

we demonstrated that the MB-ECM is not sensitive to the choice of the tuning parameter (threshold) λ_0, if this parameter is chosen to be neither too small nor too large. This is similar to the bias-variance trade-off when choosing the optimal bandwidth. We recommend using $\lambda_0 = 1 \times 10^{-5}$. This choice of value for λ_0 was found to work well in all scenarios. We also made use of it in our real data analysis.
In our real data analysis, we demonstrated the practical utility of the MB-ECM on two datasets: (1) South African Covid-19 data and (2) CO_2-GDP data on a cross-section of African countries. We fitted the SPGMNRs model on both datasets using the MB-ECM and obtained reasonable estimates. Most notably, the model fitted on the second dataset revealed that African countries formed the well-known environmental Kuznets curve (EKC). The EKC postulates that, the size of an economy increases at a high cost to the environment due to high greenhouse gas emissions from the industrialisation process. This continues up to a certain level of economic expansion, thereafter further economic expansion is followed by a decline in emissions. The fitted model shows countries at different points of the curve, corresponding to different phases of development. Some countries are at the initial stages, some are at the intermediate stage while some are close to the turning point.

5.3.2 Semi-parametric Gaussian mixtures of regressions with varying mixing proportions (SPGMRVPs)

The GMLRs (2.19) assumes that the probability that any given data point, say $(\mathbf{x}_i, y_i)$, can be explained by, for instance, the k^{th} regression component is the same for all data points, that is $\pi_k = \pi_k(t_i)$, for $i = 1, 2, \ldots, n$, where t is a covariate. This is quite a restrictive assumption. As mentioned in section 1.1, some or all of the covariates might contain information about the mixing proportions $\boldsymbol{\pi}$. Thus, an appropriate model in this case is a model of the form (1.17), where the effect of covariate(s) on the mixing proportions is explicitly specified.
In this essay, we demonstrate the proposed model-based estimation strategy for estimating model (1.17). As in the previous essays, we use simulated data and real datasets to show the performance and practical use of the proposed approach.

Estimation procedure

Consider a random sample $\{(t_i, \mathbf{x}_i, y_i) : i = 1, 2, \ldots, n\}$ from model (1.17). Assume that the parameters β_k and σ_k^2, for $k = 1, 2, \ldots, K$, are non-parametric functions of t and let $\mathcal{U} = \{u_1, u_2, \ldots, u_N\}$ be a set of N local points in the domain of the covariate t. Then, locally,

model (1.17) is a GMLRs (2.19)

$$f_u(y|\mathbf{X}=\mathbf{x}) = \sum_{k=1}^{K} \pi_k(u)\mathcal{N}\Big(y|\mathbf{x}^\intercal\boldsymbol{\beta}_k(u), \sigma_k^2(u)\Big). \tag{5.51}$$

One of these local GMLRs can be viewed as the conditional distribution of y given $\mathbf{X}=\mathbf{x}$ that generated the n data pairs $\{(\mathbf{x}_i, y_i) : i = 1, 2, \ldots, n\}$. Since we do not observe the identity of this local GMLRs model, conditional on $\mathbf{X}=\mathbf{x}$, y follows a mixture of these local GMLRs models

$$\begin{aligned} f(y|\mathbf{X}=\mathbf{x}) &= \sum_{j=1}^{N} \lambda_j f_{u_j}(y|\mathbf{X}=\mathbf{x}) \\ &= \sum_{j=1}^{N} \lambda_j \Bigg[\sum_{k=1}^{K} \pi_k(u_j)\mathcal{N}\Big(y|\mathbf{x}^\intercal\boldsymbol{\beta}_k(u_j), \sigma_k^2(u_j)\Big)\Bigg] \\ &= \sum_{j=1}^{N}\sum_{k=1}^{K} \lambda_j\pi_k(u_j)\mathcal{N}\Big(y|\mathbf{x}^\intercal\boldsymbol{\beta}_k(u_j), \sigma_k^2(u_j)\Big), \end{aligned} \tag{5.52}$$

where $\lambda_j > 0$ (satisfying $\sum_{j=1}^{N}\lambda_j = 1$) is the mixing proportion, probability or weight.
Model (5.52) represents a reformulation of model (1.17). The model can be estimated using maximum likelihood via a modified ECM-type algorithm as outlined in section 5.1. By estimating model (5.52), we are in effect simultaneously estimating the local parameters, thus avoiding the label-switching problem.
To simplify the notation, model (5.52) can be written as

$$f(y|\mathbf{X}=\mathbf{x}, T=t) = \sum_{j=1}^{N} \lambda_j \sum_{k=1}^{K} \pi_{j,k}\mathcal{N}\Big(y|\mathbf{x}^\intercal\boldsymbol{\beta}_{j,k}, \sigma_{j,k}^2\Big), \tag{5.53}$$

where $\pi_{j,k} = \pi_k(u_j)$, $\boldsymbol{\beta}_{j,k} = \boldsymbol{\beta}_k(u_j)$ and $\sigma_{j,k}^2 = \sigma_k^2(u_j)$ and t is explicitly included in the reformulated model. Recall that the local points $\mathcal{U}$ depend on the domain of t.
The log-likelihood function corresponding to (5.53) is

$$\ell_0(\boldsymbol{\lambda}, \boldsymbol{\theta}) = \sum_{i=1}^{n} \log\Bigg[\sum_{j=1}^{N}\sum_{k=1}^{K} \lambda_j\pi_{j,k}\mathcal{N}\Big(y|\mathbf{x}^\intercal\boldsymbol{\beta}_{j,k}, \sigma_{j,k}^2\Big)\Bigg], \tag{5.54}$$

where $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_N)$ and $\boldsymbol{\theta} = (\boldsymbol{\theta}(u_1), \boldsymbol{\theta}(u_2), \ldots, \boldsymbol{\theta}(u_N))$ with $\boldsymbol{\theta}(u_j) = (\boldsymbol{\pi}_{j\cdot}, \boldsymbol{\beta}_{j\cdot}, \boldsymbol{\sigma}_{j\cdot}^2)$, $\boldsymbol{\pi}_{j\cdot} = (\pi_{j,1}, \pi_{j,2}, \ldots, \pi_{j,K})$, $\boldsymbol{\beta}_{j\cdot} = (\boldsymbol{\beta}_{j,1}, \boldsymbol{\beta}_{j,2}, \ldots, \boldsymbol{\beta}_{j,K})$ and $\boldsymbol{\sigma}_{j\cdot}^2 = (\sigma_{j,1}^2, \sigma_{j,2}^2, \ldots, \sigma_{j,K}^2)$, for $j = 1, 2, \ldots, N$.

To maximise (5.54), we use an ECM-type algorithm as before. The ECM update equations for the parameters $\boldsymbol{\lambda}$ and $\boldsymbol{\theta}$ are derived similar to those obtained from maximising the log-likelihood function (5.8).
Let $\boldsymbol{\theta}^{(R)}$ and $\mathcal{T}^{(R)}$ be the resulting parameter estimates and the set of indices of the local points, respectively, obtained at convergence of the above ECM algorithm. To obtain $(\hat{\pi}_k(t_i), \hat{\boldsymbol{\beta}}_k(t_i), \hat{\sigma}^2(t_i))_{1\leq i\leq n, 1\leq k\leq K}$, respectively, we linearly interpolate over
$(\lambda_{j,k}^{(R)}, \boldsymbol{\beta}_{j,k}^{(R)}, \sigma_{j,k}^{2(R)})_{1\leq k\leq K, j\in\mathcal{T}^{(R)}}$.
Note that $\boldsymbol{\beta}$ and $\boldsymbol{\sigma}^2$ are global parameters but their first-stage estimates $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\sigma}}^2$ are non-parametric (local). Given $\hat{\boldsymbol{\pi}}$, we propose updated estimates $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$, respectively, obtained by maximising the global log-likelihood function

$$\ell_1(\boldsymbol{\beta}, \boldsymbol{\sigma}^2) = \sum_{i=1}^{n} \log\left[\sum_{k=1}^{n} \hat{\pi}_k(t_i)\mathcal{N}(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_k, \sigma_k^2)\right] \tag{5.55}$$

We can make use of these global parameter estimates ($\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$) to improve the first-stage estimates of the mixing proportion functions $\hat{\boldsymbol{\pi}}$. Thus, given $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$, we propose the estimate $\tilde{\boldsymbol{\pi}}$ obtained by maximising the local log-likelihood function

$$\ell_2[\boldsymbol{\pi}(u)] = \sum_{i=1}^{n} \log\left\{\sum_{k=1}^{K} \pi_k(u)\mathcal{N}(y_i|\mathbf{x}_i^\intercal\tilde{\boldsymbol{\beta}}_k, \tilde{\sigma}_k^2\}\right) K_h(t_i - u). \tag{5.56}$$

Note that the global parameter estimates $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$ are well labelled. This implies that each local log-likelihood function (5.56) can be maximised separately without being concerned about label switching. Let $\tilde{\boldsymbol{\theta}} = (\tilde{\boldsymbol{\pi}}, \tilde{\boldsymbol{\beta}}, \tilde{\boldsymbol{\sigma}}^2)$ be the one-step backfitting estimate of $\boldsymbol{\theta}$.
In summary, the proposed estimation procedure proceeds in two stages. In the first stage, we obtain $\hat{\boldsymbol{\pi}}$, $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\sigma}}^2$. Thereafter, in the second stage, we obtain $\tilde{\boldsymbol{\pi}}$, $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$.
The following is a one-step backfitting algorithm that can be used to carry out the above two-stage estimation procedure.

