Journal of Machine Learning Research 24 (2023) 1-47 Submitted 7/21; Revised 8/22; Published 1/23

Regularized Joint Mixture Models

Konstantinos Perrakis KONSTANTINOS.PERRAKIS@DURHAM.AC.UK
Department of Mathematical Sciences
Durham University, UK

Thomas Lartigue THOMAS.LARTIGUE@DZNE.DE
Aramis Project Team, Inria &
Center of Applied Mathematics, CNRS, École Polytechnique, IP Paris, France

Frank Dondelinger FDONDELINGER.WORK@GMAIL.COM
Lancaster Medical School
Lancaster UK

Sach Mukherjee SACH.MUKHERJEE@DZNE.DE
German Center for Neurodegenerative Diseases, Bonn, Germany
& MRC Biostatistics Unit, University of Cambridge, UK

Editor: Samuel Kaski

Abstract

Regularized regression models are well studied and, under appropriate conditions, offer fast and statistically interpretable results. However, large data in many applications are heterogeneous in the sense of harboring distributional differences between latent groups. Then, the assumption that the conditional distribution of response Y given features X is the same for all samples may not hold. Furthermore, in scientific applications, the covariance structure of the features may contain important signals and its learning is also affected by latent group structure. We propose a class of mixture models for paired data (X, Y) that couples together the distribution of X (using sparse graphical models) and the conditional $Y \mid X$ (using sparse regression models). The regression and graphical models are specific to the latent groups and model parameters are estimated jointly. This allows signals in either or both of the feature distribution and regression model to inform learning of latent structure and provides automatic control of confounding by such structure. Estimation is handled via an expectation-maximization algorithm, whose convergence is established theoretically. We illustrate the key ideas via empirical examples. An R package is available at `https://github.com/k-perrakis/regjmix`.

Keywords: distribution shifts, heterogeneous data, joint learning, latent groups, mixture models, sparse regression

1. Introduction

Regularized regression models usually assume homogeneity in the sense that the same conditional distribution of a response Y given features X is taken to hold for all samples. In the presence of latent groups that might have different underlying conditional distributions, regression modeling may be confounded, possibly severely. Similarly, covariance structure among features can be an important signal in scientific applications but its learning may be strongly affected by latent group structure.

These issues are a concern whenever data might harbour unrecognized distributional shifts or group structure, an issue that is increasingly prominent in an era of large and often heterogeneous data. Furthermore, for heterogeneous data the two aspects – the distribution of features X and the conditional $Y \mid X$ – are related in practice, since either or both may contain signals relevant to detecting and modeling group structure, which in turn is essential to overall estimation. Motivated by such heterogeneous data settings with paired data of the form (X, Y), in this paper we study a class of joint mixture models that couple together both aspects – sparse graphical models for X and parsimonious regression models for $Y \mid X$ – in one framework. Specifically, in high-level notation, we consider models of the form

$$\begin{aligned} Z &\sim \tau_Z \\ X \mid Z{=}k &\sim p_X(\mu_k, \Sigma_k) \\ Y \mid X, Z{=}k &\sim p_Y(g(X^T\beta_k), \sigma_k^2) \end{aligned} \tag{1}$$

where $Z \in \{1, \ldots, K\}$ is a latent indicator of group membership with distribution τ_Z, $p_W(m, s)$ denotes the probability distribution of a random variable W with location m and scale s, and $g(\cdot)$ is a link function. This work focuses on the familiar and important case where both X and Y are normally distributed and g is the identity function. However, the key ideas apply to any model of the general form in Eq. (1).

In this context the presence of group structure has the following consequences:

- *Confounding due to latent groups.* Associations between components of X and Y may be entirely different "globally" (with Z marginalized out) vs. "locally" (conditionally on Z) with regression coefficients differing even in signs and sparsity patterns.

- *Ambiguous group structure in feature space.* Clustering the X's alone may lead to cluster labels which do not capture the relevant structure, as instances of group signal in X may be unrelated to Y (e.g. clustering gene expression data may yield well-defined clusters; however, these may not relate to a specific biological/medical response).

- *Group-specific signal in regression coefficients.* Nonidentical coefficients or feature importance across groups provide a potentially useful discriminant signal for identifying the group structure itself. This signal cannot be detected by clustering the X's.

Two common strategies given paired data (X, Y) are: (S1) ignore any potential grouping and fit one regression model using the entire data and (S2) cluster the X's and then fit separate regression models to the group-specific data. Strategy (S1) is risky, since the resulting regression coefficients may be entirely incorrect if latent group structure is present (e.g. due to Simpson's paradox and related phenomena). Also, when the modeling aspect is of importance, under (S1) evaluation of predictive loss is not a satisfactory guide for model assessment, since prediction error may be apparently small despite severe model misspecification. Strategy (S2), although in some ways safer, is also not guaranteed to protect from such effects, unless the resulting group structure obtained from clustering the X's is correct with reference to the overall problem; this may not hold in general. Furthermore, since (S2) models the X data alone, it cannot exploit any signal in the conditionals $Y \mid X$ to guide the clustering. A common variant of (S2) is to perform a dimension reduction on X, such as PCA, and cluster on the reduced space. This, however, does not resolve the problem as the major principal components may not be predictive of Y; see e.g. Jolliffe (1982).

1.1 Related work and contribution

For discrete latent variables Z a standard way of approaching such heterogeneous problems is via mixture models. The literature on mixtures is vast; below we summarize related work according to model structure, covering the most popular approaches for the case of continuous responses.

Mixtures for $X|Z$. The most commonly used and extensively studied approach within this category is the Gaussian mixture model (GMM); see e.g., McLachlan and Peel (2000). GMMs have undergone a series of novel developments focusing on parsimonious modeling of the covariance matrices such as parametrizations based on eigenvalue decomposition (Banfield and Raftery, 1993; Celeux and Govaert, 1995; Fraley and Raftery, 2002), factorizations based on factor analysis models (McNicholas and Murphy, 2008), and extensions to sparse graphical model estimation (Anandkumar et al., 2012; Städler et al., 2017; Fop et al., 2019), among others. These approaches differ from ours in that they consider only the X signal and do not include a regression component, thus, inheriting the potential drawbacks of strategy (S2).

Mixtures for $Y|X, Z$. Finite mixtures of regression (FMR) models belong in this category. Similarly to GMMs, Gaussian FMRs have been studied and developed extensively, allowing for flexible modeling designs (see e.g. Früwirth-Schnatter, 2005) and regularized estimation (Khalili and Jiahua, 2007; Städler et al., 2010; Khalili and Lin, 2013). FMRs focus on the relationship between Y and X without including a generative probability model for X. Our approach is motivated by settings in which the X distribution itself is of interest and may be confounded by latent group structure. Furthermore, under FMRs a *new* X' cannot be allocated to *one specific* group and thereby used to obtain a group-specific prediction.

Mixtures for $Y, Z|X$. Mixtures of experts (MoE; Dayton and Macready, 1988; Jacobs et al., 1991; Jordan and Jacobs, 1994; Jacobs, 1997) jointly model the response and latent allocations. MoEs consist of expert networks (these are models that predict Y from X) and a discriminative model (the gating network) that chooses among the experts. The parsimonious covariance parametrizations for GMMs (Banfield and Raftery, 1993; Celeux and Govaert, 1995; Fraley and Raftery, 2002) have been introduced within the MoE framework initially in Dang and McNicholas (2015) for the special case where the same set of predictors enter the expert and gating networks, and, more recently, in Murphy and Murphy (2020) for the general case where different predictors are allowed to enter in the two networks; the latter work also introduces an additional noise component for outlier detection. Regularized MoE approaches include those of Khalili (2010) and Chamroukhi and Huynh (2018), among others. MoEs include FMRs as a special case in the absence of a gating network. Also, similarly to FMRs, MoEs condition upon features and, thus, lack a generative model for X. However, unlike with FMRs, group-specific prediction of the response is possible under MoEs, as the learned gating network can be used to allocate new feature observations.

Mixtures for $Y, X|Z$. A first approach within this category is profile regression (Molitor et al., 2010; Liverani et al., 2015). Under profile regression X and Y are conditionally independent given the latent group indicator Z. Specifically, the component $Y|Z$ involves a regression model including a "profile" parameter (capturing the effect of X) plus additional co-variates, while the component $X|Z$ is some multivariate distribution (e.g. Gaussian). A second approach, more relevant to our work, is the cluster weighted model (CWM) mixture

introduced by Ingrassia et al. (2012). In this case, we have a linear model component for $Y|X, Z$ and a multivariate distribution component for $X|Z$, following the hierarchical structure of Eq. (1). As illustrated, in Ingrassia et al. (2012) Gaussian CWMs lead to the same family of probability distributions generated by GMMs (see also below) and under specific conditions include FMRs and MoEs as special cases. Extensions of CWMs accommodate the use of GLMs and mixed-type data (Ingrassia et al., 2015; Punzo and Ingassia, 2016), parsimonious parametrizations (similarly to GMMs and MoEs) of covariances (Dang et al., 2017) and latent factor structures for the feature matrix (Subedi et al., 2013), among others.

The class of models proposed here – henceforth, referred to as *regularized joint mixture* (RJM) models – belong to the latter category of mixtures. Specifically, the model specification (the likelihood part of the model) is of a CWM type, but the resulting clustering and parameter learning process under RJMs is different due to regularization. CWMs rely on maximum-likelihood (ML) estimation and in this case under the normal-normal setting with identity link (as considered here), Eq. (1) is equivalent to a GMM on the concatenated matrix $[X, Y]$ as shown in Ingrassia et al. (2012). However, under RJMs, the equivalence to the GMM no longer holds, because the regression and graphical model parts are treated differently (below we include comparisons with direct Gaussian mixture modeling of the concatenated matrix $[X, Y]$). We view regularization as essential for delivering a usable solution to the problem. In many cases the number of features p may be on the same order as sample size n, or larger, and at the group level the sample sizes are of course smaller; hence without suitable regularization both the regression and graphical models will typically be ill-behaved. In the specific implementation we propose, we use the graphical lasso (Friedman et al., 2008a) for graphical model estimation, while for the regression part we consider: (i) the Bayesian lasso (Park and Casella, 2008) and (ii) the normal-Jeffreys prior (Figueiredo, 2001). We note that other choices would be possible within the general framework, subject to computational considerations and appropriate handling of tuning parameters. In summary, the merits of RJMs are the following:

(i) Learns latent group structure by combining information from the distribution over X and the regression of Y on X within a principled framework;

(ii) Provides group-specific feature importance and graphical models with explicit sparsity patterns;

(iii) Applicable in $p > n$ settings;

(iv) Allows group-specific prediction for the response given a new feature vector X'.

1.2 Motivation

Given the large literature on mixture models, it is important to clarify at the outset why the models we study are needed. We are motivated by applications in which latent group structure may be important and where aspects such as (potentially group specific) feature importance and covariance structure among the X's play a role.

To take one example, in biomedicine there is much interest in latent disease subtypes. These will often have subtype-specific covariance patterns, due to differences in underlying

regulatory networks, and the analyst may want to understand subtype-specific disease biology and feature importance. Focusing on only the X's may be insufficient, because this does not account for the response Y (and there may be many ways of clustering X not relevant to the Y of interest). For example, if X are data on human subjects and Y a cancer phenotype, many instances of cluster structure in X may be unrelated to the cancer setting. In addition, focusing solely on differences in regression models $Y|X$ means that subgroup recovery is difficult or impossible if these differences are not large enough. Similarly, in data-driven marketing, latent customer subgroups may have different covariance structure among features and at the same time manifest differences in regression models linking such features to responses (such as revenue per customer). The formulation we propose includes both sources of information in one model and thereby allows for subgroup identification and parameter estimation that accounts for both aspects.

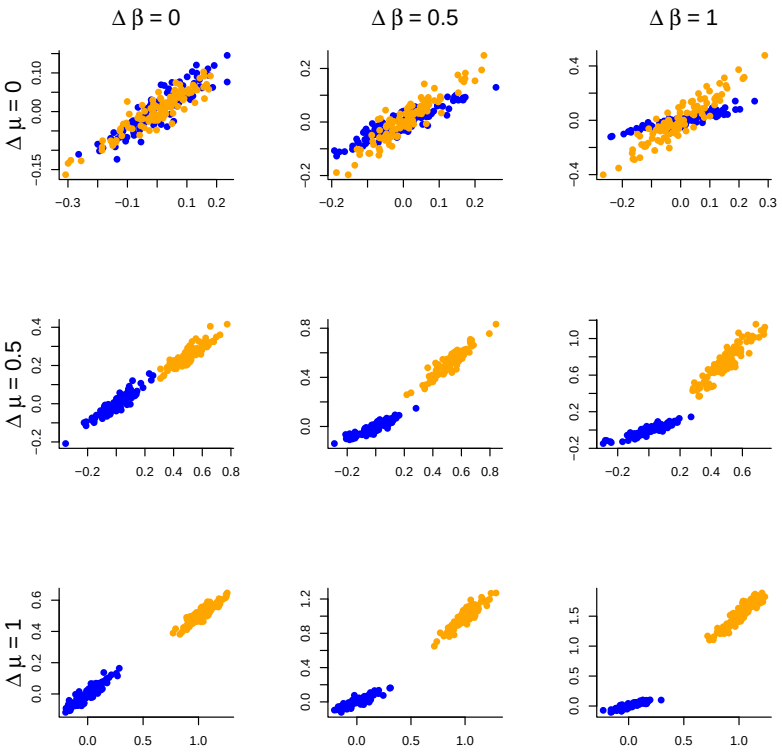

Figure 1: Examples for two subgroups. Each panel shows a specific level of difference in the regression models, quantified by the difference $\Delta\beta$ in regression coefficients. The difference in the feature distributions is controlled via a simple mean shift $\Delta\mu$.

Figure 1 shows simple illustrative examples to bring out some of these points. We emphasize that this is for illustrative purposes only, intended to highlight some interesting contrasts (full empirical results appear in Section 5 below). In these examples, we consider settings in which there may be either or both of an X signal difference (in the figure this is a simple mean shift $\Delta\mu$, but the models we propose are general and for multivariate X allow

also for differences in covariance structure) and a difference in subgroup-specific regression coefficients ($\Delta\beta$). For this initial illustration we consider two latent groups, each with sample size equal to 100, and ten potential predictors, but with only one predictor having an effect (a non-zero regression coefficient) on the response; full details of the simulations can be found in Appendix A. Results in terms of subgroup identification, as quantified by Rand Index, are summarized in Figure 2. When there is no difference in regression models ($\Delta\beta = 0$), MoEs cannot detect any structure (since the X distribution is not modelled). On the other hand, with a stronger difference in regression models ($\Delta\beta = 1$), MoE outperforms a Gaussian mixture (on the X's), since the latter does not model the regression part. The approach we propose models both aspects in a unified framework, hence works well regardless of where the signal lies. Furthermore, and as shown in detail via empirical examples below, by accounting for the latent structure, RJM is able to detect subgroup-specific sparsity patterns whilst avoiding Simpson's paradox-like effects that could otherwise arise. Later, we show detailed empirical results, including an example, based on cancer data, that highlights some of these points, and in particular how subgroup identification benefits from joint modeling, relative to simply clustering X (or clustering the stacked vector (X, Y)) or using MoE.

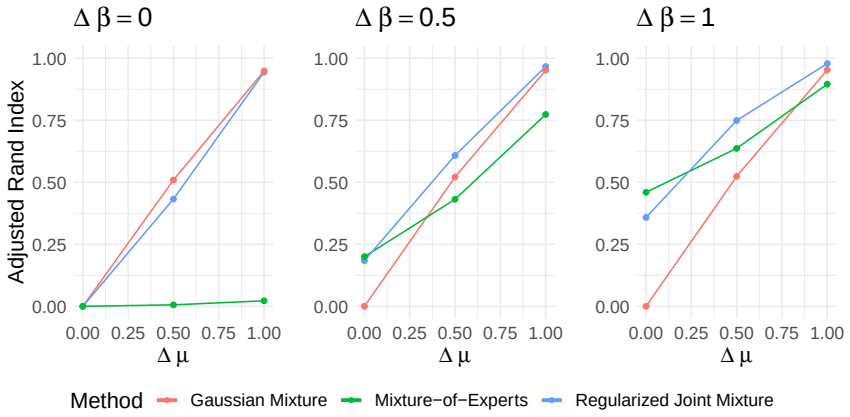

Figure 2: Role of signal location in subgroup identification. The difference $\Delta\beta$ in regression coefficients represents the signal from the regression models, from no signal (left panel), to a strong signal (right panel). In each panel the signal in the feature distribution increases from left to right via a simple mean shift $\Delta\mu$.

The remainder of the paper is structured as follows. In Section 2 we lay out the model specification and discuss the regularization methods under consideration as well as efficient tuning strategies. Computational and theoretical details of the expectation-maximization (EM) optimization are covered in Section 3. In Section 4 we discuss prediction using RJMs and discuss how predictive measures can potentially be used for cluster selection. In Section 5 we present empirical examples, focusing initially on small-scale simulations and then proceeding to larger scale semi-synthetic experiments and applications to real data. The paper concludes with a discussion in Section 6.

2. The RJM model

2.1 Model specification

Let $\mathbf{y}$ denote an n-dimensional vector of outputs or responses and $\mathbf{X}$ an $n \times p$ feature matrix. Samples are indexed by $i = 1, \ldots, n$. Let K denote the number of groups and $z_i \in \{1 \ldots K\}$ represent the true (latent) group indicator for the sample point $(y_i, \mathbf{x}_i)$ with $\Pr(z_i = k) = \tau_k$. The group-specific parameters are $\boldsymbol{\theta}_k = (\boldsymbol{\theta}_k^X, \boldsymbol{\theta}_k^Y)^T$ with $\boldsymbol{\theta}_k^X$ and $\boldsymbol{\theta}_k^Y$ being the parameters governing respectively the marginal distribution of X and the regression model of Y on X.

We allow for group-specific parameters, but assume that samples are independent and identically distributed within groups. The joint distribution of $(y_i, \mathbf{x}_i)$ in group k is

$$p(y_i, \mathbf{x}_i | \boldsymbol{\theta}_k, z_i = k) = p(y_i | \boldsymbol{\theta}_k^Y, \mathbf{x}_i, z_i = k) p(\mathbf{x}_i | \boldsymbol{\theta}_k^X, z_i = k). \tag{2}$$

The features are modeled as p-dimensional multivariate normal so that $\boldsymbol{\theta}_k^X = (\boldsymbol{\mu}_k, \mathrm{vec}(\boldsymbol{\Sigma}_k))^T$, where $\boldsymbol{\mu}_k$ is the mean and $\boldsymbol{\Sigma}_k$ the $p \times p$ covariance matrix. For the responses, we specify a normal linear regression model, with parameters $\boldsymbol{\theta}_k^Y = (\alpha_k, \boldsymbol{\beta}_k, \sigma_k^2)^T$, where α_k is the intercept, $\boldsymbol{\beta}_k$ the vector of regression coefficients and σ_k^2 the error variance. Inclusion of the intercept is necessary in the present setting, because it is not possible to center the response appropriately when the group labels are unknown. Thus, we have that

$$p(\mathbf{x}_i | \boldsymbol{\theta}_k^X, z_i = k) \equiv p(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, z_i = k) = \mathrm{N}_p(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{3}$$

and

$$p(y_i | \boldsymbol{\theta}_k^Y, \mathbf{x}_i, z_i = k) \equiv p(y_i | \alpha_k, \boldsymbol{\beta}_k, \sigma_k^2, \mathbf{x}_i, z_i = k) = \mathrm{N}(y_i | \alpha_k + \mathbf{x}_i^T \boldsymbol{\beta}_k, \sigma_k^2). \tag{4}$$

Marginalizing out the latent variables leads to a mixture representation of the form

$$p(\mathbf{y}, \mathbf{X} | \boldsymbol{\theta}, \boldsymbol{\tau}) = \prod_{i=1}^{n} \sum_{k=1}^{K} p(y_i | \boldsymbol{\theta}_k^Y, \mathbf{x}_i, z_i = k) p(\mathbf{x}_i | \boldsymbol{\theta}_k^X, z_i = k) \tau_k, \tag{5}$$

where $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K)^T$ and $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_K)^T$.