Stage 0: Initialising the algorithm Obtain appropriate initial estimates of the global parameters and the non-parametric functions, denoted $(\boldsymbol{\beta}^{(0)}, \boldsymbol{\sigma}^{2(0)})$ and $\boldsymbol{\pi}^{(0)}$, respectively, by making use of, say a mixture of experts model. Let $\mathcal{U}$ be the set of N grid points, $\mathcal{T}^{(0)} = \{1, 2, \ldots, N\}$ be the initial set of indices and specify λ_0.

Stage 1: Model-based ECM-type algorithm to maximise (5.54) Let $\boldsymbol{\lambda}^{(r)}$, $\boldsymbol{\theta}_1^{(r)}$ and $\boldsymbol{\theta}_2^{(r)}$ be the parameter estimates obtained at the r^{th} iteration.
E-Step: At the $(r+1)^{th}$ iteration, calculate $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$ and $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ by first estimating the

conditional expectations of $\mathbf{v}_i$ and $\mathbf{z}_i$, for $i = 1, 2, \ldots, n$, using (5.10) and (5.11), respectively.

CM-Step 1: Maximise $Q(\boldsymbol{\lambda}|\boldsymbol{\lambda}^{(r)})$ to obtain $\boldsymbol{\lambda}^{(r+1)}$ and $\mathcal{T}^{(r+1)}$ using (5.12) and (5.13), respectively.

CM-Step 2: Given $\mathcal{T}^{(r+1)}$, maximise $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(r)})$ to obtain $\boldsymbol{\theta}^{(r+1)} = (\pi_{j,k}^{(r+1)}, \boldsymbol{\beta}_{j,k}^{(r+1)}, \sigma_{j,k}^{2(r+1)})_{j\in\mathcal{T}^{(r+1)},1\leq k\leq K}$ using (5.15), (5.16) and (5.17).
Repeat the above E- and CM-steps until convergence.

Stage 2(a): EM algorithm to maximise ℓ_1 in in (5.55)

Given $\hat{\boldsymbol{\pi}}$ obtained from Stage 1, we obtain the global estimates $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$ of the global parameters $\boldsymbol{\beta}$ and $\boldsymbol{\sigma}^2$, respectively, by maximising ℓ_1 in (5.55) using the usual EM algorithm.

E-Step: At the $(r+1)^{th}$ iteration, calculate the expected value of the latent variable as

$$p_{ik}^{(r+1)} = \frac{\hat{\pi}_k(t_i)\mathcal{N}(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_k^{(r)}, \sigma_k^{2(r)})}{\sum_{\ell=1}^K \hat{\pi}_\ell(t_i)\mathcal{N}(y_i|\mathbf{x}_i^\intercal\boldsymbol{\beta}_\ell^{(r)}, \sigma_\ell^{2(r)})}. \tag{5.57}$$

M-Step: We obtain $\boldsymbol{\beta}_k^{(r+1)}$ and $\sigma_k^{2(r+1)}$, respectively, using the following equations

$$\boldsymbol{\beta}_k^{(r+1)} = \left(\sum_{i=1}^n p_{ik}^{(r+1)}\mathbf{x}_i\mathbf{x}_i^\intercal\right)^{-1}\left(\sum_{i=1}^n p_{ik}^{(r+1)}\mathbf{x}_i y_i\right), \tag{5.58}$$

$$\sigma_k^{2(r+1)} = \frac{\sum_{i=1} p_{ik}^{(r+1)}\left(y_i - \mathbf{x}_i^\intercal\boldsymbol{\beta}_k^{(r+1)}\right)^2}{\sum_{i=1}^K p_{ik}^{(r+1)}}. \tag{5.59}$$

Repeat the above E- and M-step until convergence

Stage 2(b): EM algorithm to maximise ℓ_2 in in (5.56)

Given $\tilde{\boldsymbol{\beta}}$ and $\tilde{\boldsymbol{\sigma}}^2$ obtained from Stage 2(a), we propose an improved estimate of the component mixing proportion functions, denoted by $\tilde{\boldsymbol{\pi}}$, obtained by maximising each local log-likelihood function in (5.56) using the usual EM algorithm.

E-Step: At the $(r+1)^{th}$ iteration, calculate the expected value of the latent variable as

$$p_{ik}^{(r+1)}(u) = \frac{\pi_k^{(r)}(u)\mathcal{N}(y_i|\mathbf{x}_i^\intercal\tilde{\boldsymbol{\beta}}_k, \tilde{\sigma}_k^2)}{\sum_{\ell=1}^K \pi_\ell^{(r)}(u)\mathcal{N}(y_i|\mathbf{x}_i^\intercal\tilde{\boldsymbol{\beta}}_\ell, \tilde{\sigma}_\ell^2)}. \tag{5.60}$$

M-Step: We obtain $\pi_k^{(r+1)}(u)$, for $u \in \mathcal{U}$, using the LPL estimator $\hat{\pi}_{k0}(u)$ obtained as the maximiser of

$$\sum_{i=1}^{n} p_{ik}^{(r+1)}(u) K_h(t_i - u) \log\Big\{ \sum_{j=0}^{p} \pi_{kj}(u)[t_i - u]^j \Big\} \tag{5.61}$$

Repeat the above E- and M-step until convergence.
At convergence of the EM algorithm of Stage 2(b), we obtain $\tilde{\boldsymbol{\pi}} = (\tilde{\boldsymbol{\pi}}_1, \tilde{\boldsymbol{\pi}}_2, \ldots, \tilde{\boldsymbol{\pi}}_K)$, where $\tilde{\boldsymbol{\pi}}_k = (\tilde{\pi}_k(t_1), \tilde{\pi}_k(t_2), \ldots, \tilde{\pi}_k(t_n))$, by linear interpolation over $\pi_k^{(R)}(u_j)$ for $j = 1, 2, \ldots, N$ and $k = 1, 2, \ldots, K$.

Simulations

In this section, we perform numerical experiments to demonstrate the performance of the proposed methods. The purpose of these experiments is two fold. First, we want to demonstrate the effectiveness of the proposed method towards addressing label-switching. Second, we want to demonstrate the practical utility of the of the proposed approach. As before, we refer to the proposed one-step backfitting algorithm as the MB-ECM algorithm.

Choosing the bandwidth h In order to implement the proposed methods, we must choose a bandwidth h. This parameter must be chosen subjectively or objectively based on the data. We will choose this parameter objectively. The most popular data-driven approach to bandwidth selection is the cross-validation (CV) approach (see chapter 7 of Hastie et al. [2009] for more details).
Let $\mathcal{D}$ denote the observed data set. We randomly divide $\mathcal{D}$, L times, into a training data set $\mathcal{R}_l$ and test data set $\mathcal{S}_l$, for $l = 1, 2, \ldots, L$. We estimate model (1.17) using the training data set $\mathcal{R}_l$ to obtain the estimates $(\hat{\boldsymbol{\pi}}(\cdot), \hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\sigma}}^2)$. We use these estimates to obtain $\hat{\pi}_k(t_j)$ and

$$\gamma_{jk} = \frac{\hat{\pi}_k(t_j)\mathcal{N}(y_j | \mathbf{x}_j^\intercal \hat{\boldsymbol{\beta}}_k, \hat{\sigma}_k^2)}{\sum_{\ell=1}^{K} \hat{\pi}_\ell(t_j)\mathcal{N}(y_j | \mathbf{x}_j^\intercal \hat{\boldsymbol{\beta}}_\ell, \hat{\sigma}_\ell^2)}, \tag{5.62}$$

for $(\mathbf{x}_j, t_j, y_j) \in \mathcal{S}_l$ and $k = 1, 2, \ldots, K$. Finally, we define the CV error as

$$\mathrm{CV}(h) = \sum_{l=1}^{L} \sum_{j \in \mathcal{S}_l} \sum_{k=1}^{K} \gamma_{jk}(y_j - \hat{y}_j)^2, \tag{5.63}$$

where $\hat{y}_j = \mathbf{x}_j^\intercal \hat{\boldsymbol{\beta}}_k$ is the fitted value of the test set response y_j.
We choose the bandwidth h that minimises the CV error (5.63). In our simulations we set

$L = 10$.

Performance measures As before, to evaluate the goodness of the non-parametric functions ($\tilde{\boldsymbol{\pi}}$), the overall model ($\hat{f}_{\tilde{\boldsymbol{\theta}}}$) and the fitted component regression parameters ($\tilde{\boldsymbol{\beta}}$), we make use of the root average squared error (RASE)

$$\text{RASE}^2(\pi_k) = \frac{1}{n}\sum_{i=1}^{n}\left[\tilde{\pi}_k(t_i) - \pi_k(t_i)\right]^2, \tag{5.64}$$

$$\text{RASE}^2(f_{\boldsymbol{\theta}}) = \frac{1}{n}\sum_{i=1}^{n}\left[\hat{f}_{\tilde{\boldsymbol{\theta}}}(y_i|x_i) - f_{\boldsymbol{\theta}}(y_i|x_i)\right]^2 \tag{5.65}$$

and

$$\text{ASE}(\boldsymbol{\beta}_k) = \frac{1}{D}\sum_{j=0}^{D}\left[\tilde{\beta}_{jk} - \beta_{jk}\right]^2, \tag{5.66}$$

where D is the size of the covariate vector $\mathbf{x}$ and β_{jk} is the coefficient of the covariate x_j, for $j = 0, 1, \ldots, D$ and $x_0 = 1$ for the intercept parameter.
To assess the clustering capability and goodness of the estimated CDF $F_{\boldsymbol{\theta}}$ for the fitted model, we use the ARI and the KS statistic, respectively. See subsection 5.3.1 for their definition and interpretation.

Initialisation strategy We initialise the proposed estimation procedure using the mixture of experts (MEs) model. To estimate the MEs model, we make use of the *hmeEM* function in the R package `mixtools` (Benaglia et al. [2009]).

Simulation study For each of our numerical experiments, we generate 500 data sets of sizes $n = 200, 400$ and 800. We make use of $N = 100$ local points chosen uniformly on the range of t. For illustrative purposes, in all our simulations $t = x$, where $\mathbf{x} = \begin{pmatrix}1 & x\end{pmatrix}^\intercal$. Moreover, the covariate is generated from the uniform distribution on the interval $(0, 1)$. We make use of the Epanechnikov kernel function. We use the proposed method (MB-ECM) to obtain the model results and compare them with the results obtained using the naiveEM algorithm.

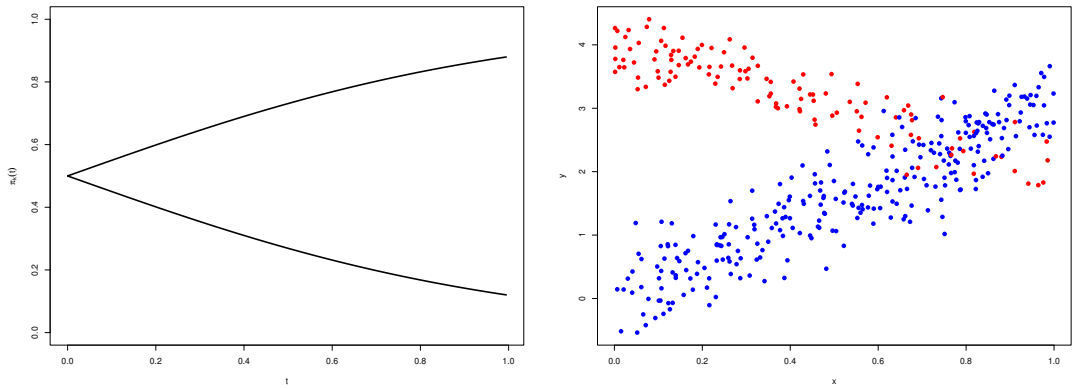
Example 1 For our first example, we generate data from a $K = 2$ component SPGMRVPs model (1.17) whose parameters are given in Table 5.10. The true mixing proportion functions (left plot) and a scatter plot (right plot) of a typical sample of size 400 are shown in Figure 5.5. It is clear from the figure that the mixing proportion functions are not constant but monotone functions of t. Figure 5.6 shows the fitted mixing proportion functions using the MB-ECM

Table 5.10: Data generating model for Example 1

k	1	2
$\pi_k(t)$	$1/(1+\exp(-2t))$	$1-\pi_1(t)$
$\boldsymbol{\beta}_k$	$\begin{pmatrix}4 & -2\end{pmatrix}^\intercal$	$\begin{pmatrix}0 & 3\end{pmatrix}^\intercal$
σ_k^2	0.09	0.16

(left plot) and the naiveEM (right plot) for three randomly selected samples of size 200, 400 and 800, respectively. It can be seen from the figure that the fitted functions based on the naiveEM are sensitive to label-switching, characterised by wiggly and non-smooth estimated functions. Moreover, this continues to be the case as we increase the sample size. On the other hand, the estimated functions based on the proposed method exhibit stability and appear to be less sensitive to label-switching.

For further evidence in support of the stability of the proposed method, Table 5.11 gives the average and standard deviation of the performance measures for the proposed method and the naiveEM calculated over the 500 simulated samples of sizes $n = 200$, 400 and 800. The tabulated results are an affirmation of the statement made based on the results shown in Figure 5.6. Based on this results, we can say that the naiveEM is unstable, unreliable and prone to result in estimates that exhibit label-switching. On the other hand, the proposed method is stable, gives estimates that are reliable and is less sensitive to label-switching.



(a) (b)

Figure 5.5: (a) The $K = 2$ mixing proportion functions: a monotone decreasing function $\pi_1(t)$ and increasing function $\pi_2(t)$. (b) A scatter plot of a typical sample of size $n = 400$. The red data points are from component 1 and the blue data points are from component 2.

Table 5.11: Average (and standard deviation) of the performance measures for the MB-ECM and naiveEM over the 500 simulated samples of sizes $n = 200$, 400 and 800 generated from the model in Example 1.

	$n = 200$		$n = 400$		$n = 800$	
	MB-ECM	naiveEM	MB-ECM	naiveEM	MB-ECM	naiveEM
RASE(π_1)	0.065	0.138	0.051	0.145	0.038	0.143
	(0.048)	(0.072)	(0.019)	(0.067)	(0.014)	(0.057)
RASE($f_{\tilde{\boldsymbol{\theta}}}$)	0.072	0.099	0.054	0.090	0.039	0.082
	(0.020)	(0.027)	(0.014)	(0.028)	(0.010)	(0.025)
ASE($\boldsymbol{\beta}_1$)	0.241	0.606	0.013	0.207	0.006	0.073
	(2.105)	(3.410)	(0.019)	(1.858)	(0.008)	(0.941)
ASE($\boldsymbol{\beta}_2$)	0.217	0.488	0.006	0.152	0.003	0.050
	(2.053)	(2.921)	(0.007)	(1.605)	(0.004)	(1.014)
KS	0.018	0.024	0.013	0.023	0.009	0.020
	(0.009)	(0.012)	(0.007)	(0.012)	(0.005)	(0.009)
ARI	0.748	0.679	0.754	0.655	0.762	0.659
	(0.066)	(0.114)	(0.043)	(0.110)	(0.033)	(0.095)

Example 2 For our next example, we generate data from the same $K = 2$ component SPGMRVPs model (1.17) as in Example 1, however, with mixing proportion functions

$$\pi_1(t) = 0.1 + 0.8\sin(\pi t) \quad \text{and} \quad \pi_2(t) = 1 - \pi_1(t) \tag{5.67}$$

The true mixing proportion functions and a scatter plot of a typical sample of size $n = 400$ are displayed in Figure 5.7. The mixing proportion functions are parabolic functions of t. The estimation of these functions is expected to be difficult due to the intersection between the functions.

Figure 5.8 shows the fitted mixing proportion functions using the MB-ECM (left-panel) and the naiveEM (right-panel) for three randomly selected samples of size 200, 400 and 800, respectively. As in the first example, the naiveEM is unstable and gives unreliable estimates of the mixing proportion functions. As before, this persists even if we increase the sample size. On the other hand, the MB-ECM exhibits stability and its estimates appear to be reliable as they are close to the true functions. To provide further support this last statement, Table 5.12 gives the average and standard deviation of the performance measures for the MB-ECM and the naiveEM calculated over the 500 simulated samples of sizes $n = 200$, 400 and 800. Based on the tabulated results we can again say that the MB-ECM overcomes the challenges experienced by the naiveEM, namely, unstable and, hence unreliable, estimates and sensitivity to label-switching.