2.2 Regularization and priors

Given the likelihood function of $\boldsymbol{\theta}$ and $\boldsymbol{\tau}$ in (5) we consider general solutions of the form

$$\hat{\boldsymbol{\theta}}, \hat{\boldsymbol{\tau}} = \arg\max_{\boldsymbol{\theta}, \boldsymbol{\tau}} \left\{ \log p(\mathbf{y}, \mathbf{X} | \boldsymbol{\theta}, \boldsymbol{\tau}) + \sum_{k=1}^{K} \mathrm{pen}(\boldsymbol{\theta}_k^{*X}) + \sum_{k=1}^{K} \mathrm{pen}(\boldsymbol{\theta}_k^{*Y}) \right\}, \tag{6}$$

where $\mathrm{pen}(\cdot)$ denotes a penalty function, and $\boldsymbol{\theta}_k^{*X}$ and $\boldsymbol{\theta}_k^{*Y}$ are parameter subsets that we wish to penalize. The particular parameters we penalize are the group-specific covariances and regression vectors; hence, $\boldsymbol{\theta}_k^{*X} \equiv \mathrm{vec}(\boldsymbol{\Sigma}_k)$ and $\boldsymbol{\theta}_k^{*Y} \equiv \boldsymbol{\beta}_k$ in (6) (although under one approach below we consider $\boldsymbol{\theta}_k^{*Y} \equiv (\boldsymbol{\beta}_k, \sigma_k^2)^T$). Penalization is required because the corresponding ML estimates may be ill-behaved or ill-defined and we are interested in group-specific feature importance and conditional independence structure.

In general, tuning of penalty parameters is challenging in the latent group setting. The common approach, based on cross-validation (CV), needs to be handled with care when the estimation of parameters requires iterative procedures converging to (local) maxima, such as the EM algorithm. Specifically, performing CV at each iteration of the algorithm would

change the penalty and, thus, also change the objective function at each iteration. A brute-force solution to the problem would be to pre-specify a grid of values for the penalty and select the value that optimizes a specific criterion. However, apart from open issues related to the range and length of the grid, this would also be a computationally burdensome task, requiring multiple EM processes (each with multiple starts) for each point at the grid.

Given these considerations, we argue that more viable strategies are the following; (i) using "universal penalties" from existing literature, (ii) using CV in a stepwise manner, and (iii) considering the penalties as free parameters under estimation. The first strategy is advisable to use for parameters whose learning is not the main goal of the analysis, but for which regularization is required in order to attain workable, non-spurious solutions. Universal penalties which satisfy certain theoretical requirements (see e.g., Donoho and Johnstone, 1994; Städler and Mukherjee, 2013) typically work well to that end. On the other hand, for the main parameters of interest, whose learning (e.g. estimation, sparsity patterns) is of importance, the second and third strategies seem to be more appropriate as they offer more specific, application-driven solutions. In short, the stepwise CV approach entails adjusting the penalties a few times during the EM and running the algorithm sufficiently long in order to reach local maxima after the last adjustment. The last strategy, requires maximizing the objective with respect to the penalties and, thus, requires the introduction of a prior distribution. Within this framework one could ideally (yet not necessarily) consider properties of the prior; for instance, its behavior under small and/or large samples and the consequent effect on shrinkage, among others. From a pragmatic perspective, the prior choice is linked to more realistic considerations relating to the maximization required in the EM algorithm. For instance, the half-Cauchy prior, which is a standard choice for penalties (Polson and Scott, 2012) in fully Bayesian implementations based on posterior sampling, may not be necessarily convenient to work with within an EM framework.

We proceed with a description of the penalty functions, discussing our choices from both penalized likelihood and Bayesian viewpoints, and explaining our reasoning for the way that penalty parameters are tuned based on the aforementioned strategies. For the remainder of this Section, it is convenient to discuss the corresponding solutions under known group labels. For the subset of datapoints where $z_i = k$, let us denote the $n_k \times 1$ response by $\mathbf{y}_k$ and the $n_k \times p$ predictor matrix by $\mathbf{X}_k$ for $k = 1, \ldots, K$. We emphasize that this is for expositional clarity only; the actual solutions under latent group labels are, of course, obtained iteratively via the EM algorithm presented in Section 3.

2.2.1 Regularization of $\boldsymbol{\Sigma}_k$

For the regularization of $\boldsymbol{\Sigma}_k$ we use the graphical lasso (Meinshausen and Bühlmann, 2006; Yuan and Lin, 2007; Friedman et al., 2008a). The graphical lasso induces sparsity in the inverse covariance matrix $\boldsymbol{\Omega}_k = \boldsymbol{\Sigma}_k^{-1}$ for group k. In this case we have $\text{pen}(\boldsymbol{\theta}_k^{*X}) \equiv \text{pen}(\boldsymbol{\Omega}_k) = -\zeta\|\boldsymbol{\Omega}_k\|_1$ in (6), where $\zeta > 0$ controls the strength of regularization and $\|\cdot\|_q$ is the L_q norm. For known group labels the graphical lasso estimate would be

$$\widehat{\boldsymbol{\Omega}}_k = \underset{\boldsymbol{\Omega}_k \in M^+}{\arg\max} \Big\{ \log|\boldsymbol{\Omega}_k| - \text{tr}\big(\boldsymbol{\Omega}_k \hat{\mathbf{S}}_k\big) - \zeta\|\boldsymbol{\Omega}_k\|_1 \Big\}, \tag{7}$$

where M^+ is the space of positive definite matrices and $\hat{\mathbf{S}}_k$ is the ML covariance estimate of $\mathbf{X}_k$. The solution in (7) is equivalent to the posterior mode under a likelihood as in (3)

and a prior distribution of the following form

$$p(\boldsymbol{\Omega}_k|\psi) \propto \left[\prod_{j=1}^{p} \text{Exp}(\omega_{kjj}|\psi/2) \prod_{j<l,l=2}^{p} \text{DE}(\omega_{kjl}|0,\psi^{-1}) \right] \mathbb{1}_{\{\boldsymbol{\Omega}_k \in M^+\}}, \tag{8}$$

where $\mathbb{1}_{\{\cdot\}}$ is the indicator function, $\text{Exp}(\cdot|r)$ is the exponential distribution with rate $r > 0$ and $\text{DE}(\cdot|\mu, b)$ is the double exponential distribution with location $\mu \in \mathbb{R}$ and scale $b > 0$. The connection between (7) and the corresponding posterior mode under the likelihood in (3) and the prior in (8) is that for any given value of ζ we have that $\psi = n_k\zeta$ (Wang, 2012).

For the graphical lasso penalty, we use the "universal" threshold $\tilde{\psi} = \sqrt{2n \log p}/2$ derived from the logical arguments developed in Städler and Mukherjee (2013). Note that here we use n instead of n_k as the group labels will be unknown. The reason for choosing the universal-threshold approach for the graphical lasso is that we are less interested in the sparsity pattern of $\boldsymbol{\Omega}_k$ itself. Rather, we mainly want a well-behaved estimate that allows group structure to be effectively accounted for.

2.2.2 Regularization of $\boldsymbol{\beta}_k$

The lasso approach. We consider a scaled version of the lasso (Tibshirani, 1996), where the penalty in (6) is given by $\text{pen}(\boldsymbol{\theta}_k^{*Y}) \equiv \text{pen}(\boldsymbol{\beta}_k, \sigma_k^2) = \lambda_k \|\boldsymbol{\beta}_k\|_1/\sigma_k$ with $\lambda_k > 0$. Here we introduce group specific penalty parameters λ_k, unlike previously, where parameter ψ is common across groups. This type of lasso regularization has been studied by Städler et al. (2010) in the context of FMR; a setting where the role of the scaling of variance parameters is much more important than in standard homogeneous regression, as the σ_k's may differ and the grouping is not fixed. If the groups were known, the lasso estimate would be

$$\hat{\alpha}_k, \hat{\boldsymbol{\beta}}_k, \hat{\sigma}_k^2 = \underset{\alpha_k, \boldsymbol{\beta}_k, \sigma_k^2}{\arg\min} \left\{ \frac{\|\mathbf{y}_k - \alpha_k \mathbf{1}_{n_k} - \mathbf{X}_k\boldsymbol{\beta}_k\|_2^2}{2\sigma_k^2} + \lambda_k \frac{\|\boldsymbol{\beta}_k\|_1}{\sigma_k} + (n_k + p + 2)\log\sigma_k \right\}, \tag{9}$$

where $\mathbf{1}_q$ denotes a q-dimensional vector of ones. The solution in (9), which is slightly different than the one in Städler et al. (2010), corresponds to the posterior mode under the Bayesian lasso formulation (Park and Casella, 2008), that specifies independent double exponential priors for the regression coefficients conditional on the error variance which is assigned the scale-invariant Jeffreys prior. Namely,

$$p(\boldsymbol{\beta}_k|\sigma_k^2, \lambda_k) = \prod_{j=1}^{p} \frac{\lambda_k}{2\sigma_k} \exp\left(-\lambda_k \frac{|\beta_{kj}|}{\sigma_k} \right) \text{ and } p(\sigma_k^2) \propto \frac{1}{\sigma_k^2}, \tag{10}$$

with the correspondence to (9) completed when $p(\alpha_k) \propto 1$. We propose two methods for handling λ_k; the *fixed-penalty lasso* (FLasso) based on a plug-in estimate and the *random-penalty lasso* (RLasso) based on the construction of a suitable prior.

FLasso. This approach is essentially a two-step tuning procedure. We start with initial estimates, $\hat{\lambda}_k^{(0)}$ obtained by minimizing the CV mean squared error based on some prior clustering of the data. Then, at a certain iteration we re-calculate the CV estimates and fix each group penalty to the new estimate $\hat{\lambda}_k^{(1)}$ for all further EM iterations. Specifically, we fix the parameter after the first iteration where the group assignments do not change. From

a Bayesian perspective the FLasso approach can be viewed as an empirical Bayes method as we use the data in order to plug-in $\hat{\lambda}_k^{(0)}$ and $\hat{\lambda}_k^{(1)}$ in the prior of $\boldsymbol{\beta}_k$ appearing in (10). The monotonic behaviour of the EM may be disrupted at the re-estimation iteration, but after that point it will hold.

RLasso. In this approach we propose placing a prior distribution on λ_k so that this parameter will be automatically updated during the EM. We construct the prior so that it satisfies the requirement of supporting no penalization asymptotically ($\lambda_k \to 0$ as $n \to \infty$). A suitable prior for our purposes is the Pareto distribution whose scale parameter is also the lower bound of its support. Specifically, we have a prior distribution with scale $a_n > 0$ and shape $b_n > 0$ (parameters are defined to depend on n) of the following form

$$p(\lambda_k) = b_n a_n^{b_n} \lambda_k^{-(b_n+1)}, \tag{11}$$

where $\lambda_k \in [a_n, \infty)$. In our setting parameter a_n does need to be specified explicitly and is regarded to be decreasing in n, while the shape parameter is specified as $b_n = (p-1) - c\sqrt{2K \log p/n}$ for some $c \in (0, 1]$. The rationale for these choices and further details are discussed in Appendix B. As shown next, RLasso will lead to a reasonable update for λ_k during the M-step.

The normal-Jeffreys approach. The normal-Jeffreys (NJ) prior (Figueiredo, 2001) consists of independent improper priors; in our context the prior is given by

$$p(\boldsymbol{\beta}_k) = \prod_{j=1}^{p} p(\beta_{kj}) \propto \prod_{j=1}^{p} |\beta_{kj}|^{-1}. \tag{12}$$

For known group labels with $p(\alpha_k, \sigma_k^2) \propto 1/\sigma_k^2$ the corresponding penalized estimate is

$$\hat{\alpha}_k, \hat{\boldsymbol{\beta}}_k, \hat{\sigma}_k^2 = \underset{\alpha_k, \boldsymbol{\beta}_k, \sigma_k^2}{\arg\min} \left\{ \frac{\|\mathbf{y}_k - \alpha_k \mathbf{1}_{n_k} - \mathbf{X}_k \boldsymbol{\beta}_k\|_2^2}{2\sigma_k^2} + \sum_{j=1}^{p} \log|\beta_{kj}| + (n_k + 2)\log\sigma_k \right\}. \tag{13}$$

Here $\text{pen}(\boldsymbol{\theta}_k^{*Y}) \equiv \text{pen}(\boldsymbol{\beta}_k) = \sum_{j=1}^{p} \log|\beta_{kj}|$. The NJ is well known in the shrinkage-prior literature (Griffin and Brown, 2005; Carvalho et al., 2010; Polson and Scott, 2010). As with most shrinkage priors, (12) can be expressed as a scale-mixture of normals; namely, $p(\beta_{kj}|s_{kj}) = (2\pi s_{kj})^{-1/2} \exp(-\beta_{kj}^2/2s_{kj})$ with $\pi(s_{kj}) \propto s_{kj}^{-1}$ and, therefore, we have that $\int p(\beta_{kj}|s_{kj})\pi(s_{kj})\mathrm{d}s_{kj} \propto |\beta_{kj}|^{-1}$. As the mixing distribution lacks a hyper-parameter the prior is characterized by the absence of a "global" scale parameter. Also, due to heavy tails small coefficients are shrunk a lot, while large signals remain relatively unaffected; similarly to other heavy-tailed priors (Carvalho et al., 2010; Griffin and Brown, 2005). Figueiredo (2003) and Bae and Mallick (2004) show that the NJ prior strongly induces sparsity and yields good performance in terms of selection.

The use of the NJ prior is appealing for the RJM framework. Handling penalties is cumbersome in our setting and the NJ prior provides an attractive "tuning-free" alternative. In general, shrinkage priors which lack a global scale parameter fail to capture the average signal density of the data (Carvalho et al., 2010); however, despite this potential shortcoming of the NJ prior the potential benefits are worth exploring. Also, the posterior mode under

(12) is easy to find through the use of an EM algorithm where the scaling parameters s_{kj} are considered latent (Figueiredo, 2003). This additional latent structure can be easily incorporated in our EM without additional computational costs. In fact, the corresponding NJ-EM update is in closed-form, which is not the case in the lasso approach.

3. The RJM-EM algorithm

In this Section we present first the expectation and maximization steps of the proposed EM algorithm. We then prove that under certain conditions on the regularization the proposed algorithm converges towards a critical point of the likelihood function.

3.1 The EM steps

The E-Step. Irrespective of regularization approach, the group-membership probabilities of the mixture model in (5) at iteration t of the algorithm are calculated as

$$m_{ki}^{(t)} \equiv \widehat{\Pr}(z_i = k|y_i, \mathbf{x}_i, \boldsymbol{\theta}_k^{(t)}) = \frac{p(y_i|\boldsymbol{\theta}_k^{Y(t)}, \mathbf{x}_i, z_i = k)p(\mathbf{x}_i|\boldsymbol{\theta}_k^{X(t)}, z_i = k)\tau_k^{(t)}}{\sum_k p(y_i|\boldsymbol{\theta}_k^{Y(t)}, \mathbf{x}_i, z_i = k)p(\mathbf{x}_i|\boldsymbol{\theta}_k^{X(t)}, z_i = k)\tau_k^{(t)}}, \tag{14}$$

for $i = 1, \ldots, n$, with the distributions appearing in the right-hand side of (14) defined in (3) and (4). Let us define some quantities that will be used throughout; namely, $n_k^{(t)} = \sum_{i=1}^n m_{ki}^{(t)}$, $\mathbf{m}_k^{(t)} = (m_{k1}^{(t)}, \ldots, m_{kn}^{(t)})^T$ and $\mathbf{M}_k^{(t)} = \text{diag}(\mathbf{m}_k^{(t)})$.

A convenient feature of the RJM design is that due to the hierarchical structure of the model the objective function can be split into separate simple parts; specifically,

$$Q(\boldsymbol{\theta}, \boldsymbol{\tau}, \boldsymbol{\lambda}|\boldsymbol{\theta}^{(t)}, \boldsymbol{\tau}^{(t)}, \boldsymbol{\lambda}^{(t)}) = Q^Y(\boldsymbol{\theta}^Y, \boldsymbol{\lambda}|\boldsymbol{\theta}^{Y(t)}, \boldsymbol{\lambda}^{(t)}) + Q^X(\boldsymbol{\theta}^X|\boldsymbol{\theta}^{X(t)}) + Q^Z(\boldsymbol{\tau}|\boldsymbol{\tau}^{(t)}), \tag{15}$$

where $\boldsymbol{\theta}^Y = (\boldsymbol{\theta}_1^Y, \ldots, \boldsymbol{\theta}_K^Y)^T$ and $\boldsymbol{\theta}^X = (\boldsymbol{\theta}_1^X, \ldots, \boldsymbol{\theta}_K^X)^T$. Here, by $\boldsymbol{\lambda}$ we denote the vector of group penalty parameters of the regression component. Depending upon regularization approach, the elements of vector $\boldsymbol{\lambda}$ at iteration t are fixed in FLasso, free and under estimation in RLasso and absent in NJ; respectively having $\boldsymbol{\lambda}^{(t)} = (\hat{\lambda}_1^{(t^*)}, \ldots, \hat{\lambda}_K^{(t^*)})^T$ (where $t^* = \{0, 1\}$ with zero and one corresponding to the initial CV estimate and the re-estimated CV value; see FLasso in Section 2.2.2), $\boldsymbol{\lambda}^{(t)} = (\lambda_1^{(t)}, \ldots, \lambda_K^{(t)})^T$ and $\boldsymbol{\lambda}^{(t)} = \varnothing$.

Starting in reverse order from the right-hand side of (15) we have that

$$Q^Z(\boldsymbol{\tau}|\boldsymbol{\tau}^{(t)}) = \sum_{k=1}^K n_k^{(t)} \log \tau_k, \tag{16}$$

while the second component of the objective function is given by

$$Q^X(\boldsymbol{\theta}^X|\boldsymbol{\theta}^{X(t)}) = \frac{1}{2}\sum_{k=1}^K \left[\sum_{i=1}^n m_{ki}^{(t)} \left[\log|\boldsymbol{\Omega}_k| - (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Omega}_k (\mathbf{x}_i - \boldsymbol{\mu}_k) \right] - \tilde{\psi}\|\boldsymbol{\Omega}_k\|_1 \right], \tag{17}$$

where $\tilde{\psi} = \sqrt{2n \log p}/2$. The last component Q^Y in (15) depends on regularization method. We define two distinct functions Q^Y_{lasso} and Q^Y_{NJ}. For lasso we use the re-parametrization

$\chi_k = \alpha_k/\sigma_k$, $\boldsymbol{\phi}_k = \boldsymbol{\beta}_k/\sigma_k$ and $\rho_k = \sigma_k^{-1}$ (Städler et al., 2010), resulting in

$$Q^Y_{\text{lasso}}(\boldsymbol{\theta}^Y, \boldsymbol{\lambda}|\boldsymbol{\theta}^{Y(t)}, \boldsymbol{\lambda}^{(t)}) = \sum_{k=1}^{K}\Bigg[-\frac{(\rho_k\mathbf{y} - \chi_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\phi}_k)^T\mathbf{M}_k^{(t)}(\rho_k\mathbf{y} - \chi_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\phi}_k)}{2} - \lambda_k\|\boldsymbol{\phi}_k\|_1 + (n_k^{(t)} + p + 2)\log\rho_k + f(\lambda_k)\Bigg], \tag{18}$$

which is convex. Here $f^{(t)}(\lambda_k) = 0$ for FLasso and $f(\lambda_k) = c\sqrt{2K\log p/n}\log\lambda_k$ for RLasso. Under the NJ approach the corresponding objective is given by

$$Q^Y_{\text{NJ}}(\boldsymbol{\theta}^Y|\boldsymbol{\theta}^{Y(t)}) = -\frac{1}{2}\sum_{k=1}^{K}\Bigg[\frac{(\mathbf{y} - \alpha_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k)^T\mathbf{M}_k^{(t)}(\mathbf{y} - \alpha_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k)}{\sigma_k^2} + \boldsymbol{\beta}_k^T\mathbf{V}_k^{(t)}\boldsymbol{\beta}_k + (n_k^{(t)} + 2)\log\sigma_k^2\Bigg], \tag{19}$$

where $\mathbf{V}_k^{(t)} = \text{diag}\Big(1/\beta_{k1}^{2(t)}, \ldots, 1/\beta_{kp}^{2(t)}\Big)$. This matrix arises from the second underlying latent structure (the latent scale parameters) in the EM; details are provided in Appendix C. As we will see below, matrix $\mathbf{V}_k$ will in fact provide the final sparse estimate of $\boldsymbol{\beta}_k$, as some of its diagonal entries go to infinity during the EM; consequently, the diagonal entries of $\mathbf{U}_k = \mathbf{V}_k^{-1}$ that go to zero correspond to the coefficients that are set equal to zero.