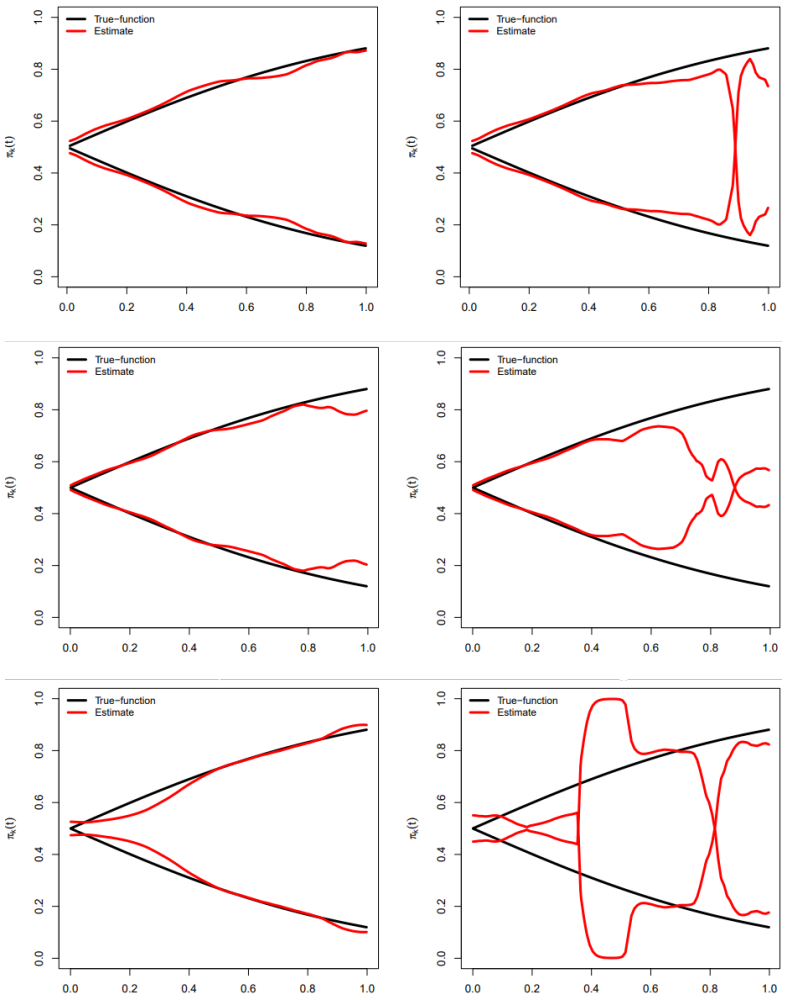


Figure 5.6: Fitted mixing proportion functions using the MB-ECM (**left panel**) and the naiveEM (**right panel**) for randomly selected samples of size $n = 200$ (**first row**), 400 (**second row**) and 800 (**third row**) generated from the model in Example 1.

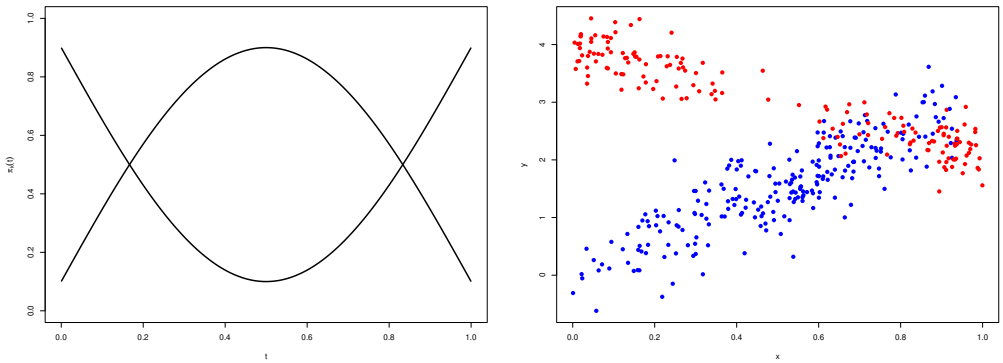


Figure 5.7: (right-panel) The $K = 2$ mixing proportion functions: two parabolic functions $\pi_1(t)$ and $\pi_2(t)$. (left-panel) A scatter plot of a typical sample of size $n = 400$. The red data points are from component 1 and the blue data points are from component 2.

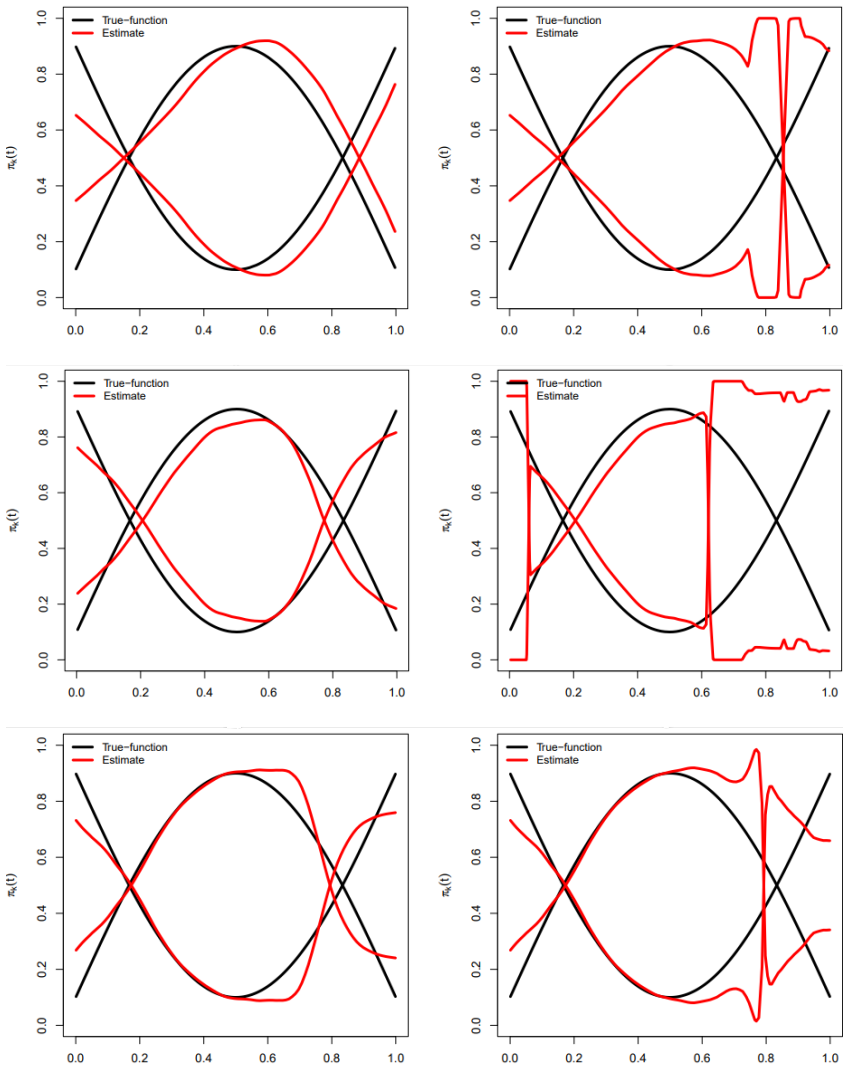


Figure 5.8: Fitted mixing proportion functions using the MB-ECM (**left-panel**) and using the naiveEM (**right-panel**) for randomly selected samples of size $n = 200$ (**first row**), 400 (**second row**) and 800 (**third row**) generated from the model in Example 2.

Application In this section, we demonstrate the practical usefulness of the proposed method on real data. For real data analysis,

1. we measure the goodness-of-fit using the Bayesian information criterion (BIC)

$$\text{BIC} = -2\ell + \text{df} \times \log(n) \tag{5.68}$$

where ℓ is the maximum log-likelihood value and $\text{df} = (K-1)\text{df}_\pi + (\text{df}_{\boldsymbol{\beta}} + 1)K$ is the

overall model degrees of freedom, with $\text{df}_{\boldsymbol{\beta}} = (D+1)$ and df_{π} is the degrees of freedom of a one-dimensional varying coefficient function (Huang et al. [2013], see page 933). Moreover, we assess the predictive ability of the fitted model using the mean squared prediction error (MPSE). Following Xiang and Yao [2016], we calculate the MSPE via a Monte Carlo cross validation (MCCV) procedure. The MCCV procedure randomly partitions the data into a training set with size $n(1-r)$ and a test set with size nr, where r is the proportion of data in the test set. The model is estimated using the data in the training set and then validated using data in the test set. The procedure is repeated T times and we take the average of the MSPEs. We take $r = 0.1, 0.2$ and 0.3. Moreover, we set $T = 200$; and

2. lastly, we use a conditional bootstrap approach to calculate the pointwise 95% confidence intervals of the fitted mixing proportion functions and the 95% confidence intervals of the component regression coefficients and variances. That is, for a given value of x, we sample the corresponding value of the response, denoted by y^*, from the fitted SPGMRVPs model $\sum_{k=1}^{K} \hat{\pi}_k(t)\mathcal{N}(y|\mathbf{x}^\intercal\hat{\boldsymbol{\beta}}_k, \hat{\sigma}_k^2)$. We repeat this sampling process n times to get a bootstrap sample $\mathcal{S} = \{(x_i, y_i^*) : i = 1, 2, \ldots, n\}$. We generate B bootstrap samples $\mathcal{S}^{(1)}, \mathcal{S}^{(2)}, \ldots, \mathcal{S}^{(B)}$ in the above manner. We fit the SPGMRVPs model (1.17) on each of these bootstrap samples, thus generating a sampling distribution of $\hat{\pi}_k(t)$, $\hat{\boldsymbol{\beta}}_k$ and $\hat{\sigma}_k^2$. To compute the 95% confidence intervals, we take the 2.5^{th} and 97.5^{th} percentiles of the

Table 5.12: Average (and standard deviation) of the performance measures for the MB-ECM and naiveEM over the 500 simulated samples of sizes $n = 200$, 400 and 800 generated from the model in Example 2.

	$n = 200$		$n = 400$		$n = 800$	
	MB-ECM	naiveEM	MB-ECM	naiveEM	MB-ECM	naiveEM
RASE(π_1)	0.126	0.161	0.082	0.132	0.071	0.117
	(0.027)	(0.058)	(0.017)	(0.046)	(0.011)	(0.035)
RASE($f_{\hat{\boldsymbol{\theta}}}$)	0.117	0.126	0.081	0.091	0.064	0.074
	(0.020)	(0.023)	(0.014)	(0.020)	(0.010)	(0.012)
ASE($\boldsymbol{\beta}_1$)	0.008	0.251	0.004	0.045	0.002	0.002
	(0.011)	(2.199)	(0.005)	(0.900)	(0.002)	(0.003)
ASE($\boldsymbol{\beta}_2$)	0.024	0.264	0.012	0.076	0.005	0.007
	(0.032)	(2.149)	(0.017)	(1.017)	(0.007)	(0.010)
KS	0.026	0.031	0.018	0.026	0.014	0.021
	(0.009)	(0.012)	(0.007)	(0.013)	(0.005)	(0.010)
ARI	0.637	0.619	0.654	0.629	0.656	0.634
	(0.071)	(0.079)	(0.048)	(0.057)	(0.034)	(0.042)

sampling distributions as the lower and upper limits, respectively, of the interval. We set $B = 200$.

CO_2 data In this application, we consider data on per capita gross national product (GNP) and carbon dioxide (CO_2) emissions in 1996 for a group of 28 OECD countries. The data was obtained from the R package `mixtools` (Benaglia et al. [2009]). A scatter plot of the data is given in Figure 5.9. Each data point on the figure is accompanied by the corresponding country's code. For instance, NOR is for Norway and MEX is for Mexico.
Previous studies of this data have found two linear regression components corresponding to two developmental paths (Hurn et al. [2003] and Young and Hunter [2010]). The first linear regression function is steep and has a positive slope (higher GNP, higher CO2 emissions) whereas the second function is almost flat with a negative slope (higher GNP, lower CO2 emissions). According to Hurn et al. [2003], analysis of this data could assist in identifying the development paths of lower GNP countries in order to ensure that they do not pursue economic growth (high GNP) at an expense to the environment (high CO_2). This is an important step towards achieving the 13th Sustainable Development Goal (SDG) of taking urgent action against climate change and its impacts.
From Figure 5.9, it does seem like most high GNP countries have low per-capita CO_2 emissions. This implies that, as per capita GNP increases so does the probability of a country belonging to the higher GNP, lower CO2 component. In other words, the mixing proportion is not constant. It is instead a function of per capita GNP. Thus, we fit the $K = 2$ component SPGMRVPs

$$f(y|X = x, T = t) = \pi(t)\mathcal{N}(y|\mathbf{x}^\intercal\boldsymbol{\beta}_1, \sigma_1^2) + (1 - \pi(t))\mathcal{N}(y|\mathbf{x}^\intercal\boldsymbol{\beta}_2, \sigma_2^2) \tag{5.69}$$

where $y = CO_2$, $\mathbf{x} = (1, x)^\intercal$ and $x = t = \text{GNP}$.
We fit model (5.69) using the proposed method. The optimal bandwidth was chosen to be $h = 22.7$. We compare the fitted model with the $K = 2$ component GMLRs (2.19) model which assumes that the mixing proportion function is constant. Table 5.13 gives the estimated parameters along with the boostrap standard errors. The estimated parameters based on the two models are similar. To evaluate the goodness of fit and the prediction performance of the fitted models, Table 5.13 and Table 5.14 gives the BIC and MSPE, respectively. As can be seen from the table, the fitted model based on the proposed method gives the best fit to the data. The first and second column of Figure 5.10 shows the fitted component regression lines (first row) and the fitted mixing proportion functions for the first (second row) and second (third row) regression components obtained from the fitted model (5.69) and the fitted GMLRs model. For the fitted component regression plots, we also include the hard clustered data points based on the responsibilities $\hat{\gamma}_{ik}$. An interpretation of the results on the figure suggests that,

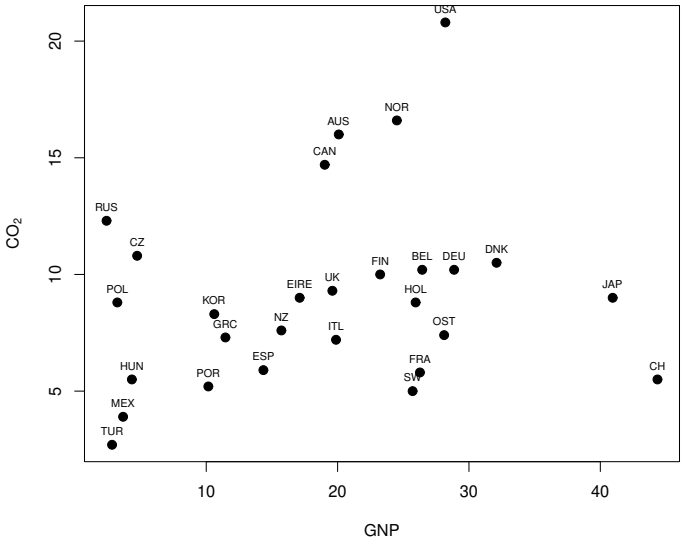


Figure 5.9: Scatter plot of the CO_2 data. Each data point is accompanied by the corresponding country's code. For instance, NOR - Norway and MEX - Mexico.

in 1996, low GNP countries such as Turkey (TUR) and Mexico (MEX) were on a higher GNP and higher CO_2 emissions developmental path. On the other hand, low GNP countries like Czech-Republic (CZ) and Poland (POL) were on a higher GNP and lower CO_2 developmental path. According to the Our World in Data (Accessed on 12 December 2023), per capita CO_2 emissions in Turkey and Mexico have increased over the period 1996 - 2022, whereas per capita CO2 emissions in Czech-Republic and Poland have decreased over the same period. This is an indication that, indeed, these countries have been on their respective development paths, as determined by the model.

For the fitted mixing proportion functions, we also include the 95% bootstrap pointwise con-

.

Table 5.13: Estimated parameters (and the bootstrap standard errors) for the fitted models. The BIC is also provided to assess the goodness-of-fit of each model.

	$\boldsymbol{\beta}_1$		$\boldsymbol{\beta}_2$		σ_1^2	σ_2^2	BIC
SPGMRVPs	1.457	0.675	8.778	-0.027	0.664	4.137	**153.418**
	(1.586)	(0.092)	(1.408)	(0.089)	(0.367)	(1.848)	
GMLRs	1.415	0.677	8.679	-0.023	0.655	4.200	157.205
	(1.449)	(0.123)	(0.940)	(0.042)	(0.832)	(1.868)	

fidence intervals. The fitted mixing proportion functions shows a gradual decrease in the probability that a country belongs to the high GNP and high CO2 emissions development path as per capita GNP increases. Consequently, the probability of belonging to the high GNP and low CO2 development path is increasing as per capita GNP increases.

Discussion and Conclusion This essay was concerned with estimating semi-parametric Gaussian mixtures of regressions with varying mixing proportions (SPGMRVPs) (1.17) using the proposed model-based one-step backfitting procedure in order to demonstrate the effectiveness and practical utility of this estimation procedure in addressing label-switching and producing useful model estimates, respectively.

We used intensive Monte Carlo simulations to show that the naive estimation procedure (naiveEM) may lead to wiggly, non-smooth and ultimately practically unreliable estimates of the non-parametric mixing proportion functions. Thus, the naiveEM, is unstable and sensitive to label-switching. On the other hand, we showed that the proposed model-based procedure (MB-ECM) overcomes the challenges of the naiveEM. The MB-ECM is stable, less sensitive to label-switching and tended to result in estimates of the mixing proportion functions that were consistent with theory. Moreover, the MB-ECM, was shown to always outperform the naiveEM in model fit accuracy and clustering capability as measured by the KS statistic and ARI, respectively.

In our real data analysis, we demonstrated the practical utility of the MB-ECM algorithm for fitting a SPGMRVPs model on a classical CO_2-GNP dataset. As is usual with this data, a $K = 2$ component SPGMRVPs model was fitted, where the two components are interpreted as two developmental paths. A high GNP - high CO_2 path and a high GNP - low CO_2 path. Our fitted model revealed more insights about this data. The model was able to assign countries into their respective developmental paths. Using recent data (2022), we were able to confirm that indeed these countries followed that trajectory or developmental path. These results places this and other mixture of regressions models as parsimonious contenders for monitoring the global effort towards achieving the 13^{th} SDG of taking action against the effects of climate

Table 5.14: The average (and standard deviation) of the mean square prediction error (MSPE) of the fitted models for values of $r = 0.1, 0.2$ and 0.3, where r is as defined in the text.

$r = 0.1$		$r = 0.2$		$r = 0.3$	
SPGMRVPs	GMLRs	SPGMRVPs	GMLRs	SPGMRVPs	GMLRs
13.420 (7.847)	15.951 (10.580)	14.926 (8.536)	15.494 (9.367)	15.801 (7.063)	16.474 (7.972)

change by lowering greenhouse gas emissions.