The M-Step. From (16) we have that the group probabilities are updated as

$$\tau_k^{(t+1)} = n_k^{(t)}/n. \tag{20}$$

Concerning parameter block $\boldsymbol{\theta}_k^X$, from (17) we obtain the following updating equations

$$\boldsymbol{\mu}_k^{(t+1)} = \frac{\sum_{i=1}^n m_{ki}^{(t)}\mathbf{x}_i}{n_k^{(t)}}, \tag{21}$$

$$\boldsymbol{\Omega}_k^{(t+1)} = \arg\max_{\boldsymbol{\Omega}_k}\Big\{\log|\boldsymbol{\Omega}_k| - \text{tr}\big(\boldsymbol{\Omega}_k\mathbf{S}_k^{(t)}\big) - \zeta_k^{(t)}\|\boldsymbol{\Omega}_k\|_1\Big\}, \tag{22}$$

where in (22) we have that $\mathbf{S}_k^{(t)} = n_k^{-(t)}\sum_{i=1}^n m_{ki}^{(t)}(\mathbf{x}_i - \boldsymbol{\mu}_k^{(t+1)})(\mathbf{x}_i - \boldsymbol{\mu}_k^{(t+1)})^T$ and penalty given by $\zeta_k^{(t)} = \tilde{\psi}/n_k^{(t)} = \sqrt{2n\log p}/(2n_k^{(t)})$. For the lasso objective in (18) the FLasso group penalties are simply $\lambda_k^{(t+1)} = \hat{\lambda}_k^{(t^*)}$, while the RLasso update is

$$\lambda_k^{(t+1)} = \frac{cK^{1/2}}{\|\boldsymbol{\phi}_k^{(t)}\|_1}\sqrt{\frac{2\log p}{n}} = \frac{cK^{1/2}}{\|\boldsymbol{\beta}_k^{(t)}\|_1}\left(\sigma_k^{(t)}\sqrt{\frac{2\log p}{n}}\right). \tag{23}$$

Note that the RLasso update is a scaled version of the optimal universal penalty under orthonormal predictors (quantity inside the parenthesis in (23)) and that the scaling depends

on the sparsity of the coefficients and on c. For the components of $\boldsymbol{\theta}_k^Y$, the updates are as follows

$$\rho_k^{(t+1)} = \frac{\mathbf{y}^T\mathbf{M}_k^{(t)}(\chi_k^{(t)}\mathbf{1}_n + \mathbf{X}\boldsymbol{\phi}_k^{(t)}) + \sqrt{\left(\mathbf{y}^T\mathbf{M}_k^{(t)}(\chi_k^{(t)}\mathbf{1}_n + \mathbf{X}\boldsymbol{\phi}_k^{(t)})\right)^2 + 4\mathbf{y}^T\mathbf{M}_k^{(t)}\mathbf{y}(n_k^{(t)} + p + 2)}}{2\mathbf{y}^T\mathbf{M}_k^{(t)}\mathbf{y}}, \tag{24}$$

$$\chi_k^{(t+1)} = \frac{(\rho_k^{(t+1)}\mathbf{y} - \mathbf{X}\boldsymbol{\phi}_k^{(t)})^T\mathbf{m}_k^{(t)}}{n_k^{(t)}}, \tag{25}$$

$$\boldsymbol{\phi}_k^{(t+1)} = \arg\min_{\boldsymbol{\phi}_k} \frac{1}{2}\|\mathbf{M}_k^{1/2(t)}(\rho_k^{(t+1)}\mathbf{y} - \chi_k^{(t+1)}\mathbf{1}_n - \mathbf{X}\boldsymbol{\phi}_k)\|_2^2 + \lambda_k^{(t+1)}\|\boldsymbol{\phi}_k\|_1. \tag{26}$$

Finally, the EM updates of $\boldsymbol{\theta}_k^Y$ under the NJ prior are the following

$$\sigma_k^{2(t+1)} = \frac{\left(\mathbf{y} - \alpha_k^{(t)}\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k^{(t)}\right)^T\mathbf{M}_k^{(t)}\left(\mathbf{y} - \alpha_k^{(t)}\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k^{(t)}\right)}{n_k^{(t)} + 2}, \tag{27}$$

$$\alpha_k^{(t+1)} = \frac{(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}_k^{(t)})^T\mathbf{m}_k^{(t)}}{n_k^{(t)}}, \tag{28}$$

$$\boldsymbol{\beta}_k^{(t+1)} = \left(\mathbf{X}^T\mathbf{M}_k^{(t)}\mathbf{X} + \sigma_k^{2(t+1)}\mathbf{V}_k^{(t)}\right)^{-1}\mathbf{X}^T\mathbf{M}_k^{(t)}\left(\mathbf{y} - \alpha_k^{(t+1)}\mathbf{1}_n\right). \tag{29}$$

As remarked previously, in practice we work with $\mathbf{U}_k^{(t)} = \mathbf{V}_k^{-1(t)}$ Specifically, we have two available options for $\boldsymbol{\beta}_k$; the first is suited for the $n > p$ case and is given by

$$\boldsymbol{\beta}_k^{(t+1)} = \mathbf{U}_k^{\frac{1}{2}(t)}\left(\sigma_k^{2(t+1)}\mathbf{I}_p + \mathbf{U}_k^{\frac{1}{2}(t)}\mathbf{X}^T\mathbf{M}_k^{(t)}\mathbf{X}\mathbf{U}_k^{\frac{1}{2}(t)}\right)^{-1}\mathbf{U}_k^{\frac{1}{2}(t)}\mathbf{X}^T\mathbf{M}_k^{(t)}\left(\mathbf{y} - \alpha_k^{(t+1)}\mathbf{1}_n\right), \tag{30}$$

while the second, which is faster to compute when $n < p$, is given by

$$\boldsymbol{\beta}_k^{(t+1)} = \sigma_k^{-2(t+1)}\mathbf{U}_k^{(t)}\left[\mathbf{I}_p - \mathbf{X}^T\left(\sigma_k^{2(t+1)}\mathbf{M}_k^{-1(t)} + \mathbf{X}\mathbf{U}_k^{(t)}\mathbf{X}^T\right)^{-1}\mathbf{X}\mathbf{U}_k^{(t)}\right]\mathbf{X}^T\mathbf{M}_k^{(t)}\left(\mathbf{y} - \alpha_k^{(t+1)}\mathbf{1}_n\right) \tag{31}$$

Additional details on practical implementation appear in Appendix D.

3.2 Convergence guarantees

3.2.1 Preliminaries

The proposed EM algorithm is an expectation/conditional-maximization (ECM) as introduced by Meng and Rubin (1993). Let us recall some elements of their formalism. We call $\boldsymbol{\xi} \in \Xi$ the variable optimised in the M step. The corresponding optimised function is $Q(\boldsymbol{\xi}|\boldsymbol{\xi}^{(t)})$, where $\boldsymbol{\xi}^{(t)}$ is the value of the parameter after t ECM steps. Then, the exact M step is defined as

$$\boldsymbol{\xi}^{(t+1)} := \arg\max_{\boldsymbol{\xi}\in\Xi} Q(\boldsymbol{\xi}|\boldsymbol{\xi}^{(t)}). \tag{32}$$

When the optimisation in (32) is inconvenient, Meng and Rubin (1993) proposed to replace it by $S \in \mathbb{N}^*$ successive block-wise updates ("conditional maximization"; CM). Given S constraint functions $\{g_s(\boldsymbol{\xi})\}_{s=1}^S$, the CM step is decomposed into the S intermediary steps:

$$\boldsymbol{\xi}^{(t+s/S)} := \underset{\boldsymbol{\xi}\in\Xi; g_s(\boldsymbol{\xi})=g_s(\boldsymbol{\xi}^{(t+(s-1)/S)})}{\arg\max} Q(\boldsymbol{\xi}|\boldsymbol{\xi}^{(t)}), \tag{33}$$

for $s = 1, ..., S$. In their theorems 2 and 3, Meng and Rubin (1993) provide conditions under which all limit points of any ECM sequence are critical points of the observed likelihood. We propose a reformulation of their theorem 3, where we list explicitly all the required conditions.

Theorem 1 (Theorem 3 of Meng and Rubin (1993)) *With $r \in \mathbb{N}^*$, let Ξ be a subset of the Euclidean space $\mathbb{R}^r$. Let $\left\{\boldsymbol{\xi}^{(t)}\right\}_{t\in\mathbb{N}} \in \Xi^{\mathbb{N}}$ be an ECM sequence that has an observed log-likelihood called $L(\boldsymbol{\xi})$ as its objective function. The initial term $\boldsymbol{\xi}^{(0)}$ is such that $L(\boldsymbol{\xi}^{(0)}) > -\infty$. Let $Q(\boldsymbol{\xi}|\boldsymbol{\xi}^{(t)})$ be the corresponding expected complete likelihood that is conditionally maximised at each CM step, with constraints functions $\{g_s(\boldsymbol{\xi})\}_{s=1}^S$. Finally, call Ξ° the interior of Ξ, and assume that:*

- *Each conditional maximisation in the CM step* (33) *has a unique optimum*
- *$\forall s, g_s(\boldsymbol{\xi})$ is differentiable and the gradient $\nabla g_s(\boldsymbol{\xi}) \in \mathbb{R}^{r\times d_s}$ is of full rank on Ξ°*
- *$\bigcap_{s=1}^S \left\{\nabla g_s(\theta)u | u \in \mathbb{R}^{d_s}\right\} = \{0\}$*
- *The condition (6)-(10) of Wu (1983):*

  (6) $\Xi_{\boldsymbol{\xi}^{(0)}} := \left\{\boldsymbol{\xi} \in \Xi | L(\boldsymbol{\xi}) \geqslant L(\boldsymbol{\xi}^{(0)})\right\}$ is compact for any $L(\boldsymbol{\xi}^{(0)}) > -\infty$

  (7) L is continuous on Ξ and differentiable on Ξ°

  (9) $\Xi_{\boldsymbol{\xi}^{(0)}} \subseteq \Xi^\circ$

  (10) $Q(\boldsymbol{\xi}_1|\boldsymbol{\xi}_2)$ is continuous in both $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_2$.

Then all limit points of $\left\{\boldsymbol{\xi}^{(t)}\right\}_{t\in\mathbb{N}}$ are stationary points of the objective $L(\boldsymbol{\xi})$.

Note that condition (8) of Wu (1983):

(8) The sequence $\left\{L(\boldsymbol{\xi}^{(t)})\right\}_t$ is upper bounded for any $\boldsymbol{\xi}^{(0)} \in \Xi$

is verified as a direct consequence of (6) and (7) and is actually not an additional condition.

3.2.2 Main results

Here, we apply the convergence Theorem 1 to the proposed ECM algorithm for the RJM model. First we show that without modification, our ECM verifies almost all the hypotheses of Theorem 1. In particular the ones specific to the ECM procedure, as laid out by Meng and Rubin (1993). Then, we provide conditions on the ECM penalization under which the remaining, more restrictive, regularity hypotheses in Wu (1983) are also verified.

In our case, the optimization variable is $\boldsymbol{\xi} := (\boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\phi}, \boldsymbol{\chi}, \boldsymbol{\rho}, \boldsymbol{\lambda}, \boldsymbol{\tau}) \in \Xi$. Where the closure of the parameter set Ξ is $\overline{\Xi} = \mathbb{R}^{Kp} \times S_p(\mathbb{R})^+ \times \mathbb{R}^{Kp} \times \mathbb{R}^K \times {\mathbb{R}_+}^K \times {\mathbb{R}_+}^K \times$

$S_K \subset \mathbb{R}^{K(p^2+2p+3)}$, with $S_p(\mathbb{R})^+$ the cone of positive semi-definite matrices of size p and $S_K := \left\{\boldsymbol{\tau} \in [0,1]^K | \sum_k \tau_k = 1\right\}$. Its interior is $\Xi^\circ = \mathbb{R}^{Kp} \times S_p(\mathbb{R})^{++} \times \mathbb{R}^{Kp} \times \mathbb{R}^K \times {\mathbb{R}_+^*}^K \times {\mathbb{R}_+^*}^K \times S_K^\circ$, with $S_p(\mathbb{R})^{++}$ the open cone of positive definite matrices of size p and $S_K^\circ := \left\{\boldsymbol{\tau} \in]0,1[^K | \sum_k \tau_k = 1\right\}$. The ECM sequence takes its values in Ξ. With the proper priors, the parameters $\boldsymbol{\Omega}, \boldsymbol{\rho}, \boldsymbol{\lambda}$, and $\boldsymbol{\tau}$ cannot take values on the border of their respective sets during an ECM sequence. In such a scenario, the ECM sequence lives in Ξ° and we can simply consider that $\Xi = \Xi^\circ$, which helps with several of the hypotheses. To ensure this property, it is sufficient to set the regularization such that the objective $L(\boldsymbol{\xi})$ is infinite everywhere on the border. This objective is the penalized observed log-likelihood function:

$$\begin{aligned} L(\boldsymbol{\xi}) &= \sum_{i=1}^n \log \left(\sum_{k=1}^K p(y_i|\boldsymbol{\theta}_k^Y, \mathbf{x}_i, z_i = k) p(\mathbf{x}_i|\boldsymbol{\theta}_k^X, z_i = k)\tau_k \right) - \text{pen}(\boldsymbol{\xi}) \\ &= \sum_{i=1}^n \log \sum_{k=1}^K \exp \Bigg(-\frac{1}{2} \Bigg((y_i\rho_k - \chi_k - \mathbf{x}_i^T\boldsymbol{\phi}_k)^2 - 2\log\rho_k \\ &\qquad + \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2_{\boldsymbol{\Omega}_k} - \log|\boldsymbol{\Omega}_k| \\ &\qquad - 2\log\tau_k + (p+1)\log 2\pi + \frac{2}{n}\text{pen}(\boldsymbol{\xi}) \Bigg) \Bigg), \end{aligned} \tag{34}$$

where $\|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2_{\boldsymbol{\Omega}_k} := (\mathbf{x}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Omega}_k(\mathbf{x}_i - \boldsymbol{\mu}_k)$. The form $\text{pen}(\boldsymbol{\xi})$ for the penalty term is a generalization of the separable penalty $\sum_k \text{pen}(\boldsymbol{\theta}_k^{*X}) + \sum_k \text{pen}(\boldsymbol{\theta}_k^{*Y})$ proposed in Eq. (6). With the posterior weights $m_{ki}^{(t)} = \widehat{\Pr}(z_i = k|y_i, \mathbf{x}_i, \xi^{(t)})$ as defined in the E-step (14), we can define the function $Q(\boldsymbol{\xi}|\boldsymbol{\xi}^{(t)})$ to maximise in the CM step:

$$\begin{aligned} Q(\boldsymbol{\xi}|\boldsymbol{\xi}^{(t)}) = \sum_{i=1}^n \sum_{k=1}^K -\frac{1}{2} m_{ki}^{(t)} \Bigg(&(y_i\rho_k - \chi_k - \mathbf{x}_i^T\boldsymbol{\phi}_k)^2 - 2\log\rho_k \\ &+ \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2_{\boldsymbol{\Omega}_k} - \log|\boldsymbol{\Omega}_k| \\ &- 2\log\tau_k + (p+1)\log 2\pi + \frac{2}{n}\text{pen}(\boldsymbol{\xi}) \Bigg). \end{aligned} \tag{35}$$

The conditional maximization of this function is carried out in Eq. (20) to (26), for one specific version of the penalty $\text{pen}(\boldsymbol{\xi})$. As required by Theorem 1, each of these optmizations is uniquely defined. This property is penalty dependent. Hence, in general it is required to use penalties/priors on the parameters that lead to uni-modal posterior distributions.

On the other hand, the general structure of the conditional updates is independent of the penalty. Indeed, we propose a block-type update where each block is updated conditionally to all other being fixed. The order of the updates is: $\boldsymbol{\tau} \to \boldsymbol{\mu} \to \boldsymbol{\Omega} \to \boldsymbol{\lambda} \to \boldsymbol{\rho} \to \boldsymbol{\chi} \to \boldsymbol{\phi}$.

This correspond to constraint functions of the form:

$$\begin{aligned}
g_1(\boldsymbol{\xi}) &= \boldsymbol{\xi}\backslash\boldsymbol{\tau} := (\boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\chi}, \boldsymbol{\phi}) \,, \\
g_2(\boldsymbol{\xi}) &= \boldsymbol{\xi}\backslash\boldsymbol{\mu} := (\boldsymbol{\tau}, \boldsymbol{\Omega}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\chi}, \boldsymbol{\phi}) \,, \\
g_3(\boldsymbol{\xi}) &= \boldsymbol{\xi}\backslash\boldsymbol{\Omega} := (\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\chi}, \boldsymbol{\phi}) \,, \\
g_4(\boldsymbol{\xi}) &= \boldsymbol{\xi}\backslash\boldsymbol{\lambda} := (\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\rho}, \boldsymbol{\chi}, \boldsymbol{\phi}) \,, \\
g_5(\boldsymbol{\xi}) &= \boldsymbol{\xi}\backslash\boldsymbol{\rho} := (\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\lambda}, \boldsymbol{\chi}, \boldsymbol{\phi}) \,, \\
g_6(\boldsymbol{\xi}) &= \boldsymbol{\xi}\backslash\boldsymbol{\chi} := (\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\phi}) \,, \\
g_7(\boldsymbol{\xi}) &= \boldsymbol{\xi}\backslash\boldsymbol{\phi} := (\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\chi}) \,.
\end{aligned} \tag{36}$$

When the joint optimization in several consecutive blocks is possible, usually because they are separate in the objective, this approach can be simplified by "fusing" the corresponding blocks. For instance, in Eq (20) to (24), we perform a joint optimization in $\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Omega}, \boldsymbol{\lambda}, \boldsymbol{\rho}$ under the constraint that $\boldsymbol{\phi}, \boldsymbol{\chi}$ is fixed. Then, in Eq (25), an optimization on $\boldsymbol{\chi}$ with $\boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\phi}$ fixed. Finally, in Eq (26), an optimization on $\boldsymbol{\phi}$ with $\boldsymbol{\lambda}, \boldsymbol{\rho}, \boldsymbol{\chi}$ fixed. Note that in this case, the optimization in $\boldsymbol{\tau}, \boldsymbol{\mu}, \boldsymbol{\Omega}$ is a true M step and is independent on the other parameters. Hence, once this block has been updated in the first step, constraining it to remain fixed in the subsequent steps is unnecessary.

The functions $g_s(\boldsymbol{\xi})$ defined in (36) are obviously differentiable on Ξ with gradients of the form:

$$\nabla g_s(\boldsymbol{\xi}) = \begin{bmatrix} 0_{d_s \times d_1} & \dots & 0_{d_s \times d_{s-1}} & I_{d_s} & 0_{d_s \times d_{s+1}} & \dots & 0_{d_s \times d_S} \end{bmatrix}^T \in \mathbb{R}^{r \times d_s} \,,$$

which are of rank d_s (full rank), with $r := \sum_s d_s = K(p^2+2p+3)$. We can also see that there is no "overlap" between their non-zero components, which results in the desired property that $\bigcap_{s=1}^{S} \left\{ \nabla g_s(\theta) u | u \in \mathbb{R}^{d_s} \right\} = \{0\}$. As a consequence, as long as the penalty is chosen such that the posterior distribution in each parameter is unimodal, the algorithm verifies the three "ECM-specific" conditions for convergence introduced by Meng and Rubin (1993).

Among the basic conditions identified by Wu (1983), some are also systematically verified by our algorithm with little assumption on the penalty; namely:

(7) The model part of $L(\boldsymbol{\xi})$ is always continuous and differentiable in $\Xi = \Xi^\circ$. Hence, this property is guaranteed for $L(\boldsymbol{\xi})$ as long as the penalty term is also continuous and differentiable.

(9) $\Xi_{\boldsymbol{\xi}^{(0)}} := \left\{ \boldsymbol{\xi} \in \Xi | L(\boldsymbol{\xi}) \geqslant L(\boldsymbol{\xi}^{(0)}) \right\} \subseteq \Xi = \Xi^\circ$.

(10) $Q(\boldsymbol{\xi}_1 | \boldsymbol{\xi}_2)$ is continuous in ξ_1 for the same reason that $L(\boldsymbol{\xi})$ is continuous in ξ. The dependency of $Q(\boldsymbol{\xi}_1 | \boldsymbol{\xi}_2)$ on $\boldsymbol{\xi}_2$ is entirely through the terms $p(z_i = k | y_i, \mathbf{x}_i, \boldsymbol{\xi}_2) = p(y_i, \mathbf{x}_i, z_i = k | \boldsymbol{\xi}_2) / \sum_l p(y_i, \mathbf{x}_i, z_i = l | \boldsymbol{\xi}_2)$ which are continuous in $\boldsymbol{\xi}_2$ for the same reason that the likelihood is.