5.4 Conclusion

In this chapter, we proposed another novel estimation procedure in an effort to address the label-switching problem when estimating any model of the form (1.1). The proposed approach proceeds by first reformulating the model as a mixture of local Gaussian mixture models (GMMs), where each local grid point represents a component of the mixture. Finally, using the usual EM algorithm, we can estimate this mixture of GMMs which is equivalent to simultaneously maximising the local-likelihood functions using the same posterior probabilities along similar lines as the objective-based approach. Due to its mixture of mixtures structure, a popular and useful structure for model-based clustering (see Punzo and McNicholas [2016]), we refer to this novel estimation procedure as the model-based estimation procedure.
Alternatively, we can use the Expectation-Conditional-Maximisation (ECM) algorithm to first automatically choose the number and location of the local grid points. This is equivalent to choosing the number of components. Finally, estimate the resulting mixture of GMMs. This approach is more advantageous given its additional feature of automatically choosing the local points. Moreover, this is the approach used in our numerical demonstrations. It is interesting to note that this alternative estimation procedure encompasses, as a special case, the objective-based estimation procedure. We can choose the global responsibilities as the local responsibilities that belong to the local GMM with the largest mixing weight. See the text for more details on the significance of the weights assigned to each local GMM.
The effectiveness and practical usefulness of the proposed method was demonstrated using using intensive Monte Carlo simulations and real data analysis, respectively. First, compared with the naïve estimation procedure (see subsection 1.2.1), the proposed approach is less sensitive to label-switching and produces reasonably smooth estimates of the non-parametric functions that are in line with expectations. Second, the proposed method performs at least as well as a competitive approach that uses a similar idea as in the effective EM algorithm.
For illustrative purposes, the efficacy of the proposed method was demonstrated for the case of estimating two special cases of the general model (1.1). However, the method is applicable for estimating any model of the form (1.1).

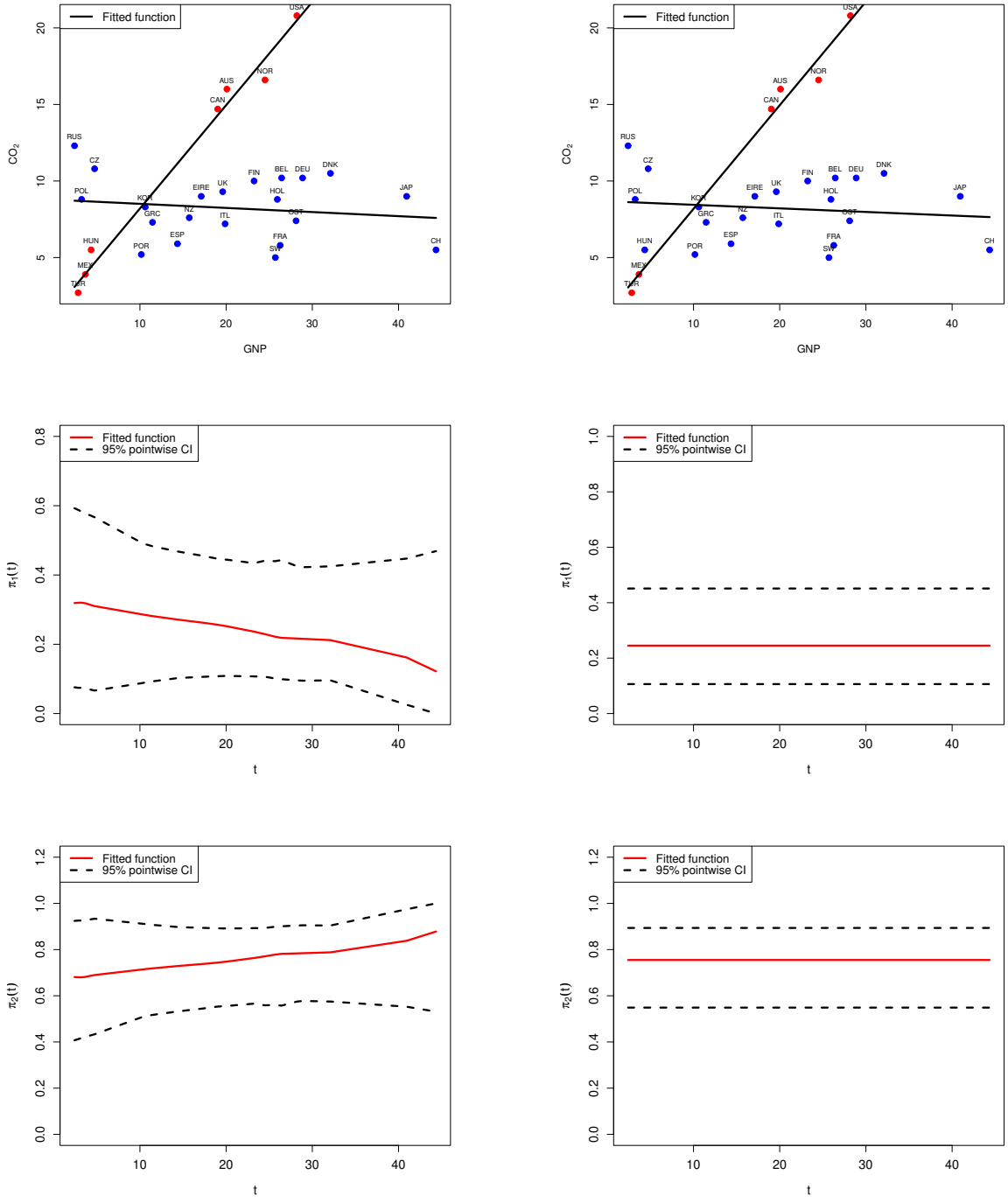


Figure 5.10: Fitted component linear regression functions (first row) with hard clustered data points, fitted component mixing proportion functions for the first component (second row) and the second component (third row) based on model (5.69) (first column) and the GMLRs model (second column). Also included are the 95% bootstrap pointwise confidence intervals.

Chapter 6

Conclusion and Future research

6.1 Conclusion

The focus of this thesis was centered around the development of estimation techniques to address label-switching when estimating a flexible class of finite Gaussian mixtures of regression models.

In chapter 1, we gave a formal description of this general model which is an ensemble of an additive model, a partial linear model, a varying-coefficient model and a Gaussian mixture of linear regressions model. Moreover, this model assumes that the mixing proportions and/or variances are non-parametric functions of a covariate.

In chapter 3, we proposed two estimation procedures to estimate the general model. Next, we showed that a local-likelihood estimation of the non-parametric functions via the traditional Expectation-Maximisation (EM) algorithm may be subject to label-switching. To estimate the non-parametric functions, we have to define a local-likelihood function for each local grid point on the domain of a given covariate. If we separately maximise each local-likelihood function, via the EM algorithm, the labels attached to the mixture components may experience label switching from one local grid point to the next.
The practical consequence of label-switching is characterised by non-smooth and discontinuous non-parametric functions exhibiting irregular and non-uniform behaviour at local points where the label-switch took place. Finally, we gave a formal description and illustration of this label-switching problem.

To address label-switching and thus achieve objective 1 of this thesis, we develop new EM-type estimation strategies. We proposed two novel estimation techniques to address label-

switching. The common feature among the proposed methods is that the estimation takes place simultaneously at different local grid points instead of separately as in the traditional naively implemented EM estimation approach.

In chapter 4, we proposed the first approach to address label-switching. The proposed approach is based on the idea of using the same responsibilities (global responsibilities) at each local M-step of the EM algorithm. The proposed approach proceeds in two stages. In the first-stage, we estimate all the local responsibilities at each local grid point. In the second-stage, based on an appropriate objective function, we choose one set of local responsibilities, among those estimated in the first-stage, as the global responsibilities. Finally, we replace the local responsibilities at each local E-step by the global responsibilities and then proceed to the M-step.

In chapter 5, we proposed a second approach to address label-switching. The proposed approach proceeds by first reformulating the model as a mixture of local Gaussian mixture models (GMMs), where each local GMM or grid point represents a component of the mixture. Finally, using the usual EM algorithm, we can estimate this mixture of GMMs which is, in effect, equivalent to simultaneously maximising (estimating) the local-likelihood functions (local parameters). We can also use the Expectation-Conditional-Maximisation (ECM) algorithm to first, "automatically" select the number and location of the local grid points, from an initial set of grid points. This is equivalent to choosing the number of components. Finally, estimate the resulting mixture of GMMs. This approach is more advantageous given its additional feature of automatically choosing the local points.

To achieve objective 3, we provide comprehensive summaries of the estimation procedures that we proposed in this thesis. Moreover, we provide the R code used in our numerical experiments in a public repository (see Appendix A).

The effectiveness of the proposed estimation strategies was demonstrated using intensive Monte Carlo simulations. Moreover, to achieve objective 2 of this thesis, we demonstrate the practical usefulness of the proposed methods on real data.

6.2 Future studies

In this section, we provide directions for further research and possible extensions of some aspects of this thesis.

6.2.1 Objective-based approach: other possible objective functions

In this thesis, we proposed a novel approach (Objective-based) to address label switching. This approach involves optimising an objective function. As our objective function, we used

the roughness function due to its intuitive interpretation as a measure of the degree to which a given function is wiggly and non-smooth as a result of label-switching. Furthermore, we mentioned, among other things, that any function that is intuitively appealing and analytically advantageous, can be used as an objective function. There are a lot of functions that satisfy these two conditions such as the classification likelihood function, mutual information, coefficient of determination and entropy, to mention a few. An investigation of the usefulness of any of these and other possible objective functions is a topic that may be of interest for future research.

6.2.2 Model-based approach: choosing the parameter λ_0

In this thesis, we proposed a novel approach (Model-based) to address label-switching in which the original Gaussian mixture of non-parametric regressions model is reformulated as a mixture of Gaussian mixture models. Associated with each component of the mixture of GMM is a (mixing) weight λ at a local grid point u. As a way to "automatically" choose the local grid points to be used in the estimation, we introduced a parameter λ_0 as a threshold mixing weight. The local mixing weights, the λ's, are tracked at each iteration of the fitting algorithm. A local model with a mixing weight below λ_0 is removed and the iterations continue without the local model or grid point. This process continues until there is no further removal of a local model. We mentioned that λ_0 is a hyperparameter that can be chosen subjectively or objectively. Following an intensive Monte Carlo study, which showed, empirically, that the estimation procedure is not sensitive to the choice of λ_0, in this thesis λ_0 was chosen subjectively based on the results of the simulation study. However, as with the choice of the smoothing parameter or bandwidth h, λ_0 cannot be chosen to be too small or too large. In the first extreme case, the algorithm may not be effective in choosing the appropriate number and location of the local grid points whereas in the second extreme case it may cease to function. It can be difficult to subjectively maintain a balance between efficacy and functionality. The choice of λ_0 can be done objectively by making use of a data-driven approach such as the cross-validation. The investigation of the potential of this and other objective approaches to choosing λ_0 should be a research topic for future study.

6.2.3 Generalised linear modelling (GLM) framework

In this thesis, we considered only continuous (Gaussian) response variables. However, the methods proposed in this thesis can be easily extended for the case of discrete (Binomial or Poisson) response variables. Given the recent growth in interest for the theoretical and practical study of models of the form (1.1), an extension of the proposed methods to the generalised linear modelling (GLM) framework can be considered for future study.

6.2.4 Extension to higher-order local polynomial likelihood (LPL) estimators

At the close of chapter 3, we mentioned that, in order to estimate, among other things, the non-parametric mixing proportion functions, we are limited to using local-constant or Nadaraya-Watson estimators, that is LPL estimators with a local polynomial degree of $p = 0$. This is due to the fact that the mixing proportion functions have no closed form expressions for higher-order LPL estimators ($p > 0$). Given the advantage of LPL estimators with $p > 0$ over $p = 0$, in terms of bias reduction when say, the true function has a first derivative or curvature, it may be desirable and consequently advantageous to use higher-order LPL estimators. For future studies, we will consider the benefits of extending higher-order LPL estimators to estimate all the non-parametric functions of the general GMNPRs model (1.1).

6.2.5 Some open areas for theoretical research

Note that the performance of the estimators and algorithms proposed in this thesis (in chapter 3, 5 and 4) was demonstrated using Monte Carlo simulations. It would be interesting to theoretically study the statistical properties of the proposed estimators. Another interesting area for future research include the theoretical study of the convergence of the proposed algorithms (Objective-based EM algorithm and Model-based EM algorithm).

Bibliography

H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

R. E. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

T. Benaglia, D. Chauveau, D. R. Hunter, and D. Young. mixtools: An R Package for Analyzing Finite Mixture Models. *Journal of Statistical Software*, 32(6):1–29, 2009.

C. M. Bishop. *Pattern recognition and machine learning*. Springer New York, 2006.

G. E. P. Box and G. C. Tiao. A Bayesian approach to some outlier problems. *Biometrika*, 55 (1):119–129, 1968.

L. Breiman and J. H. Friedman. Estimating optimal transformations for multiple regression and correlation. *Journal of the American Statistical Association*, 80(391):580–598, 1985.

A. Buja, T. Hastie, and R. Tibshirani. Linear smoothers and additive models. *The Annals of Statistics*, 17(2):453–510, 1989.

R. J. Carroll, J. Fan, I. Gijbels, and M. P. Wand. Generalized partially linear single-index models. *Journal of the American Statistical Association*, 92(438):477–489, 1997.

G. Celeux, D. Chauveau, and J. Diebolt. Stochastic versions of the EM algorithm: an experimental study in the mixture case. *Journal of Statistical Computation and Simulation*, 55(4): 287–314, 1996.

P. Craven and G. Wahba. Smoothing noisy data with spline functions. *Numerische Mathematik*, 31(4):377–403, 1979.

C. De Boor. *A practical guide to splines*. Springer-Verlag, New York, 1978.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

W. S. DeSarbo and W. L. Cron. A maximum likelihood methodology for clusterwise linear regression. *Journal of Classification*, 5:249–282, 1988.

S. Dinda. Environmental Kuznets curve hypothesis: a survey. *Ecological Economics*, 49(4): 431–455, 2004.

J. Duffy and C. Papageorgiou. A Cross-Country Empirical Investigation of the Aggregate Production Function Specification. *Journal of Economic Growth*, 5:87–120, 2000.

R. B. Durand, W. H. Greene, M. N. Harris, and J. Khoo. Heterogeneity in speed of adjustment using finite mixture models. *Economic Modelling*, 107:105713, 2022. doi:https://doi.org/10.1016/j.econmod.2021.105713.

J. Fan. Design-adaptive nonparametric regression. *Journal of the American Statistical Association*, 87(420):998–1004, 1992.

J. Fan and I. Gijbels. *Local polynomial modelling and its applications: monographs on statistics and applied probability 66*. CRC Press, 1996.

J. Fan and W. Zhang. Statistical estimation in varying coefficient models. *The Annals of Statistics*, 27(5):1491–1518, 1999.

J. Fan and W. Zhang. Statistical methods with varying coefficient models. *Statistics and its Interface*, 1(1):179, 2008.

J. H. Friedman and W. Stuetzle. Projection pursuit regression. *Journal of the American Statistical Association*, 76(376):817–823, 1981.

S. Frühwirth-Schnatter. *Finite mixture and Markov switching models*. Springer, 2006.

S. Fruhwirth-Schnatter, G. Celeux, and C. P. Robert. *Handbook of mixture analysis*. CRC Press, 2019.

A. E. Gelfand and A. F. M. Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical association*, 85(410):398–409, 1990.

S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, 1984.

I. J. Good and R. A. Gaskins. Nonparametric roughness penalties for probability densities. *Biometrika*, 58(2):255–277, 1971.

P. J. Green and B. W. Silverman. *Nonparametric Regression and Generalized Linear Models: A roughness penalty approach*. Taylor & Francis, 1994.

D. N. Gujarati and D. C. Porter. *Basic Econometrics*. McGraw Hill Irwin, 5th edition, 2011.

S. A. Hamilton and Y. K. Truong. Local linear estimation in partly linear models. *Journal of Multivariate Analysis*, 60(1):1–19, 1997.

W. Hardle, P. Hall, and H. Ichimura. Optimal smoothing in single-index models. *The Annals of Statistics*, 21(1):157–178, 1993.

W. Härdle, H. Liang, and J. Gao. *Partially linear models*. Springer Science & Business Media, 2000.

W. K. Härdle, M. Müller, S. Sperlich, and A. Werwatz. *Nonparametric and Semiparametric Models*. Springer Berlin, Heidelberg, 2004.

T. Hastie and C. Loader. Local regression: Automatic kernel carpentry. *Statistical Science*, 8 (2):120–129, 1993.

T. Hastie and R. Tibshirani. Generalized additive models: some applications. *Journal of the American Statistical Association*, 82(398):371–386, 1987.

T. Hastie and R. Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 55(4):757–779, 1993.

T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning: Data mining, Inference, and Prediction*. Springer New York, 2009.

T. J. Hastie and R. J. Tibshirani. *Generalized Additive Models*. Taylor & Francis, 1990.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

C. Hennig. Identifiablity of Models for Clusterwise Linear Regression. *Journal of Classification*, 17(2):273–296, 2000.

J. Horowitz, J. Klemelä, and E. Mammen. Optimal estimation in additive regression models. *Bernoulli*, 12(2):271–298, 2006.

M. Huang. *Nonparametric techniques in finite mixture of regression models*. PhD thesis, Pennsylvania State University, 2009. Unpublished.

M. Huang and W. Yao. Mixture of regression models with varying mixing proportions: a semiparametric approach. *Journal of the American Statistical Association*, 107(498):711–724, 2012.

M. Huang, R. Li, and S. Wang. Nonparametric mixture of regression models. *Journal of the American Statistical Association*, 108(503):929–941, 2013.

M. Huang, W. Yao, S. Wang, and Y. Chen. Statistical inference and applications of mixture of varying coefficient models. *Scandinavian Journal of Statistics*, 45(3):618–643, 2018.

L. Hubert and P. Arabie. Comparing partitions. *Journal of Classification*, 2:193–218, 1985.

M. Hurn, A. Justel, and C. P. Robert. Estimating mixtures of regressions. *Journal of Computational and Graphical Statistics*, 12(1):55–79, 2003.