The final missing hypothesis is (6), the compacity of the level lines of the likelihood function. This property is much more restrictive and requires specific hypotheses on the regularization. The following theorem synthesizes every observation made so far and provides sufficient conditions to verify the final hypothesis.

Theorem 2 (Convergence of the ECM algorithm for RJMs) *Consider an ECM sequence* $\left\{\boldsymbol{\xi}^{(t)}\right\}_{t\in\mathbb{N}} \in \Xi^{\mathbb{N}}$ *with objective function the observed log-likelihood* $L(\boldsymbol{\xi})$ *of the RJM model* (34)*. The initial term* $\boldsymbol{\xi}^{(0)}$ *is such that* $L(\boldsymbol{\xi}^{(0)}) > -\infty$ *and the conditional maximization step of the expected complete likelihood* $Q(\boldsymbol{\xi}|\boldsymbol{\xi}^{(t)})$ *in Eq.* (35) *is conducted using the block-wise scheme defined by the constraint function* $g_s(\boldsymbol{\xi})$ *in Eq.* (36)*. Assume that the regularization term* $\text{pen}(\boldsymbol{\xi})$ *is continuous, differentiable and such that each of the block maximizations is unique (uni-modal posterior). Assume additionally that there exists a positive constant* $\delta > 0$ *such that*

$$\text{pen}(\boldsymbol{\xi}) \geqslant \delta \sum_{k=1}^{K} \left(\log \tau_k^{-1} + \|\mu_k\| + \|\boldsymbol{\Omega}_k\| + \log\left|\boldsymbol{\Omega}_k^{-1}\right| + f_\lambda(\lambda_k) + \rho_k + \log \rho_k^{-1} + |\chi_k| + \|\boldsymbol{\phi}_k\|\right), \tag{37}$$

where f_λ *is a lower bounded function on* $\mathbb{R}_+^*$ *such that* $f_\lambda(x) \underset{x\to 0}{\longrightarrow} +\infty$ *and* $f_\lambda(x) \underset{x\to+\infty}{\longrightarrow} +\infty$*. Then, Theorem 1 applies and all limit points of* $\left\{\boldsymbol{\xi}^{(t)}\right\}_{t\in\mathbb{N}}$ *are stationary points of the objective* $L(\boldsymbol{\xi})$*.*

Remark 3

- *The norms* $\|.\|$ *on each parameters in Eq.* (45) *(in Appendix E) are unspecified since all norms are equivalent in finite dimension.*
- *For* f_λ*, the lower bound on the penalty in* λ_k*, a function such as* $f_\lambda(x) = x - \log x$ *is suitable.*

Sketch of proof: The full details of the proof can be found in Appendix E, providing here a brief proof sketch. We prove that under the assumptions of Theorem 2, all the hypotheses of Theorem 1 are met, which yields the desired convergence result.

As previously discussed, all the hypotheses of Theorem 1, save for hypothesis (6), of Wu (1983) are organically verified within the RJM model. Provided that $L(\boldsymbol{\xi})$ is infinite on the border of Ξ, we can safely take $\Xi = \Xi^\circ$, and then the few required assumptions on the penalty are verified: it is continuous, differentiable and such that each of the block maximizations is unique. Hence, all the efforts of the proof are spent on proving that $L(\boldsymbol{\xi})$ is infinite on the border of Ξ and that hypothesis (6) is verified. This is done all at once; thanks to the control (45), we are able to define an increasing family Ξ_m of compacts of Ξ° such that the log-likelihood $L(\boldsymbol{\xi})$ on any point of $\overline{\Xi}\backslash\Xi_m$ is as low as desired with a well chosen compact Ξ_m. With this result, we have that (i) $L(\boldsymbol{\xi})$ is $-\infty$ outside of Ξ° and (ii) $\Xi_{\boldsymbol{\xi}^{(0)}} := \left\{\boldsymbol{\xi} \in \Xi | L(\boldsymbol{\xi}) \geqslant L(\boldsymbol{\xi}^{(0)})\right\}$ is compact for any $L(\boldsymbol{\xi}^{(0)}) > -\infty$ (hypothesis (6)), allowing us to conclude.

4. Prediction and cluster selection

An interesting feature of RJMs relates to prediction. Specifically, a new observation $\mathbf{x}^* \in \mathbb{R}^p$ can be allocated to a cluster via the quantities $\hat{\pi}_k^* \propto \hat{\tau}_k \varphi_p(\mathbf{x}^* | \hat{\boldsymbol{\mu}}_k, \hat{\boldsymbol{\Sigma}}_k)$, where $\hat{\tau}_k$, $\hat{\boldsymbol{\mu}}_k$ and $\hat{\boldsymbol{\Sigma}}_k$ are EM estimates for $k = 1, \ldots, K$. A simple prediction of y^* then follows via $\hat{y}^* = \hat{\alpha}_{\tilde{k}} + \mathbf{x}^{*T}\hat{\boldsymbol{\beta}}_{\tilde{k}}$, where $\tilde{k} = \arg\max_k \hat{\pi}_k$ and $\hat{\alpha}_{\tilde{k}}$ and $\hat{\boldsymbol{\beta}}_{\tilde{k}}$ are the corresponding EM estimates.

Although not our primary focus in this paper, it is interesting to briefly consider the idea of setting K based on predictive loss. In more detail, let $\mathcal{G}$ denote the set of the number of clusters under consideration, so that the cluster indicator under model $g \in \mathcal{G}$ is $k_g = 1, \ldots, K_g$. Under each model g we can obtain cluster allocations for a subset of held-out test data $\mathbf{y}^*$ and $\mathbf{X}^*$. Denote by $\mathbf{y}^*_{k_g}$ the $n^*_{k_g} \times 1$ test-response vector and by $\mathbf{X}^*_{k_g}$ the $n^*_{k_g} \times p$ test-feature matrix assigned to group $k_g = 1, \ldots, K_g$ conditional on g. Then, the solution for the "best" group-wise predictive model (in an ℓ_2 sense) is given by

$$\hat{g}^{\text{pred}} = \arg\min_{g \in \mathcal{G}} \left\{ \frac{1}{K_g} \sum_{k_g=1}^{K_g} \frac{\left\| \mathbf{y}^*_{k_g} - \hat{\alpha}_{k_g} \mathbf{1}_{n^*_{k_g}} - \mathbf{X}^{*T}_{k_g} \hat{\boldsymbol{\beta}}_{k_g} \right\|_2^2}{n^*_{k_g}} \right\}. \tag{38}$$

This effectively sets K to minimize predictive loss and connects in a way supervised and unsupervised learning, providing a simple and quite natural way of determining the number of clusters based on a "guided" search that aims to optimize prediction of $\mathbf{y}$.

Of course, standard approaches for inferring the number of clusters, such as information criteria, can be used within the RJM framework. For instance, using BIC (Schwarz, 1978) in our case translates in selecting the number of clusters as

$$\hat{g}^{\text{BIC}} = \arg\max_{g \in \mathcal{G}} \Big\{ 2 \log p(\mathbf{y}, \mathbf{X} | \boldsymbol{\theta}_g, \boldsymbol{\tau}_g) - \log(n) \nu_g \Big\}, \tag{39}$$

where ν_g is the number of elements in $\boldsymbol{\theta}_g$ that are not set equal to zero. Similarly, for AIC (Akaike, 1974) we have

$$\hat{g}^{\text{AIC}} = \arg\max_{g \in \mathcal{G}} \Big\{ 2 \log p(\mathbf{y}, \mathbf{X} | \boldsymbol{\theta}_g, \boldsymbol{\tau}_g) - 2 \nu_g \Big\}. \tag{40}$$

5. Empirical examples

In this Section we present results from simulation experiments, starting with a small-scale simulation in Section 5.1 which allows us to evaluate and visualize easily the various learning aspects of RJMs. In Section 5.2 we use data from The Cancer Genome Atlas in semi-synthetic examples of much larger scale, providing detailed comparisons with baseline and various oracle-type approaches. All simulations are based on data-generating mechanisms which are multivariate generalizations of three elementary problems which are depicted in Figure 3, the purpose of which is to facilitate understanding of more complex multivariate problems as the ones in Section 5.2 via illustration of simpler univariate analogues. Finally, in Section 5.3 we show results using fully empirical data.

5.1 Small-scale simulations

Set-up. We consider two groups ($K = 2$) with total $n = 100$ and balanced groups, i.e. $n_k = 50$ for $k \in \{1, 2\}$. The number of predictors is $p = 10$, where in each group only the first predictor ($\mathbf{x}_{k1}$) has a non-zero coefficient, i.e. only $\beta_{k1} \neq 0$ for $k \in \{1, 2\}$. The covariates are generated as $\mathbf{X}_k \sim \text{N}_p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$, with $\boldsymbol{\mu}_1 = (0, \ldots, 0)^T$ and $\boldsymbol{\mu}_2 = (1, \ldots, 1)^T$ being of dimensionality $p \times 1$. For the covariances we consider two scenarios: an *uncorrelated-scenario* with diagonal covariances of the form $\boldsymbol{\Sigma}_1 = \boldsymbol{\Sigma}_2 = \mathbf{I}_p$ and a *correlated-scenario* with

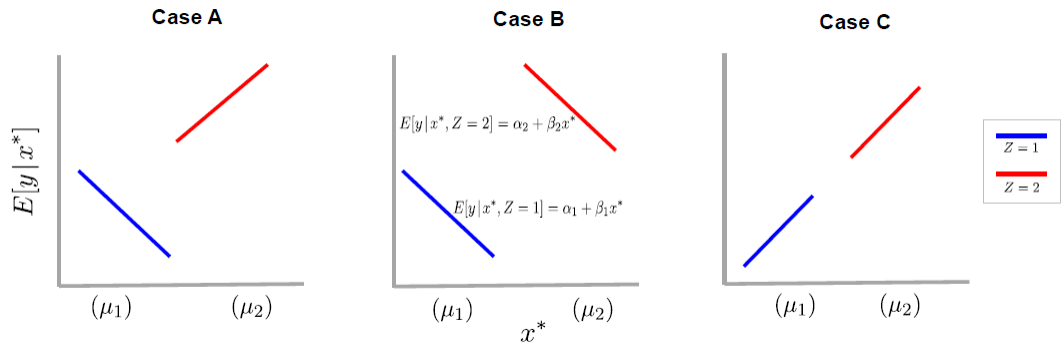

Figure 3: Some interesting cases of group structure. Univariate analogues of problems considered in the empirical examples (all of which are multivariate) in order to illustrate the key ideas. Shown are two latent groups with some separation between the group-wise means of a single feature x^*. Three specific cases, that differ with respect to the regression model linking y and x^*, are shown: equal-intercept/unequal-slope (Case A); unequal-intercept/equal-slope (Case B); and equal intercept and slope (Case C).

non-diagonal covariances, where each variable $\mathbf{x}_{kj}$, for $j = 2, \ldots, p$, is again Gaussian noise, but the signal variable $\mathbf{x}_{k1}$ is generated as $\mathbf{x}_{k1} \sim \mathrm{N}_{n_k}\big(1.5\mathbf{x}_{k3} + 0.5\mathbf{x}_{k5} - 0.7\mathbf{x}_{k7},\ 0.5\mathbf{I}_{n_k}\big)$. The response of each group is generated as $\mathbf{y}_k \sim \mathrm{N}_{n_k}\big(\mathbf{1}_{n_k}\alpha_k + \mathbf{x}_{k1}\beta_{k1},\ \sigma_k^2\mathbf{I}_{n_k}\big)$. Specification of the slopes and intercepts is based on the three cases of Figure 3; see Table 3 (Appendix F). Finally, the error variance σ_k^2 of each group is set to fix signal strength in a label-oracle sense, namely so that the correlation between test data (for test group sample sizes of 250) and predictions from a lasso model is approximately 0.8 when group labels are known. The results that follow are from 50 repetitions of each simulation. We focus on regression signal detection, estimation of coefficients and group assignment performance.

Signal detection and estimation. The variable inclusion frequencies over 50 repetitions of the simulations for the uncorrelated scenario are presented in Figure 4. RJM-NJ performs better overall as it detects influential effects almost all of the times, while the inclusion rates of non-influential effects are much lower than 50%. RJM-FLasso is effective in detecting the signals but produces much denser solutions. RJM-RLasso solutions are sparser in comparison to FLasso; we note, however, that RLasso tends to over-shrink the coefficients of the influential predictors as well. The inclusion frequencies for the correlated scenario are similar (Appendix F, Figure 12). Violin plots of slope estimates are presented in Appendix F (Figure 13, uncorrelated scenario; Figure 14, correlated scenario), while the corresponding plots for intercepts are presented in Figures 15 and 16. The NJ estimates are overall more accurate. In all comparisons we include results from mixtures of experts obtained from R package `MoEClust` (Murphy and Murphy, 2020), which performs simultaneous selection for experts, gates and covariance structures using forward search model selection based on BIC. In terms of variable selection MoE performs exceptionally well under case A and yields similar results to RJM-NJ under cases B and C (see Figures 4 and 12). As for estimation, MoE leads to overall accurate slope and intercept estimates in case A, however,

the corresponding estimates in cases B and C have a higher variance in comparison to RJM estimates (see Figures 13 to 16, Appendix F).

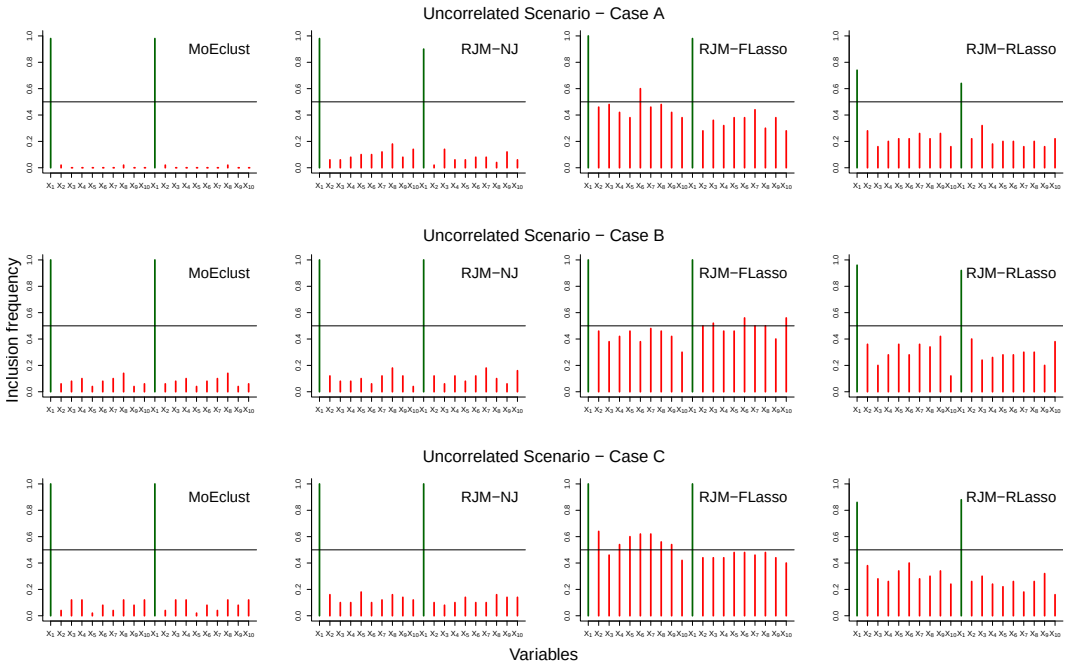

Figure 4: First simulation, uncorrelated scenario. Variable inclusion frequencies (over 50 repetitions) for signal variables (in green) and non-signal variables (in red) for regression cases A, B and C. Horizontal black lines correspond to a frequency of 0.5.

Latent group assignment. A natural question is whether including the regression part of the model within a unified framework provides any gains with respect to simply clustering the X matrix. In practice, between-group differences in means may be subtle. Hence, we consider in particular performance as the magnitude of the mean difference varies; i.e., we set $\boldsymbol{\mu}_1 = \mathbf{0}_p$ and $\boldsymbol{\mu}_2 = \mathbf{0}_p + d\mathbf{1}_p$ with $d = h \cdot U$, where h defines a grid ranging from 0.1 to 1 with a step size of 0.05 and $U \in \{-1, +1\}$ is a uniformly random sign. Hence, $|d|$ is a measure of the strength of the mean signal. We compare to k-means, hierarchical clustering, GMMs and MoEs as implemented in R using the default options of `kmeans`, `hclust`, `mclust` (Scrucca et al., 2016) and `MoEClust` respectively; for the latter two using the BIC-optimal model. In addition, for the clustering approaches we use as input: (i) only $\mathbf{X}$ and (ii) $\mathbf{X}$ together with $\mathbf{y}$ stacked in one data matrix. For these simulations we use 20 repetitions.

One standard-error plots of adjusted Rand index averages as functions of $|d|$ are shown in Figure 5. As seen, in case A, RJMs generally outperform all methods except of MoE; the latter performs better for lower values of $|d|$, while RJMs perform better for higher values. In cases B and C (where RJM is over-parameterized) our methods remain competitive in the uncorrelated scenario, while lead to better overall results in the correlated scenario. On the other hand, MoE is not effective under cases B and C, which as argued in Section 1 (recall Figure 2) is to be expected when there are no differences in regression coefficients.

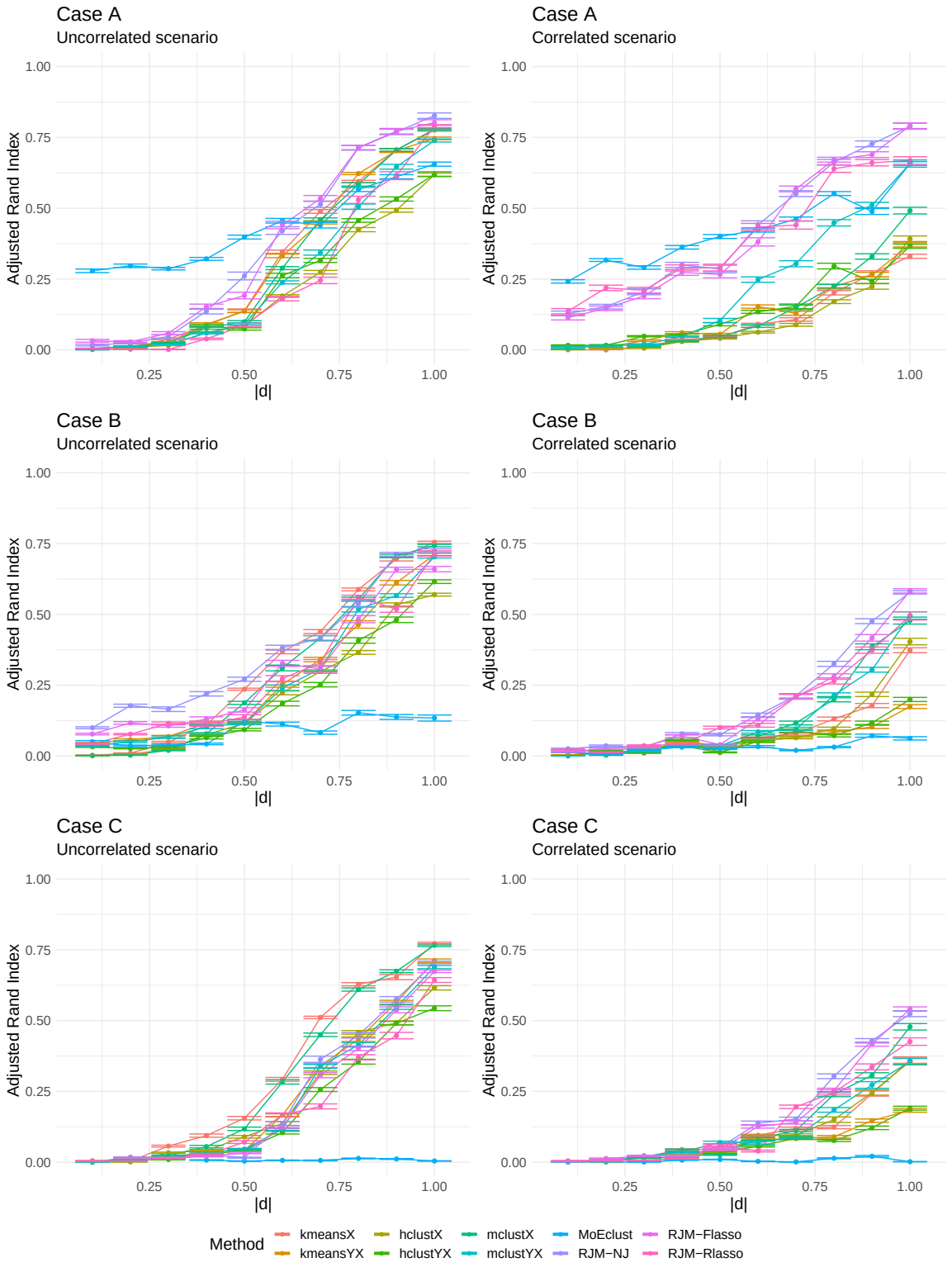

Figure 5: First simulation, cases A (top), B (middle) and C (bottom). One standard-error plots of adjusted Rand index averages (from 20 repetitions) vs absolute distance ($|d|$) of the group-wise covariate means under the uncorrelated scenario (left) and correlated scenario (right).