H. Ichimura. Semiparametric least squares (SLS) and weighted SLS estimation of single-index models. *Journal of Econometrics*, 58(1-2):71–120, 1993.

R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991. doi:10.1162/neco.1991.3.1.79.

G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning: with Applications in R*. Springer New York, 2021.

A. Jasra, C. C. Holmes, and D. A. Stephens. Markov Chain Monte Carlo Methods and the Label Switching Problem in Bayesian Mixture Modeling. *Statistical Science*, 20(1):50–67, 2005.

R. Jiang and M. Sun. Single-index composite quantile regression for ultra-high-dimensional data. *TEST*, 31:443–460, 2021. doi:https://doi.org/10.1007/s00181-016-1224-z.

M. Konte. Do remittances not promote growth? A finite mixture-of-regressions approach. *Empirical Economics*, 54:747–782, 2018.

K. H. Lee and L. Xue. Nonparametric finite mixture of Gaussian graphical models. *Technometrics*, 60(4):511–521, 2018.

B. Li. The multinomial logit model revisited: A semi-parametric approach in discrete choice analysis. *Transportation Research Part B: Methodological*, 45(3):461–473, 2011.

K. C. Li. Sliced inverse regression for dimension reduction. *Journal of the American Statistical Association*, 86(414):316–327, 1991.

O. Linton and J. P. Nielsen. A kernel method of estimating structured nonparametric regression based on marginal integration. *Biometrika*, 82(1):93–100, 1995.

O. B. Linton. Miscellanea efficient estimation of additive nonparametric regression models. *Biometrika*, 84(2):469–473, 1997.

C. Loader. *Local regression and likelihood*. Springer New York, 1999.

C. Loader. *locfit: Local Regression, Likelihood and Density Estimation*, 2023. URL https://CRAN.R-project.org/package=locfit. R package version 1.5-9.8.

S. Ma and L. Yang. Oracally efficient two-step estimation for additive regression. In J. S. Racine, S. Liangjun, and U. Aman, editors, *The Oxford Handbook of Applied Nonparametric and Semiparametric Econometrics and Statistics*, pages 149–175. Oxford University Press, 2014.

E. Mammen, O. Linton, and J. Nielsen. The existence and asymptotic properties of a backfitting projection algorithm under weak conditions. *The Annals of Statistics*, 27(5):1443–1490, 1999.

G. J. McLachlan and T. Krishnan. *The EM algorithm and extensions*. John Wiley & Sons, 1997.

P. D. McNicholas. Model-based clustering. *Journal of Classification*, 33:331–373, 2016. doi:https://doi.org/10.1007/s00357-016-9211-9.

X. L. Meng and D. B. Rubin. Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika*, 80(2):267–278, 1993.

S. M. Millard and F. H. J. Kanfer. Mixtures of Semi-Parametric Generalised Linear Models. *Symmetry*, 12(4), 2022. doi:10.3390/sym14020409.

E. A. Nadaraya. On estimating regression. *Theory of Probability & Its Applications*, 9(1): 141–142, 1964.

S. Newcomb. A generalized theory of the combination of observations so as to obtain the best result. *American Journal of Mathematics*, 8(4):343–366, 1886.

S. K. Ng, X. Liming, and K. K. Wing-Yau. *Mixture Modelling for Medical and Health Sciences*. CRC Press, 2019.

P. Patil, Y. Wu, and R. Tibshirani. Failures and successes of cross-validation for early-stopped gradient descent. In *International Conference on Artificial Intelligence and Statistics*, pages 2260–2268. PMLR, 2024.

K. Pearson. Contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London. A*, 185:71–110, 1894.

A. Punzo and P. D. McNicholas. Parsimonious mixtures of multivariate contaminated normal distributions. *Biometrical Journal*, 58(6):1506–1537, 2016.

R. E. Quandt. A new approach to estimating switching regressions. *Journal of the American Statistical Association*, 67(338):306–310, 1972.

R. E. Quandt and J. B. Ramsey. Estimating mixtures of normal distributions and switching regressions. *Journal of the American Statistical Association*, 73(364):730–738, 1978.

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2023. URL https://www.R-project.org/.

P. M. Robinson. Root-n-consistent semiparametric regression. *Econometrica*, 56(4):931–954, 1988.

P. Schlattmann. *Medical applications of finite mixture models.* Springer Berlin, Heidelberg, 2009.

G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2):461–464, 1978.

D. W. Scott. *Multivariate Density Estimation: Theory, Practice, and Visualization.* John Wiley & Sons, 1992.

S. B. Skhosana, F. H. J. Kanfer, and S. M. Millard. Fitting Non-Parametric Mixture of Regressions: Introducing an EM-Type Algorithm to Address the Label-Switching Problem. *Symmetry*, 14(5):1058, 2022.

P. Speckman. Kernel smoothing in partial linear models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(3):413–436, 1988.

M. Stephens. Dealing with label switching in mixture models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(4):795–809, 2000.

T. M. Stoker. Consistent estimation of scaled coefficients. *Econometrica*, 54(6):1461–1481, 1986.

C. J. Stone. Additive regression and other nonparametric models. *The Annals of Statistics*, 13 (2):689–705, 1985.

D. M. Titterington, A. F. M. Smith, and U.E. Makov. *Statistical Analysis of Finite Mixture Distributions*. John Wiley & Sons, 1985.

G. Wahba. Practical approximate solutions to linear operator equations when the data are noisy. *SIAM Journal on Numerical Analysis*, 14(4):651–667, 1977. doi:10.1137/0714044.

J. Wang and L. Yang. Efficient and fast spline-backfitted kernel smoothing of additive models. *Annals of the Institute of Statistical Mathematics*, 61(3):663–690, 2009.

L. Wang and L. Yang. Spline-backfitted kernel smoothing of nonlinear additive autoregression model. *Annals of Statistics*, 35(6):2474–2503, 2007. doi:10.1214/009053607000000488.

G. S. Watson. Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, 26(4):359–372, 1964.

H. Wu and J. Zhang. *Nonparametric regression methods for longitudinal data analysis: mixed-effects modeling approaches*. John Wiley & Sons, 2006.

X. Wu and T. Liu. Estimation and testing for semiparametric mixtures of partially linear models. *Communications in Statistics-Theory and Methods*, 46(17):8690–8705, 2017. doi:https://doi.org/10.1080/03610926.2016.1189569.

Y. Xia, H. Tong, and W. K. Li. Single-index volatility models and estimation. *Statistica Sinica*, 12(3):785–799, 2002.

S. Xiang and W. Yao. Semiparametric mixtures of nonparametric regressions. *Annals of the Institute of Statistical Mathematics*, 70:131–154, 2016. doi:https://doi.org/10.1007/s10463-016-0584-7.

S. Xiang and W. Yao. Semiparametric mixtures of regressions with single-index for model based clustering. *Advances in Data Analysis and Classification*, 14(2):261–292, 2020. doi:https://doi.org/10.1007/s11634-020-00392-w.

L. Xu and M. I. Jordan. On convergence properties of the EM algorithm for Gaussian mixtures. *Neural Computation*, 8(1):129–151, 1996. doi:https://doi.org/10.1162/neco.1996.8.1.129.

J. Xue and W. Yao. Machine learning embedded semiparametric mixtures of regressions with covariate-varying mixing proportions. *Econometrics and Statistics*, 22:159–171, 2022. doi:https://doi.org/10.1016/j.ecosta.2021.10.018.

J. Xue, W. Yao, and S. Xiang. Machine learning embedded EM algorithms for semiparametric mixture regression models. *Computational Statistics*, pages 1–20, 2024.

L. Xue and L. Yang. Estimation of semi-parametric additive coefficient model. *Journal of Statistical Planning and Inference*, 136(8):2506–2534, 2006.

W. Yao. Model based labeling for mixture models. *Statistics and Computing*, 22:337–347, 2012. doi:https://doi.org/10.1007/s11222-010-9226-8.

D. S. Young and D. R. Hunter. Mixtures of regressions with predictor-dependent mixing proportions. *Computational Statistics & Data Analysis*, 54(10):2253–2266, 2010.

H. M. Yousof and A. M. Gad. Bayesian estimation and inference for the generalized partial linear model. *International Journal of Probability and Statistics*, 4(2):51–64, 2015.

P. Zeng. Finite mixture of heteroscedastic single-index models. *Open Journal of Statistics*, 2 (1):12–20, 2012. doi:10.4236/ojs.2012.21002.

Y. Zhang and W. Pan. Estimation and inference for mixture of partially linear additive models. *Communications in Statistics-Theory and Methods*, 51(8):2519–2533, 2022.

Y. Zhang and Q. Zheng. Semiparametric mixture of additive regression models. *Communications in Statistics-Theory and Methods*, 47(3):681–697, 2018.

Appendix A

R programs

The simulations and real data analysis conducted in chapter 4 and 5, the proposed estimation methods were written and implemented in the R programming software (R Core Team [2023]). The R code used to perform these numerical experiments has been made available on a public repository: https://github.com/Sphiwe-Skhosana/GMNRs.