5.2 Semi-synthetic simulations based on real cancer data

The simulations presented below are based on data from the The Cancer Genome Atlas (TCGA, `https://cancergenome.nih.gov`). The rationale is to anchor the simulation in real covariance structures. Specifically, we use data previously used in Taschler et al. (2019), consisting of gene expression values from four cancer types; breast (BRCA), kidney renal clear cell (KIRC), lung adenocarcinoma (LUAD) and thyroid (THCA). Our strategy is to treat the cancer type as hidden: this allows us to test our approaches in the context of differential covariance structure as seen in a real group-structured problem whilst having access to true gold-standard labels.

Table 1: Second simulation. Intercept values and slope-generating mechanisms for the two groups under the three cases illustrated in Figure 3. [$\text{TN}(\mu, \sigma^2, l, u)$ denotes a truncated normal distribution, where $\mu \in \mathbb{R}$, $\sigma^2 > 0$ and l, u are the respective lower and upper truncation bounds, while $\text{mTN}(\mu, \sigma^2, a, b)$ denotes the specific mixture of $\text{TN}(\mu, \sigma^2, -\infty, a)$ and $\text{TN}(\mu, \sigma^2, b, \infty)$ with $a < b$ and mixing parameter equal to 0.5; i.e., a truncated normal with support everywhere except in (a, b)].

Case	**Group**	**Intercept**	**Slopes**	
			Common locations	**Disjoint locations**
A	1	$\alpha_1 = \alpha_2 = 0$	$\beta_1^* \sim \text{TN}(0, \tilde{\sigma}^2, -\infty, -0.1)$	$\beta_1^* \sim \text{mTN}(0, \tilde{\sigma}^2, -0.1, 0.1)$
	2		$\beta_2^* \sim \text{TN}(0, \tilde{\sigma}^2, 0.1, \infty)$	$\beta_2^* \sim \text{mTN}(0, \tilde{\sigma}^2, -0.1, 0.1)$
B	1	$\alpha_1 = 0$	$\beta_1^* = \beta_2^* \sim \text{mTN}(0, \tilde{\sigma}^2, -0.1, 0.1)$	$\beta_1^* \sim \text{mTN}(0, \tilde{\sigma}^2, -0.1, 0.1)$
	2	$\alpha_2 = 1$		$\beta_2^* \sim \text{mTN}(0, \tilde{\sigma}^2, -0.1, 0.1)$
C	1	$\alpha_1 = \alpha_2 = 0$	$\beta_1^* = \beta_2^* \sim \text{mTN}(0, \tilde{\sigma}^2, -0.1, 0.1)$	$\beta_1^* \sim \text{mTN}(0, \tilde{\sigma}^2, -0.1, 0.1)$
	2			$\beta_2^* \sim \text{mTN}(0, \tilde{\sigma}^2, -0.1, 0.1)$

Set-up. In these experiments we use covariates from two cancer types; namely, the BRCA and KIRC groups. For all simulations we use $n = 250$, balanced group sample sizes, i.e. $n_k = 125$ for $k = 1, 2$, and varying dimensionality for the features; namely, i) $p = 100$ ($n > p$ problem), ii) $p = 250$ ($n = p$ problem) and iii) $p = 500$ ($n < p$ problem). We consider sparse problems where the percentage of non-zero coefficients ($\boldsymbol{\beta}^*$) is $s = 4\%$ and the setting in which some of the non-zero coefficients are at common locations and others are at disjoint locations across the two groups (placing half the non-zero coefficients at common locations). Specification of the common-location $\boldsymbol{\beta}_k^*$'s will determine the three general cases depicted initially in Figure 3. To rule out very small coefficients, we draw from a truncated normal distribution, with support excluding the interval $(-0.1, 0.1)$. Group specific intercept values and slope-generating mechanisms (based on Figure 3), are summarized in Table 1. Given the matrices $\mathbf{X}_k$, the intercepts α_k and the sparse vectors $\boldsymbol{\beta}_k$ the response is generated as $\mathbf{y}_k \sim \text{N}_{n_k}(\boldsymbol{m}_k,\ \mathbf{I}_{n_k}\sigma_y^2)$, where $\boldsymbol{m}_k = \mathbf{I}_{n_k}\alpha_k + \mathbf{X}_k\boldsymbol{\beta}_k$ for $k = 1, 2$ and $\sigma_y^2 = 1$. The scale parameter $\tilde{\sigma}^2$ in Table 1 is tuned so that the overall signal-to-noise under each case is approximately equal to three; i.e. $\text{Var}(\boldsymbol{m})/\sigma_y^2 \approx 3$ and $\boldsymbol{m} = (\boldsymbol{m}_1, \boldsymbol{m}_2)^T$.

Performance is evaluated as a function of the absolute distance $|d|$ between the group-wise feature means $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$. We initially normalize the features, so that $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \mathbf{0}$, and consider again the case where each element of $\boldsymbol{\mu}_2$ is shifted by $d = h \cdot u$, where $h \in$

$\{0.1, 0.2, \ldots, 0.8, 0.9\}$ and u is a random uniform sign. Each simulation is repeated 20 times using random subsamples of features from the original data. Here we present results for the $n < p$ setting ($p = 500$); results for $p \in \{100, 250\}$ can be found in Appendix G. For the regression questions addressed below our aim is to compare RJM with the "clustering-then-regression" approach. Obviously, a range of regression methods could be used in the second step. As the simulations are sparse and linear by design, to ensure that the simple "clustering-then-regression" approach is not disadvantaged we use lasso in the second step.

Group assignment. We compare to the same methods considered in Section 5.1, except of `MoEClust` as the dimensionalities are too large for a forward model search for optimal selection of expert and gating functions. Figure 6 presents error plots of adjusted Rand index averages. In general, we observe a phase-transition type of behaviour as all methods improve as $|d|$ increases. However, the transition is faster with RJM which outperforms the other methods and stabilizes relatively quickly to correct assignment. For $p = 500$ lasso-based RJM outperforms the NJ variant for cases B and C, however, for $p = \{100, 250\}$ all RJM methods perform equally well more or less; see Figures 17 and 18 in Appendix G.

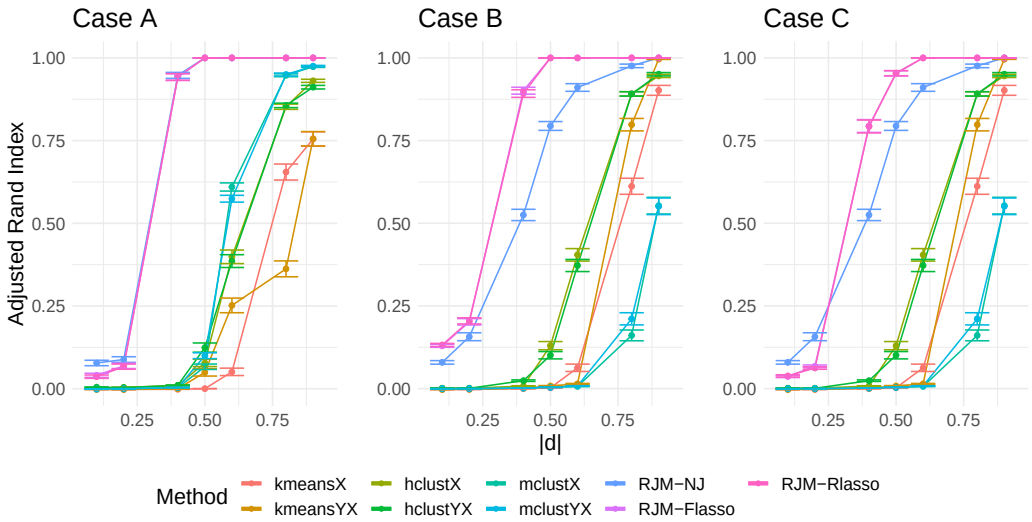

Figure 6: Second simulation, $p = 500$, group assignment. Average adjusted Rand Index as a function of the absolute distance ($|d|$) of the group-wise covariate means, for cases A (left), B (center) and C (right). [Error bars indicate standard errors from 20 repetitions.]

Variable selection. For this comparison we initially set a benchmark model called the *label-oracle-lasso*, under which the true group labels are assumed known and we fit separate lasso regressions via `glmnet` (Friedman et al., 2008b). We also consider a *cluster-lasso* model which involves separate regressions based on estimated group labels. This approach involves an initial clustering step: we give an advantage to the cluster-lasso by using, for each dimension considered, the clustering approach that performs best (`hclust` for $p = 100$ and `Mclust` for $p \in \{250, 500\}$). Naturally, the cluster-lasso will be equivalent to the oracle-lasso when group assignment is perfect.

We summarize results via the area under the ROC curve (AUC) based on the ranking of the absolute values of the coefficients. In particular, we consider the difference between

the AUC from oracle-lasso and AUC from competing methods (cluster-lasso and RJM approaches). One standard-error plots for $p = 500$ are presented in Figure 7. As expected cluster-lasso yields smaller selection loss (approaching oracle-lasso) as the separation of group-wise means increases, but so do the RJM methods. RJM-FLasso is overall better and even seems to result in slightly improved selection in comparison to the oracle-lasso as $|d|$ increases, possibly due to the fact that RJM uses weighted estimation based on the entire sample. Importantly, RJM methods overall outperform the common cluster-lasso approach in low and/or medium magnitude regions of $|d|$. These results illustrate the nontrivial gains possible from a unified treatment of the various aspects of the model vs. the simple approach of clustering followed by sparse regression. For the simulations with $p = 100$ and $p = 250$ see Figures 19 and 20 (Appendix G), respectively.

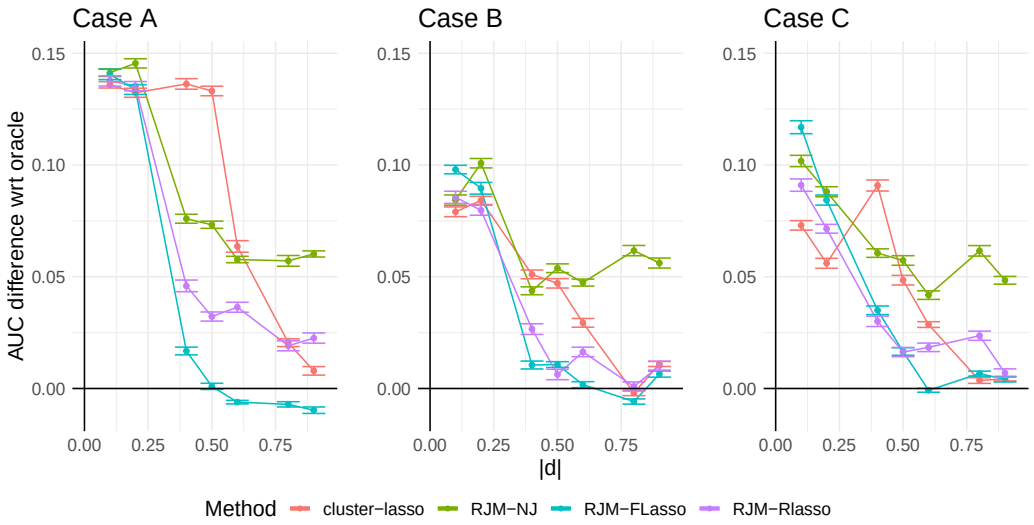

Figure 7: Second simulation, $p = 500$, variable selection. AUC loss from oracle-lasso as a function of the absolute distance ($|d|$) of the group-wise covariate means, for cases A (left), B (center) and C (right). [Error bars indicate standard errors from 20 repetitions.]

Estimation. Comparisons are made again with respect to the label-oracle-lasso; this time we consider the increase in root mean squared error (RMSE) resulting from the fact that the group labels are unknown. Here we also consider the *pooled-lasso*; a "naive" model which does not take into account group structure. This allows us to investigate the effect of ignoring group structure under each case in Table 1; this is of particular interest in Case A where the common-location coefficients have opposite signs.

Results for the $p = 500$ are summarized in Figure 8. We use standardized coefficients for the calculation of RMSE in order to have a common scale across simulations and cases. As expected, under Case A the pooled-lasso model performs poorly, while RJM methods provide overall better estimates than cluster-lasso. Under cases B and C, cluster-lasso and RJM which are over-parameterized (common-location effects are equal) perform more or less the same and are in general comparable to the pooled-lasso which is under-parameterized (due to the disjoint-location effects). The $p = 100$ and $p = 250$ cases are shown in Appendix G (Figures 21 and 22); results are in general similar with the difference that cluster-lasso

performs better when $|d| \approx 1$ and RJM-RLasso performs overall worse. Overall, our illustrations suggest that the RJM-FLasso is the most stable method, followed by RJM-NJ.

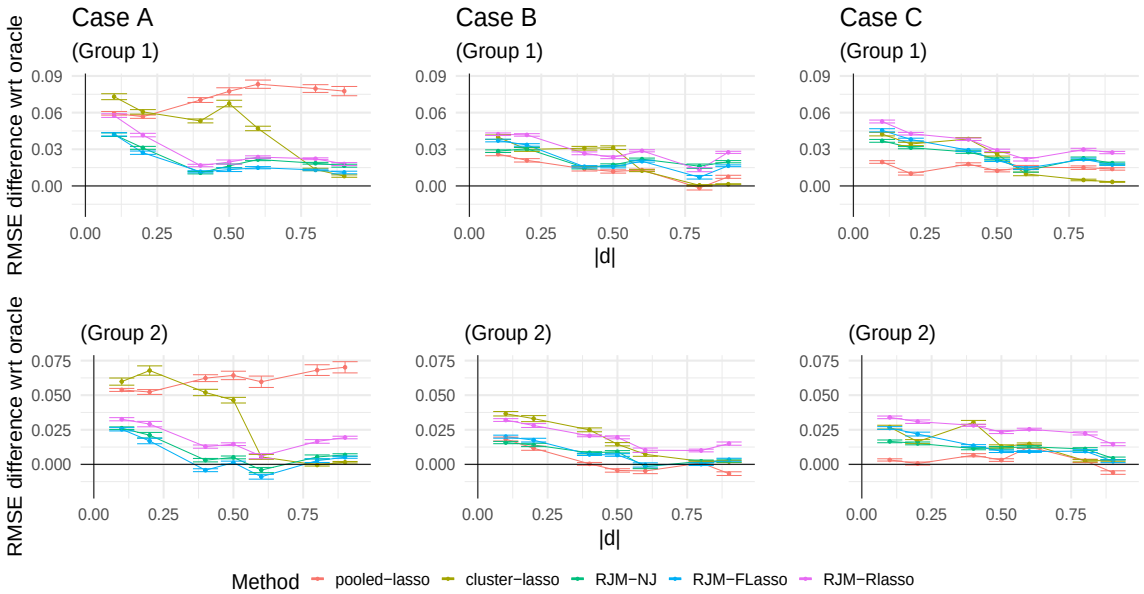

Figure 8: Second simulation, $p = 500$, estimation. Increase in RMSE relative to the oracle-lasso as a function of the absolute distance ($|d|$) of the group-wise covariate means, under group one (top) and group two (bottom), and for cases A (left), B (center) and C (right). [Error bars indicate standard errors from 20 repetitions.]

Selection of number of clusters. In Appendix H we further consider the case where the number of clusters is not known *a-priori* in simulation experiments which take into account all four cancer types. Results on cluster selection using the predictive approach described in Section 4 are shown in Figure 23.

5.3 Real cancer data example

In this Section we consider a fully non-synthetic example, where both features and response are real empirical data. The general strategy is as follows. We use the TCGA data as introduced above, with gene expression levels treated as features. Specifically, we include all four cancer types used in the previous section, selecting via stratified random sampling $n = 250$ samples in total. Stratified sampling ensures that the cancer-type proportions are preserved; specifically the dataset under consideration consists of 102 BRCA, 51 KIRC, 49 LUAD and 48 THCA observations (abbreviations as previously introduced). A total of $p = 100$ gene expression levels (selected at random from all genes) are used in these experiments. As before, in the applications that follow the true labels (i.e., the cancer type indicator) are treated as latent and hence not used in analysis, but only to evaluate performance. As responses, we use one of the $p = 100$ gene expression levels, with the remaining forming the feature set. This procedure has the advantage of allowing us to consider many different responses (genes) whilst entirely eschewing synthetic data generation. We first show a

illustrative example using one particular gene as response and then show results from all responses considered.

Illustrative analysis for a particular response gene (NAPSA). We consider the gene NAPSA (napsin A aspartic peptidase) (gene ID 9476) to illustrate the set-up. For this illustrative analysis we assume that the cancer types are given so that it is known *a priori* that there are four classes. This particular gene is used to illustrate the approach as it is informative with respect to hidden group structure, as shown in Figure 9 (left panel), but not to the extent of fully revealing the class structure. Heatmaps of the sample covariance matrices of the remaining 99 genes under each cancer type are presented in Figure 9 (panels in the right); these generally indicate slight differences in covariance structures.

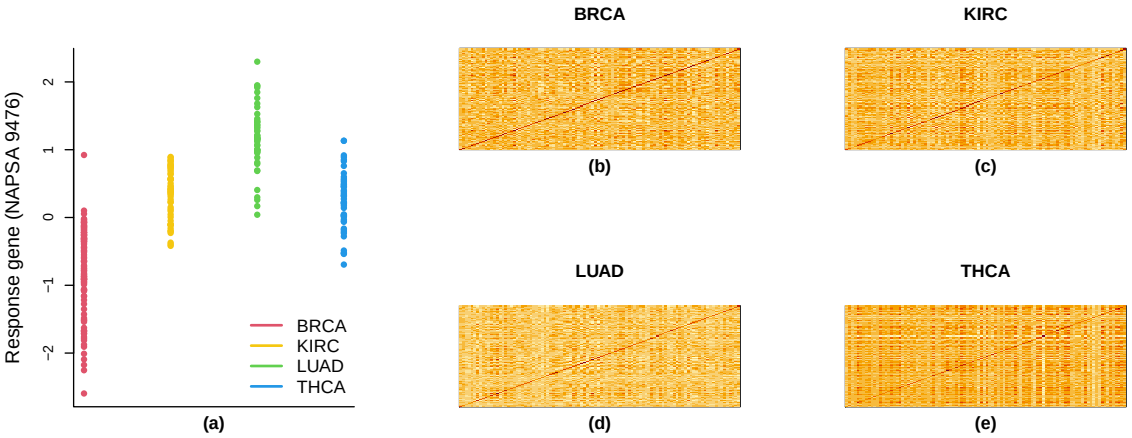

Figure 9: Real data example, data visualization for single, illustrative response. Plot (a) of response gene NAPSA in the four cancer types, and heatmaps of feature covariances in (b) BRCA, (c) KIRC, (d) LUAD and (e) THCA cancer types.

For evaluation of clustering performance we compare to the same methods as in Section 5.1, including again MoE as implemented in package `MoEclust`; however, for computational convenience, we use the option of a classification-EM algorithm, which is a faster but generally sub-optimal algorithm (although we note that initial tests suggested a gain in clustering performance for this dataset). We further consider k-medoids, fuzzy c-means (implemented via `par` and `fanny` in package `cluster`, respectively) and clustered support vector machines (clustSVM, package `SwarmSVM`; Gu and Han, 2013). Finally, for the purely cluster-oriented approaches (k-means/medoids, fuzzy c-means, hclust and mclust) we use as input the concatenated matrix containing the response and the predictor genes. Table 2 shows the resulting adjusted Rand index under each method (for k-means and clustSVM, which are highly sensitive to initialization, the values are averages from 100 runs). As seen, RJMs clearly outperform the other approaches.

Figure 10 shows the resulting regression coefficient estimates (396 in total given the four cancer types), from the three RJM variants, ranked in absolute value from highest to lowest (non-zero values in green; zero values in red). Consistent with the results presented in

Table 2: Real data application, clustering performance. Adjusted Rand index values for the ten methods under consideration using gene NAPSA as response variable.

Methods and clustering performance					
Method	k-means	k-medoids	fuzzy c-means	hclust	clustSVM
Adj. Rand Index	0.43	0.44	0.59	0.51	0.42
Method	mclust	MoEclust	RJM-NJ	RJM-FLasso	RJM-RLasso
Adj. Rand Index	0.29	0.55	0.72	0.68	0.75

Section 5.1, we observe again that RJM-NJ results in the most parsimonious model (fewer than 100 predictors), followed by RJM-RLasso (around 100 predictors), while RJM-FLasso includes the most predictors (more than 100). We also observe that the lasso variants tend to shrink the coefficients of influential predictors more than RJM-NJ; this is also generally anticipated as the NJ prior has heavier tails in comparison to the Bayesian lasso prior. Finally, the forward search of MoEclust included a few predictors in the gating networks, but resulted in entirely sparse expert networks as all regression coefficients were set to zero.

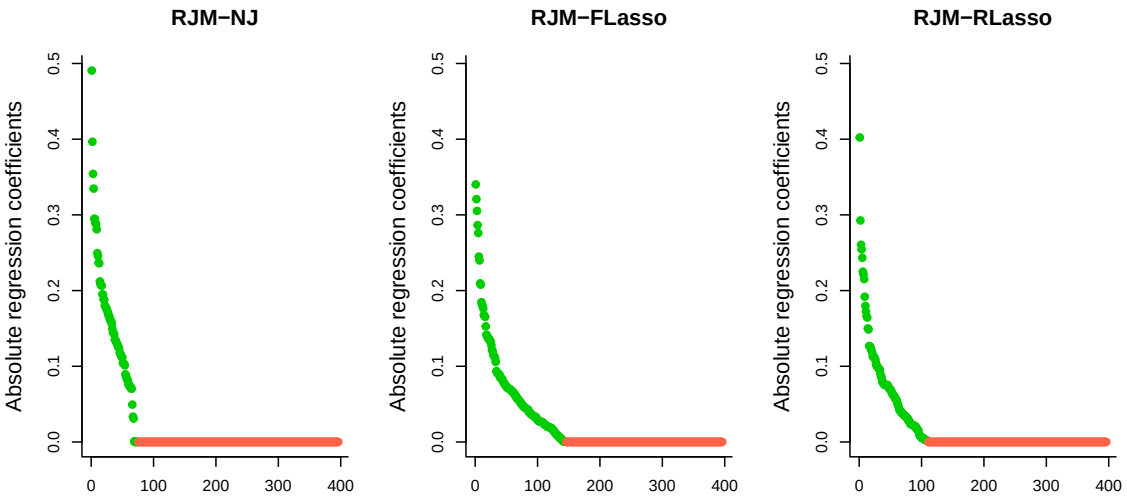

Figure 10: Real data application, regression performance. Absolute values of RJM regression coefficients ranked in decreasing order, using gene NAPSA as response variable. Green points indicate non-zero coefficients; red points, coefficients that are set equal to zero.

Performance over all responses. Above we considered a specific gene to illustrate the key ideas; here we show results from all responses. That is, we consider in turn each of the genes as response, treating all others as features. Thus, there are 100 problems considered in total (each with the same four latent subgroups). In this case, the input for the purely clustering methods is the data matrix of the predictor variables.

Violin plots of the resulting adjusted Rand index values from the ten methods are presented in Figure 11. These results, spanning one hundred different responses, support the results seen above, as the three RJM variants consistently perform relatively well over most of the responses and the results are broadly in line with some of the results in Sections 5.1 and 5.2. In Appendix I, we further consider BIC-based model selection; as shown there, RJMs select more frequently the correct number of groups in comparison to GMMs and MoEs.

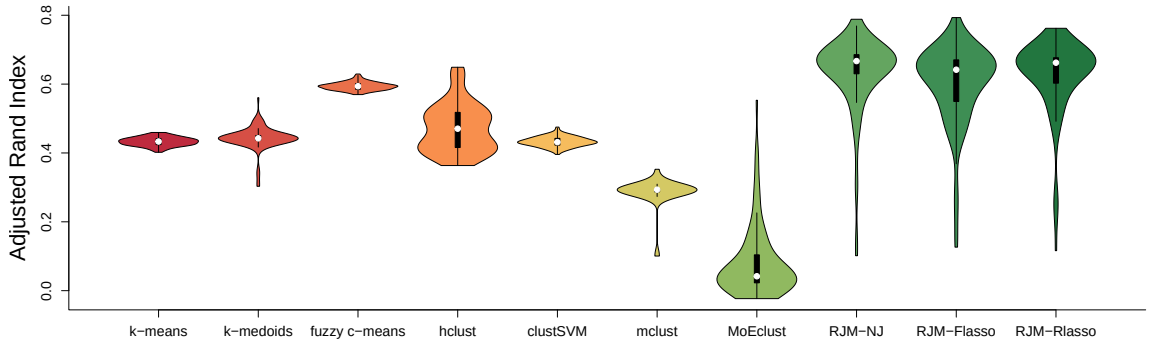

Figure 11: Real data application, clustering performance. Violin plots of adjusted Rand index values from TCGA-based experiments spanning a hundred different responses (see text for details).

5.4 Summary

Broadly speaking the RJM variants performed similarly to one another in terms of clustering/group assignment. In several situations they outperformed the other methods compared with, while, at the same time, they tended to remain competitive across the range of scenarios tested. Also, RJM can improve selection for the number of groups when this is not known at the outset. There were some differences between the RJM variants with respect to regression modelling. The NJ approach tended to perform well, in terms of variable selection and estimation, in sparse settings characterized by moderate to strong signals, while the lasso approaches yielded relatively denser models overall.

6. Discussion

We introduced a class of regularized mixture models that jointly deal with sparse covariance structure and sparse regression in the context of latent groups. We showed that principled joint modeling of these two aspects leads to gains with respect to simpler decoupled or pooled strategies and that exploiting established ℓ_1-penalized tools and related Bayesian approaches leads to practically applicable solutions. The RJM methods presented in this paper are implemented as an R package `regjmix`, available at `https://github.com/k-perrakis/regjmix`. Future research directions include extensions to generalized linear models and

mixed models. Below we discuss some additional aspects and point to specific directions for future work.

Distribution shifts and shift-robust learning. By accounting for data heterogeneity, RJMs help to guard against (potentially severe) confounding of multivariate regression models by hidden group structure. This has interesting connections to distribution shifts in machine learning and shift-robust learning; see e.g. Recht et al. (2018); Heinze-Deml and Meinshausen (2021). In particular, we think RJM would be a useful tool for shift-robust learning, since it could be used to block paired data (X, Y) into distributionally non-identical groups which could in turn be used to train and test predictors in a shift-robust fashion, facilitating shift-robust learning under unknown distributional regimes.

Choice of regularization. In this work we used the graphical lasso approach for covariance estimation, mainly motivated by certain biomedical applications where network models are of interest. However, the general RJM strategy could be used with other kinds of multivariate models (e.g. factor models). For the regression coefficients we considered: (i) the Bayesian lasso prior under two strategies (FLasso/RLasso) and (ii) the NJ prior. For practitioners seeking the closest analogue to the popular lasso approach based on cross-validation in the non-latent regression setting, we recommend FLasso. When it is preferable to use a shrinkage prior with heavier tails we recommend using NJ, which can be very effective in detecting sparsity patterns without over-shrinking large coefficients. In general, our main goal was to explore some of the available regularization options, but we note that the RJM model is fairly modular in the sense that other methods from the penalized likelihood or the Bayesian literatures – recent reviews provided by Desboulets (2018) and van Erp et al. (2019), respectively – can be used within the same framework. Of course, specifics will depend upon approach; for instance, the elastic-net (Zou and Hastie, 2005) and the adaptive lasso (Zou, 2006) are fairly easy to incorporate at present, while other methods such as the horseshoe estimator (Carvalho et al., 2010) require further investigation.

High-dimensional issues. RJM remains effective when $p > n$, but a general issue when jointly modeling (Y, X) is that for relatively large p cluster allocation will be mainly guided by X. In the empirical examples RJM outperformed `mclust` (recall that without regularization RJM is equivalent to a GMM); to provide some intuition about that let us consider the two regularization steps. The first on the covariance matrix of X can be viewed as p lasso regressions (Meinshausen and Bühlmann, 2006) that essentially discard non-influential relationships among features. The second discards non-influential predictor effects on the response. Overall this sparsification may be viewed as a dimensionality reduction, which mitigates over-emphasis on X. One idea for handling this issue as p grows larger is to consider explicit weighting of the effect of X, e.g. by replacing the multivariate normal in (3) with a density of the form $\mathrm{N}_p(\mathbf{x}_i|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)^{1/\delta}$, where the "power-parameter" ($\delta > 1$) would inflate the covariance. While from a computational standpoint the proposed framework is scalable and can also handle the $p > n$ case, in very high dimensions it would become computationally burdensome. This can be potentially addressed via high-dimensional projections. Finally, although our EM convergence result is general, there remain open theoretical questions concerning rates of convergence and optimality of the estimators themselves.

Acknowledgments

We would like to thank Keefe Murphy for interesting discussions and for updating R package `MoEClust` in order to make it possible to implement the comparisons presented in this paper. We acknowledge support via the German Bundesministerium für Bildung und Forschung (BMBF) project "MechML", the Medical Research Council (programme number MC_UU_00002/17) and the National Institute for Health Research (NIHR) Cambridge Biomedical Research Centre.

Appendix A. Simulations for Section 1.2

The simulation results presented in Section 1.2 are based on 20 repetitions where we consider two groups ($K = 2$) with total $n = 200$ and a balanced design, i.e. $n_k = 100$ for $k \in \{1, 2\}$. The number of predictors is $p = 10$. In each group only one predictor ($\mathbf{x}_k^*$) has a non-zero coefficient (β_k^*), and this predictor is chosen randomly over the 20 repetitions. The three cases in Figures 1 and 2 correspond to: (i) $\beta_1^* = \beta_2^* = 0.5$ (plots on the left), (ii) $\beta_1^* = 0.5, \beta_2^* = 1$ (plots on the middle), and (iii) $\beta_1^* = 0.5, \beta_2^* = 1.5$ (plots on the right). The features are generated as $\mathbf{X}_k \sim \mathrm{N}_{10}(\boldsymbol{\mu}_k, 0.5\mathbf{I}_p)$. That is, the two feature groups share the same diagonal covariance structure, but we let the mean vectors to vary. Specifically, the first mean vector is always $\boldsymbol{\mu}_1 = (0, \ldots, 0)^T$, while for the second mean vector we consider again three cases (which lead to the variability with respect to the x-axis of the plots in Figure 2); namely, (i) $\boldsymbol{\mu}_2 = (0, \ldots, 0)^T$, (ii) $\boldsymbol{\mu}_2 = (0.5, \ldots, 0.5)^T$ and (iii) $\boldsymbol{\mu}_2 = (1, \ldots, 1)^T$. The response of each group is generated as $\mathbf{y}_k \sim \mathrm{N}_{n_k}\big(\mathbf{x}_k^*\beta_k^*,\ \sigma_k^2\mathbf{I}_{n_k}\big)$, where $\sigma_k^2 = \mathrm{Var}(\mathbf{x}_k^*\beta_k^*)/5$ for $k \in \{1, 2\}$. The regularized joint mixture model is based on the normal-Jeffreys prior discussed in Section 2.2.2. For the implementation of the Gaussian mixture and mixture-of-experts models we used `R` packages `mclust` (Scrucca et al., 2016) and `MoEClust` (Murphy and Murphy, 2020), respectively, using the default model-search options, selecting the BIC-optimal model.

Appendix B. Justification for the RLasso Pareto prior

We generally want the Pareto prior to be such that it will not penalize the regression coefficients asymptotically. Under the prior in (11) the mode of λ_k is a_n, while the prior mean is given by

$$\mathbb{E}(\lambda_k) = a_n \frac{b_n}{b_n - 1},$$

for $b_n > 1$ and the prior variance by

$$\mathrm{Var}(\lambda_k) = a_n^2 \frac{b_n}{(b_n - 1)^2(b_n - 2)}$$

for $b_n > 2$. Given that $a_n \to 0$ as $n \to \infty$, in order to meet our requirement we want $b_n \to C > 2$ as $n \to \infty$; to that end, we specify $b_n = (p - 1) - c\sqrt{2K \log p/n}$, for $c \in (0, 1]$. Explicit specification of a_n is not required as it does not affect the posterior mode; any decreasing function of n (subject to $a_n > 0$) will satisfy the recuirement. As for c, we recommend setting it equal to $\min\{\sqrt{2p/3n}, 1\}$ as a default option.

Appendix C. Objective function under the NJ prior

The hierarchical form of the NJ prior is $\boldsymbol{\beta}_k|\mathbf{S}_k \sim \mathrm{N}_p(\mathbf{0}, \mathbf{S}_k)$, where $\mathbf{S}_k = \mathrm{diag}(s_{k1}, \ldots, s_{kp})$ assuming *latent* s_{kj} with $\pi(s_{kj}) \propto s_{kj}^{-1}$ for $k = 1, \ldots, K$ and $j = 1, \ldots, p$. The conditional distribution of any s (dropping momentarily subscripts k, j for simplicity) is

$$p(s|\beta) = \frac{q(s)}{\int q(s)\mathrm{d}s}, \text{ with } q(s) = p(\beta|s)s^{-1} \text{ and } \int q(s)\mathrm{d}s = |\beta|^{-1}.$$

Given this, it follows that

$$\mathbb{E}_{s|\beta}[s^{-1}] = \int s^{-1} p(s|\beta)\mathrm{d}s = \beta^{-2} \tag{41}$$

This result will be needed in the derivation of the E-step below. The joint prior of $\boldsymbol{\beta}_k$ and σ_k^2 is $p(\boldsymbol{\beta}, \boldsymbol{\sigma}^2|\mathbf{S}) = p(\boldsymbol{\beta}|\mathbf{S})p(\boldsymbol{\sigma}^2) \propto \prod_{k=1}^K \exp\big(-\frac{1}{2}\boldsymbol{\beta}_k^T \mathbf{S}_k^{-1}\boldsymbol{\beta}_k\big)\frac{1}{\sigma_k^2}$. The objective under the NJ prior presented in Eq. (19) in the main paper, is derived as follows.

$$\begin{aligned}
Q_{\mathrm{NJ}}^Y(\boldsymbol{\theta}^Y|\boldsymbol{\theta}^{Y(t)}) =& \mathbb{E}_{\mathbf{z},\mathbf{S}|\mathbf{y},\mathbf{X},\boldsymbol{\theta}^{Y(t)}}\big[\log f(\mathbf{y}|\mathbf{X}, \mathbf{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\sigma}^2) + \log \pi(\boldsymbol{\beta}|\mathbf{S}) + \log \pi(\boldsymbol{\sigma}^2)\big] \\
=& \mathbb{E}_{\mathbf{z}|\mathbf{y},\mathbf{X},\boldsymbol{\theta}^{Y(t)}}\big[\log f(\mathbf{y}|\mathbf{X}, \mathbf{z}, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\sigma}^2)\big] + \mathbb{E}_{\mathbf{S}|\boldsymbol{\beta}^{(t)}}\big[\log \pi(\boldsymbol{\beta}|\mathbf{S})\big] + \log \pi(\boldsymbol{\sigma}^2) \\
=& \sum_i \mathbb{E}_{\mathbf{z}|\mathbf{y},\mathbf{X},\boldsymbol{\theta}^{Y(t)}}\big[\log f(y_i|\mathbf{x}_i, z_i, \boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\sigma}^2)\big] + \mathbb{E}_{\mathbf{S}|\boldsymbol{\beta}^{(t)}}\big[\log \pi(\boldsymbol{\beta}|\mathbf{S})\big] + \log \pi(\boldsymbol{\sigma}^2) \\
=& \sum_{i=1}^n \sum_{k=1}^K m_{ki}^{(t)} \left\{ -\frac{1}{2\sigma_k^2}(y_i - \alpha_k - \mathbf{x}_i^T\boldsymbol{\beta}_k)^2 - \frac{1}{2}\log \sigma_k^2 \right\} \qquad (42) \\
& + \sum_{k=1}^K \left\{ \mathbb{E}_{\mathbf{S}_k|\boldsymbol{\beta}_k^{(t)}}\left[-\frac{1}{2}\boldsymbol{\beta}_k^T \mathbf{S}_k^{-1}\boldsymbol{\beta}_k \right] \right\} - \sum_{k=1}^K \left\{ \log \sigma_k^2 \right\} \\
=& \sum_{k=1}^K \left\{ -\frac{1}{2\sigma_k^2}(\mathbf{y} - \alpha_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k)^T \mathbf{M}_k^{(t)} (\mathbf{y} - \alpha_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k) - \frac{n_k^{(t)}}{2}\log \sigma_k^2 \right\} \\
& + \sum_{k=1}^K \left\{ -\frac{1}{2}\boldsymbol{\beta}_k^T \mathbb{E}_{\mathbf{S}_k|\boldsymbol{\beta}_k^{(t)}}\big[\mathbf{S}_k^{-1}\big]\boldsymbol{\beta}_k \right\} - \sum_{k=1}^K \left\{ \log \sigma_k^2 \right\} \qquad (43) \\
=& \sum_{k=1}^K \left\{ -\frac{1}{2\sigma_k^2}(\mathbf{y} - \alpha_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k)^T \mathbf{M}_k^{(t)} (\mathbf{y} - \alpha_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k) - \frac{n_k^{(t)} + 2}{2}\log \sigma_k^2 \right\} \\
& + \sum_{k=1}^K \left\{ -\frac{1}{2}\boldsymbol{\beta}_k^T \mathbf{V}_k^{(t)} \boldsymbol{\beta}_k \right\} \qquad (44) \\
=& -\frac{1}{2}\sum_{k=1}^K \left\{ \frac{(\mathbf{y} - \alpha_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k)^T \mathbf{M}_k^{(t)} (\mathbf{y} - \alpha_k\mathbf{1}_n - \mathbf{X}\boldsymbol{\beta}_k)}{\sigma_k^2} + \boldsymbol{\beta}_k^T \mathbf{V}_k^{(t)} \boldsymbol{\beta}_k \right. \\
& \qquad \left. + (n_k^{(t)} + 2)\log \sigma_k^2 \right\},
\end{aligned}$$

where $m_{ki}^{(t)}$ appearing in (42) is given in (14) in the main paper, $\mathbf{M}_k^{(t)} = \text{diag}(\mathbf{m}_k^{(t)})$ with $\mathbf{m}_k^{(t)} = (m_{k1}^{(t)}, \ldots, m_{kn}^{(t)})^T$ and $\mathbf{V}_k^{(t)} = \text{diag}\Big(1/\beta_{k1}^{2(t)}, \ldots, 1/\beta_{kp}^{2(t)}\Big)$. The transition from (43) to (44) is due to (41).

Appendix D. Details and implementation of the EM

For the graphical lasso optimization in (22) in the main paper we use the efficient R package `glassoFast` (Sustik and Calderhead, 2012). For the lasso optimizations in (26) we use `glmnet` (Friedman et al., 2008b) with penalty equal to $\lambda_k^{(t+1)}/n_k^{(t)}$.

We initialize the algorithm via a simple clustering of the data. For this we use R package `mclust`. Through the resulting group assignments we obtain initial estimates $\boldsymbol{\theta}_k^{X(0)}$ and $\boldsymbol{\theta}_k^{Y(0)}$. In order to initiate EMs from different starting points we add random perturbations to $\boldsymbol{\mu}_k^{(0)}$, $\boldsymbol{\beta}_k^{(0)}$ and $\sigma_k^{2(0)}$ and positive random perturbations to the diagonal elements of $\boldsymbol{\Sigma}_k^{(0)}$. The multiple EMs can be easily run in parallel. As a default option we use ten EM starts.

For the termination of the algorithm we use a combination of two criteria that are commonly used in practice. The first is to simply set a maximum number (T) of EM iterations. Empirical results suggest that the option $T = 20$ is sufficient. The second criterion takes into account the relative change in the objective function in (15); namely, the algorithm is stopped when

$$\left| \frac{Q(\boldsymbol{\theta}, \boldsymbol{\tau}, \boldsymbol{\lambda} | \boldsymbol{\theta}^{(t)}, \boldsymbol{\tau}^{(t)}, \boldsymbol{\lambda}^{(t)})}{Q(\boldsymbol{\theta}, \boldsymbol{\tau}, \boldsymbol{\lambda} | \boldsymbol{\theta}^{(t-1)}, \boldsymbol{\tau}^{(t-1)}, \boldsymbol{\lambda}^{(t-1)})} - 1 \right| \leqslant \epsilon$$

using as default option $\epsilon = 10^{-6}$. Moreover, the algorithm is stopped, and results are discarded, when the sample size of a certain group becomes prohibitively small for estimation. We define this criterion as a function of total sample size and the number of groups. Specifically, we terminate if $\min_k n_k^{(t)} \leqslant n/(10K)$.

Appendix E. Proof of Theorem 2

We need to prove that with our model and under the assumptions of Theorem 2, all the hypotheses of Theorem 3 of Meng and Rubin (1993) (Theorem 1) are met.

As discussed throughout section 3.2.2 in the main paper, under the RJM model, the following conditions are sufficient to verify all the hypotheses of Theorem 4.1 except for hypothesis 6: the penalty is continuous, differentiable, such that each of the block maximisation is unique and $L(\boldsymbol{\xi})$ infinite on the border of Ξ. The first three conditions are already assumptions of our Theorem. Hence, it remains only to be shown that $L(\boldsymbol{\xi})$ is infinite on the border of Ξ and that hypothesis (6) is met. Both are very similar properties, we prove both of them together. For this task, we make use of the "penalty lower bound assumption" of our Theorem, recalled in Eq. (45).

$$\text{pen}(\boldsymbol{\xi}) \geqslant \delta \sum_{k=1}^{K} \left(\log \tau_k^{-1} + \|\mu_k\| + \|\boldsymbol{\Omega}_k\| + \log \left|\boldsymbol{\Omega}_k^{-1}\right| + f_\lambda(\lambda_k) + \rho_k + \log \rho_k^{-1} + |\chi_k| + \|\boldsymbol{\phi}_k\| \right) . \tag{45}$$

To begin with, we have:

$$L(\boldsymbol{\xi}) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \exp\Bigg(-\frac{1}{2}\Big((y_i\rho_k - \chi_k - \mathbf{x}_i^T\boldsymbol{\phi}_k)^2 - 2\log\rho_k \\ + \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2_{\boldsymbol{\Omega}_k} - \log|\boldsymbol{\Omega}_k| \\ - 2\log\tau_k + (p+1)\log 2\pi + \frac{2}{n}\mathrm{pen}(\boldsymbol{\xi})\Big)\Bigg) .$$

Let

$$f_{i,k}(\boldsymbol{\xi}) = (y_i\rho_k - \chi_k - \mathbf{x}_i^T\boldsymbol{\phi}_k)^2 - 2\log\rho_k + \|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2_{\boldsymbol{\Omega}_k} - \log|\boldsymbol{\Omega}_k| - 2\log\tau_k + \frac{2}{n}\mathrm{pen}(\boldsymbol{\xi}) .$$

Such that

$$L(\boldsymbol{\xi}) = -\frac{n(p+1)\log 2\pi}{2} + \sum_{i=1}^{n} \log \sum_{k=1}^{K} \exp\left(-\frac{1}{2} f_{i,k}(\boldsymbol{\xi})\right) . \tag{46}$$

From Eq. (45) and the fact that $(y_i\rho_k - \chi_k - \mathbf{x}_i^T\boldsymbol{\phi}_k)^2 \geqslant 0$ and $\|\mathbf{x}_i - \boldsymbol{\mu}_k\|^2_{\boldsymbol{\Omega}_k} \geqslant 0$, we have:

$$\begin{aligned} f_{i,k}(\boldsymbol{\xi}) &\geqslant \frac{2}{n}\delta \sum_{l=1}^{K} \Big(-(1 + \frac{n}{\delta}\mathbb{1}_{l=k})\log\tau_l + \|\mu_l\| + \|\boldsymbol{\Omega}_l\| - (1 + \frac{n}{2\delta}\mathbb{1}_{l=k})\log|\boldsymbol{\Omega}_l| \\ &\qquad + f_\lambda(\lambda_l) + \rho_l - (1 + \frac{n}{\delta}\mathbb{1}_{l=k})\log\rho_l + |\chi_l| + \|\boldsymbol{\phi}_l\| \Big) \\ &= \frac{2}{n}\delta \sum_{l=1}^{K} \Big(f^{\boldsymbol{\tau}}_{k,l}(\tau_l) + f^{\boldsymbol{\mu}}_{k,l}(\boldsymbol{\mu}_l) + f^{\boldsymbol{\Omega}}_{k,l}(\boldsymbol{\Omega}_l) + f^{\boldsymbol{\lambda}}_{k,l}(\lambda_l) + f^{\boldsymbol{\rho}}_{k,l}(\rho_l) + f^{\boldsymbol{\chi}}_{k,l}(\chi_l) + f^{\boldsymbol{\phi}}_{k,l}(\boldsymbol{\phi}_l)\Big) . \end{aligned} \tag{47}$$

Where:

$$\begin{aligned} f^{\boldsymbol{\tau}}_{k,l}(\tau_l) &:= -(1 + \frac{n}{\delta}\mathbb{1}_{l=k})\log\tau_l , \\ f^{\boldsymbol{\mu}}_{k,l}(\boldsymbol{\mu}_l) &:= \|\boldsymbol{\mu}_l\| , \\ f^{\boldsymbol{\Omega}}_{k,l}(\boldsymbol{\Omega}_l) &:= \|\boldsymbol{\Omega}_l\| - (1 + \frac{n}{2\delta}\mathbb{1}_{l=k})\log|\boldsymbol{\Omega}_l| , \\ f^{\boldsymbol{\lambda}}_{k,l}(\lambda_l) &:= f_\lambda(\lambda_l) , \\ f^{\boldsymbol{\rho}}_{k,l}(\rho_l) &:= \rho_l - (1 + \frac{n}{\delta}\mathbb{1}_{l=k})\log\rho_l , \\ f^{\boldsymbol{\chi}}_{k,l}(\chi_l) &:= |\chi_l| , \\ f^{\boldsymbol{\phi}}_{k,l}(\boldsymbol{\phi}_l) &:= \|\boldsymbol{\phi}_l\| . \end{aligned} \tag{48}$$

The dependency on k, l is denoted in the indices of all these functions for the sake of uniformity, although only $f^{\boldsymbol{\tau}}_{k,l}, f^{\boldsymbol{\Omega}}_{k,l}$ and $f^{\boldsymbol{\rho}}_{k,l}$ actually depend on k and l. We recall that, with $a > 0$, the function $x \mapsto x - a\log x$, is lower bounded on $\mathbb{R}^*_+$, and converges towards $+\infty$ both when $x \to 0$ and when $x \to +\infty$. To analyse $f^{\boldsymbol{\Omega}}_{k,l}$, it is convenient to consider the nuclear norm for $\|\boldsymbol{\Omega}_k\|$ and rewrite the whole as: $f^{\boldsymbol{\Omega}}_{k,l}(\boldsymbol{\Omega}_l) = \sum_j \psi_{l,j} - (1 + \frac{n}{2\delta}\mathbb{1}_{l=k})\log\psi_{l,j}$, with $\{\psi_{l,j}\}_{j=1}^p$ the eigenvalues of $\boldsymbol{\Omega}_l$.

With these observations at hand, note that all the functions in (48) can be lower bounded by the same constant $c > -\infty$, valid for all values of k and l. They also all converge towards $+\infty$ on the boundary of their respective sets of definition.

For $m > 0$, we define Ξ_m as the compact subset of Ξ such that $\xi \in \Xi_m \iff \xi \in \Xi$ and $\forall k = 1, ..., K$:

$$\begin{aligned} \tau_k &\geqslant \frac{1}{m}\,, \\ \|\boldsymbol{\mu}_k\| &\leqslant m\,, \\ \psi_{\max}(\boldsymbol{\Omega}_k) &\leqslant m\,, \\ \psi_{\min}(\boldsymbol{\Omega}_k) &\geqslant \frac{1}{m}\,, \\ \frac{1}{m} &\leqslant \lambda_k \leqslant m\,, \\ \frac{1}{m} &\leqslant \rho_k \leqslant m\,, \\ |\chi_k| &\leqslant m\,, \\ \|\boldsymbol{\phi}_k\| &\leqslant m\,. \end{aligned} \tag{49}$$

It is clear that $\forall m > 0,\ \Xi_m \subseteq \Xi^\circ$.

With all these objects defined, we can finish the proof. For any real number $A > -\infty$, let us show that there exists $M > 0$ such that $\forall \xi \in \Xi \backslash \Xi_M,\ L(\xi) < A$. First consider the following: for any real number $B > -\infty$, there exists a $m_B > 0$ such that, for all k and l:

$$\begin{aligned} &\text{if } \tau_l < \frac{1}{m_B}, \text{ then } f^{\boldsymbol{\tau}}_{k,l}(\tau_l) > B\,, \\ &\text{if } \|\boldsymbol{\mu}_l\| > m_B, \text{ then } f^{\boldsymbol{\mu}}_{k,l}(\boldsymbol{\mu}_l) > B\,, \\ &\text{if } \psi_{\max}(\boldsymbol{\Omega}_l) > m_B, \text{ then } f^{\boldsymbol{\Omega}}_{k,l}(\boldsymbol{\Omega}_l) > B\,, \\ &\text{if } \psi_{\min}(\boldsymbol{\Omega}_l) < \frac{1}{m_B}, \text{ then } f^{\boldsymbol{\Omega}}_{k,l}(\boldsymbol{\Omega}_l) > B\,, \\ &\text{if } \lambda_l < \frac{1}{m_B} \text{ or } \lambda_l > m_B, \text{ then } f^{\boldsymbol{\lambda}}_{k,l}(\lambda_l) > B\,, \\ &\text{if } \rho_l < \frac{1}{m_B} \text{ or } \rho_l > m_B, \text{ then } f^{\boldsymbol{\rho}}_{k,l}(\rho_l) > B\,, \\ &\text{if } |\chi_l| > m_B, \text{ then } f^{\boldsymbol{\chi}}_{k,l}(\chi_l) > B\,, \\ &\text{if } \|\boldsymbol{\phi}_l\| > m_B, \text{ then } f^{\boldsymbol{\phi}}_{k,l}(\boldsymbol{\phi}_l) > B\,. \end{aligned} \tag{50}$$

If $\xi \in \Xi \backslash \Xi_{m_B}$, then by definition of the sets Ξ_m (49), there exist at least one $l \in \{1, ..., K\}$ such that at least one of the above scenarios is realised. By injecting the resulting lower bound into the inequality (47), we get:

$$\forall k, \quad f_{i,k}(\boldsymbol{\xi}) > \frac{2}{n}\delta(B + (7K - 1)c)\,.$$

Then, from Eq. (46):

$$\begin{aligned} L(\boldsymbol{\xi}) &= -\frac{n(p+1)\log 2\pi}{2} + \sum_{i=1}^{n} \log \sum_{k=1}^{K} \exp\left(-\frac{1}{2} f_{i,k}(\boldsymbol{\xi})\right) \\ &< -\frac{n(p+1)\log 2\pi}{2} + \sum_{i=1}^{n} \log \sum_{k=1}^{K} \exp\left(-\frac{1}{n}\delta(B + (7K-1)c)\right) \\ &= -\frac{n(p+1)\log 2\pi}{2} + n\log K - \delta(B + (7K-1)c)\,. \end{aligned}$$

Since $\delta > 0$, then there exists $B_A > 0$ such that for all $B \geqslant B_A$:

$$-\frac{n(p+1)\log 2\pi}{2} + n\log K - \delta(B + (7K-1)c) < A\,.$$

As a consequence, $M := m_{B_A}$ is such that $\forall \boldsymbol{\xi} \in \Xi \backslash \Xi_M,\ L(\boldsymbol{\xi}) < A$. In other words $\{\boldsymbol{\xi} \in \Xi | L(\boldsymbol{\xi}) \geqslant A\} \subseteq \Xi_M$.

We have proven that for any $A > -\infty$, there exists $M > 0$ such that $\{\boldsymbol{\xi} \in \Xi | L(\boldsymbol{\xi}) \geqslant A\} \subseteq \Xi_M$. Since the Ξ_m are compacts, this means that the closed set $\{\boldsymbol{\xi} \in \Xi | L(\boldsymbol{\xi}) \geqslant A\}$ is also a compact, hence hypothesis (6) of Wu (1983) is verified. Moreover, since $\Xi_m \subseteq \Xi^\circ$, this means that the log-likelihood goes to $-\infty$ on the border of Ξ. Hence no EM sequence will take values on the border, hence we can safely consider that $\Xi = \Xi^\circ$. With these last two hypotheses verified, we can apply Theorem 3 of Meng and Rubin (1993) and benefit from the convergence guarantees.

Appendix F. Further results from Section 5.1

Table 3: First simulation. Intercept and slope parameter values for the two groups under the three cases illustrated in Figure 3 in the main paper.

Case	Group	Intercept	Slope
A	1	0	1
	2	0	-1
B	1	0	1
	2	1	1
C	1	0	1
	2	0	1

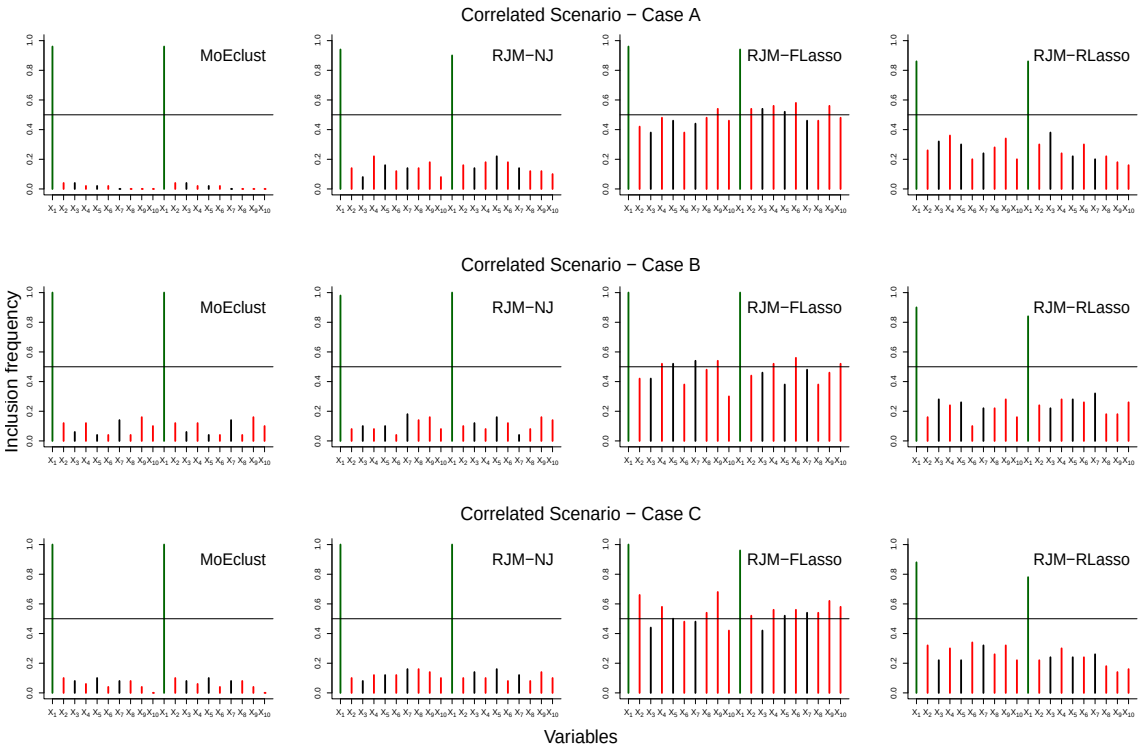

Figure 12: First simulation, correlated scenario. Variable inclusion frequencies (under 50 repetitions) for signal variables (in green), correlated noise variables (in black) and uncorrelated noise variables (in red) for regression cases A, B and C. Horizontal black lines correspond to a frequency of 0.5.

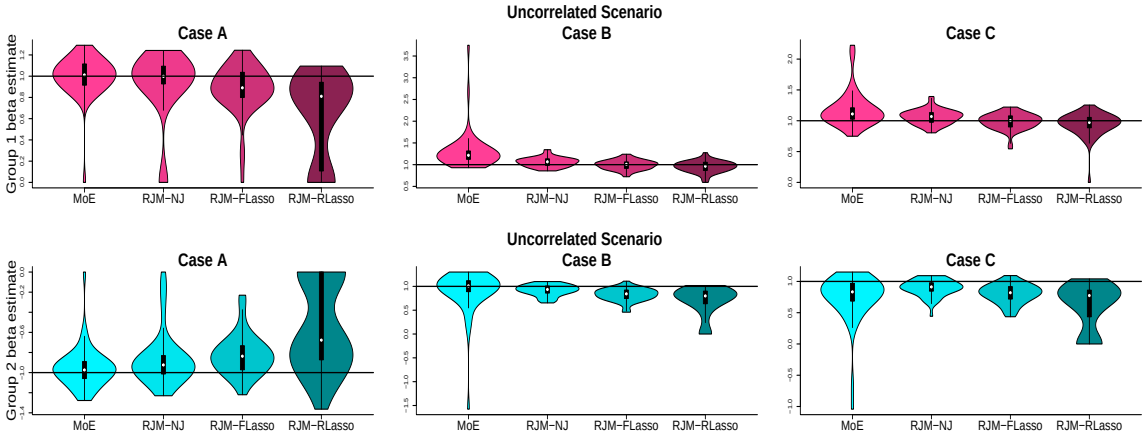

Figure 13: First simulation, uncorrelated scenario. Violin plots of MoEclust and RJM slope estimates (from 50 repetitions) for cases A, B and C. Horizontal black lines correspond to the true slopes.

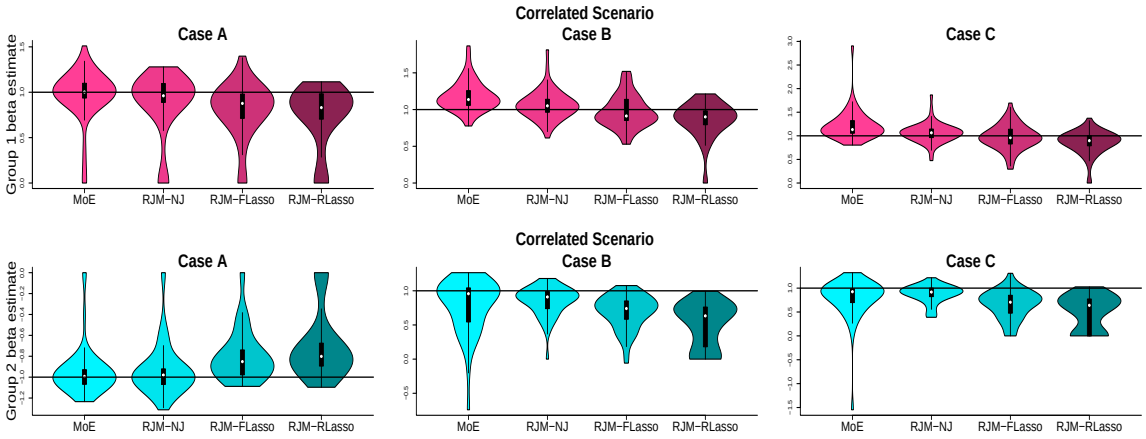

Figure 14: First simulation, correlated scenario. Violin plots of MoEclust and RJM slope estimates (from 50 repetitions) for cases A, B and C. Horizontal black lines correspond to the true slopes.

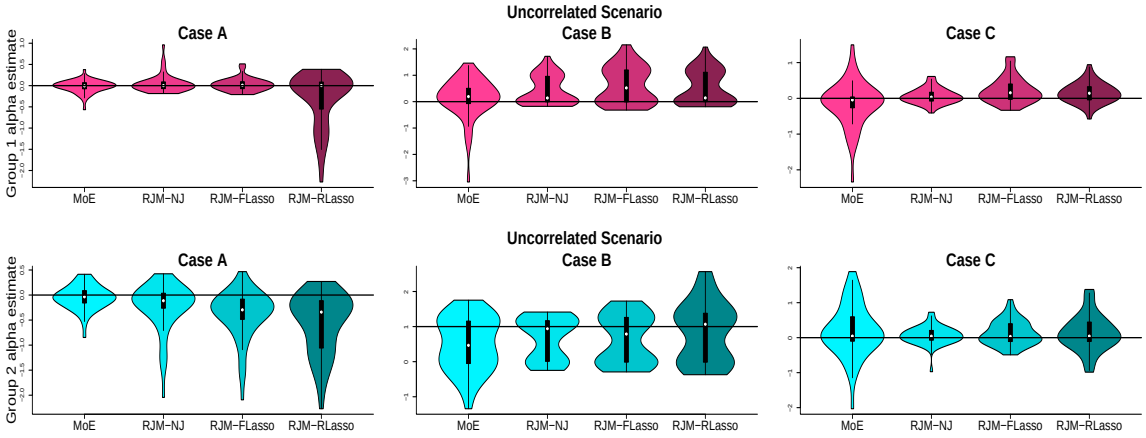

Figure 15: First simulation, uncorrelated scenario. Violin plots of MoEclust and RJM intercept estimates (from 50 repetitions) for cases A, B and C. Horizontal black lines correspond to the real intercepts.

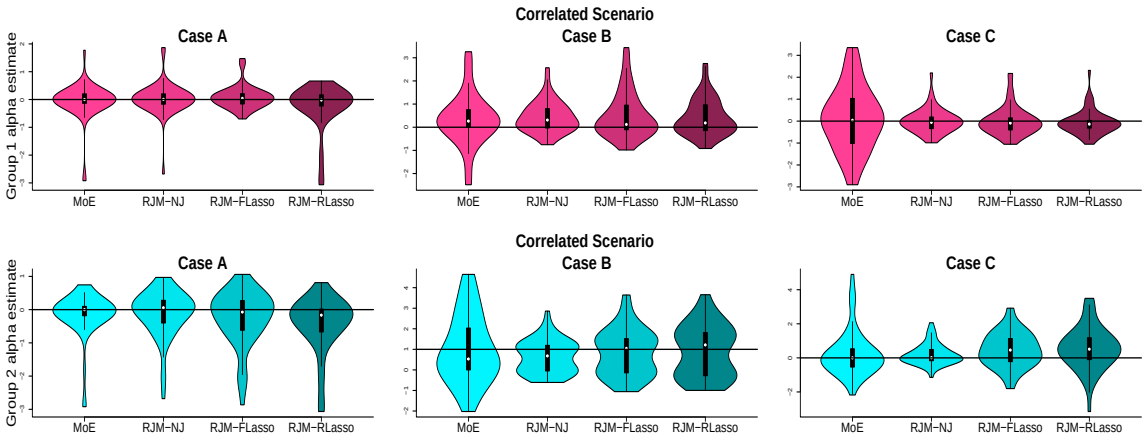

Figure 16: First simulation, correlated scenario. Violin plots of MoEclust and RJM intercept estimates (from 50 repetitions) for cases A, B and C. Horizontal black lines correspond to the real intercepts.

Appendix G. Further results from Section 5.2

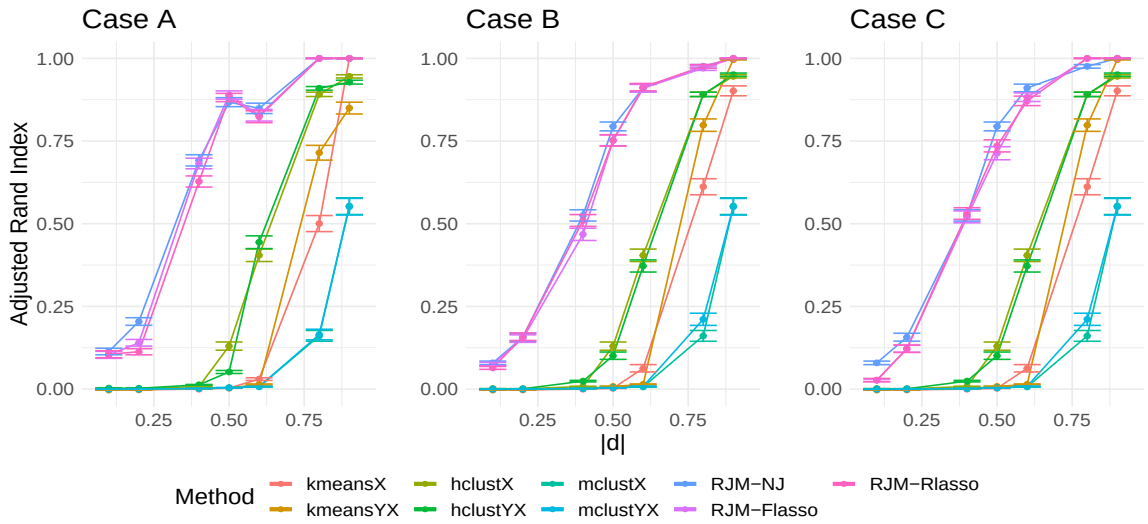

Figure 17: Second simulation, $p = 100$, group assignment. Average adjusted Rand Index as a function of the absolute distance ($|d|$) of the group-wise covariate means, for cases A (left), B (center) and C (right). [Error bars indicate standard errors from 20 repetitions.]

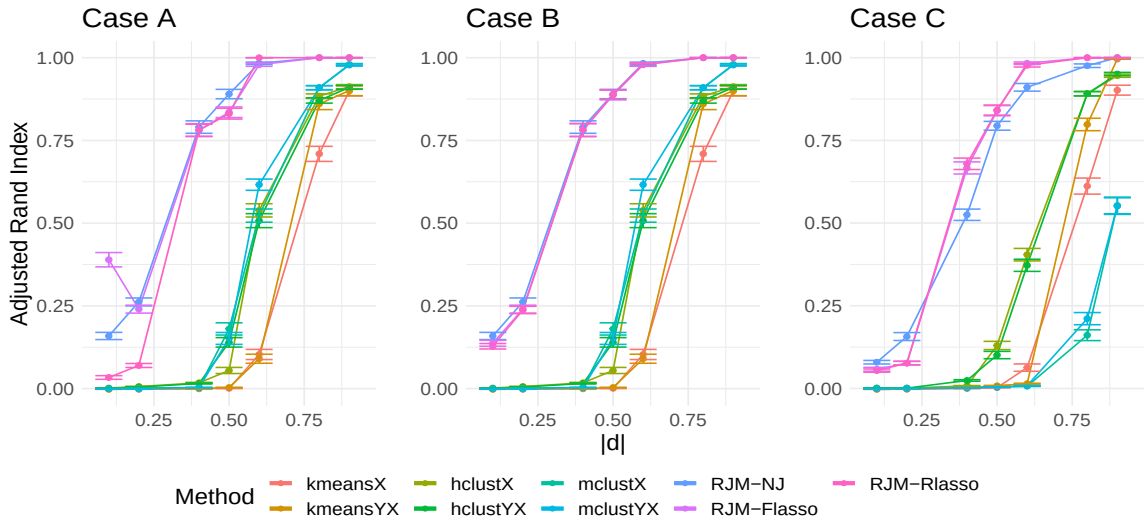

Figure 18: Second simulation, $p = 250$, group assignment. Average adjusted Rand Index as a function of the absolute distance ($|d|$) of the group-wise covariate means, for cases A (left), B (center) and C (right). [Error bars indicate standard errors from 20 repetitions.]

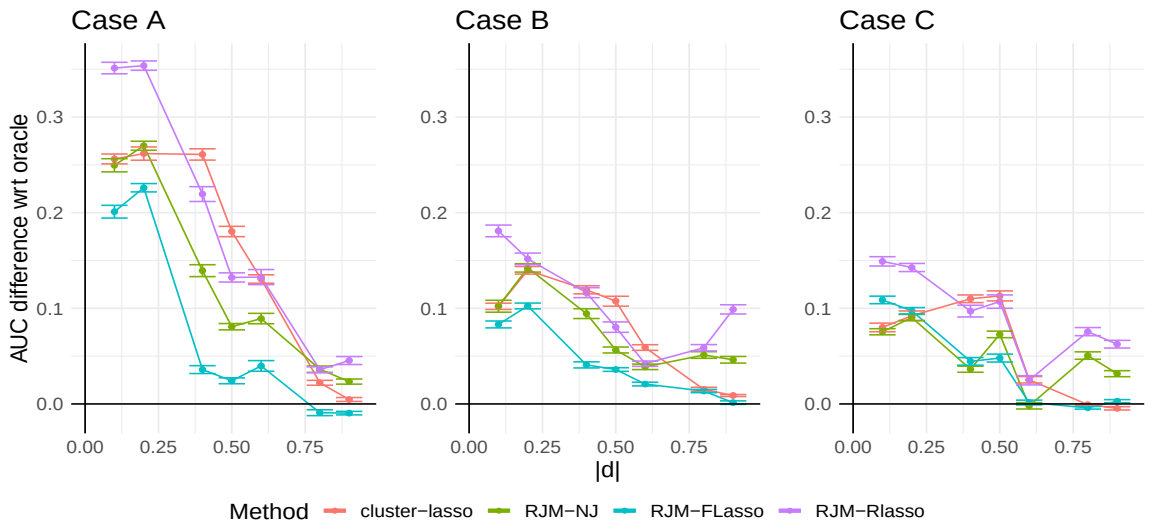

Figure 19: Second simulation, $p = 100$, variable selection. AUC loss from oracle-lasso as a function of the absolute distance ($|d|$) of the group-wise covariate means, for cases A (left), B (center) and C (right). [Error bars indicate standard errors from 20 repetitions.]

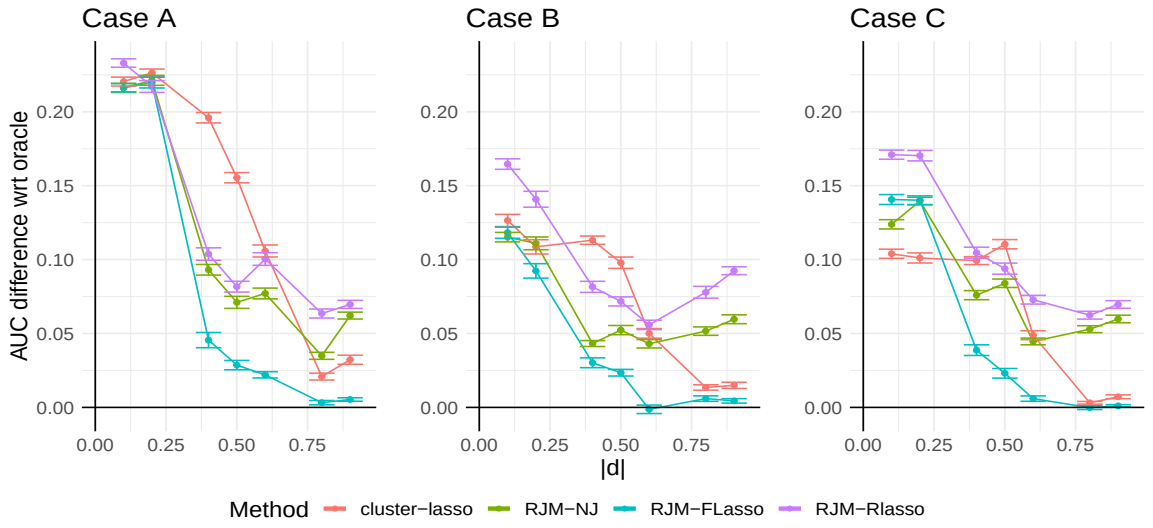

Figure 20: Second simulation, $p = 250$, variable selection. AUC loss from oracle-lasso as a function of the absolute distance ($|d|$) of the group-wise covariate means, for cases A (left), B (center) and C (right). [Error bars indicate standard errors from 20 repetitions.]

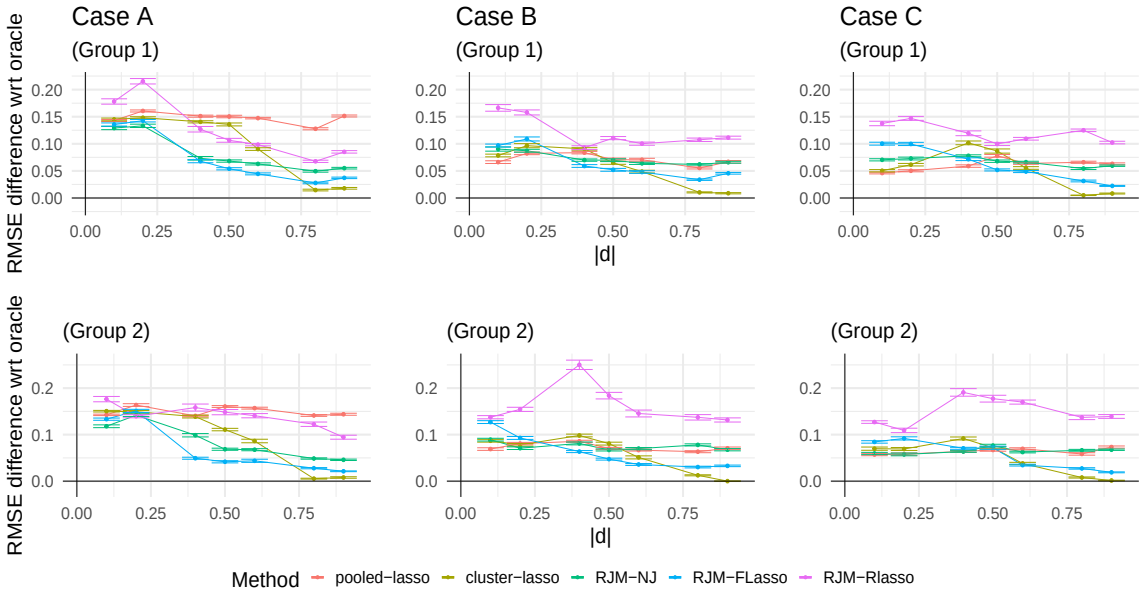

Figure 21: Second simulation, $p = 100$, regression coefficients estimation. Increase in RMSE relative to the oracle-lasso as a function of the absolute distance ($|d|$) of the group-wise covariate means, under group one (top) and group two (bottom), and for cases A (left), B (center) and C (right). [Error bars indicate standard errors from 20 repetitions.]

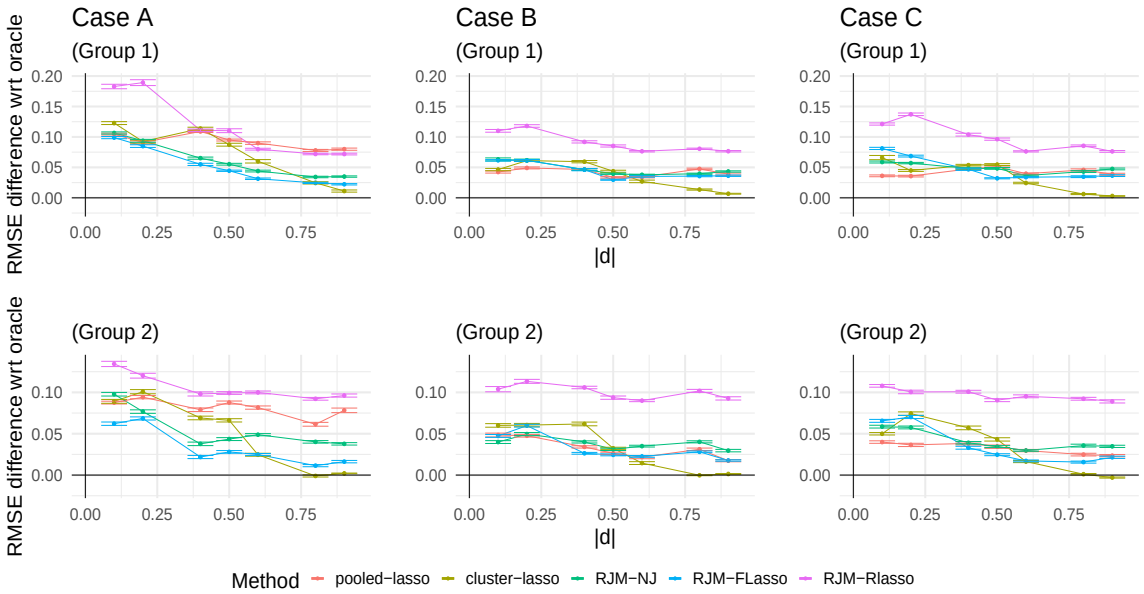

Figure 22: Second simulation, $p = 250$, regression coefficients estimation. Increase in RMSE relative to the oracle-lasso as a function of the absolute distance ($|d|$) of the group-wise covariate means, under group one (top) and group two (bottom), and for cases A (left), B (center) and C (right). [Error bars indicate standard errors from 20 repetitions.]

Appendix H. Selection of number of groups in Section 5.2

Here we consider all four cancer types (BRCA, KIRC, LUAD, THCA) and provide some results on cluster selection under unknown number of groups using the predictive approach described in Section 4. We consider three simulation settings where the respective true number of groups is $g^* = 2$ (using the BRCA and KIRC cancer types), $g^* = 3$ (BRCA, KIRC, LUAD) and $g^* = 4$ (further including THCA). For each setting we fit RJM models with two, three and four components. The simulations are along the lines of Section 5.2 considering Case A of Table 1 for $p = 100$. The conditions outlined in Table 1 for the β_j^*'s at the common locations are used again for $g^* \in \{3, 4\}$. Here we use the real group-sample proportions as they occur in the TCGA data set and assume that sample size grows with number of groups (for the simulations to be on an equal basis). Specifically, we set $n = 250 \times g^*$. The resulting group sample sizes for $g^* \in \{2, 3, 4\}$ are $(n_1 = 335, n_2 = 165)$, $(n_1 = 382, n_2 = 188, n_3 = 180)$ and $(n_1 = 410, n_2 = 200, n_3 = 200, n_4 = 190)$, respectively. We use 80% of the samples for training and 20% for testing.

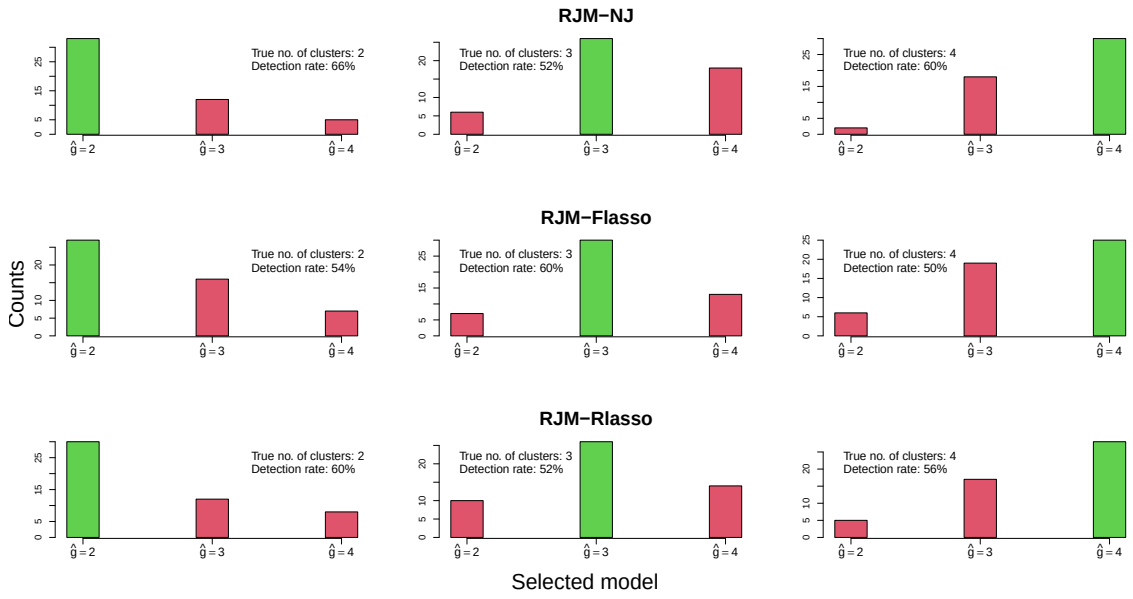

Figure 23: Second simulation, cluster selection. Barplots of selected clusters (50 repetitions) using the predictive approach in Section 4. Correct identification is highlighted in green; true number of clusters and detection rate of the correct model is annotated in each panel.

Figure 23 shows barplots of the selected number of clusters resulting from 50 repetitions of the simulations. As seen, the correct model is selected in majority under all cases. The detection rate of the correct model is also annotated in each panel of Figure 23. Here, RJM-NJ is slightly better with average overall detection rate of 59%, while RLasso and FLasso have 56% and 55%, respectively.

Appendix I. Selection of number of groups in Section 5.3

Here we present additional results concerning performance including a model selection step. The setting is as in Section 5.3 in the main text except that sample size is equal to 200. In particular, treating in turn each of the genes as response, with all others considered as features. A model selection step is included over the number of clusters, selecting between models with $g \in \{2, 3, 4\}$ components based on BIC. Here we compare with GMMs (`mclust`) and an MoE implementation from package `flexmix` (Grün and Leisch, 2008). Under the latter method we use elastic-net regularization (Zou and Hastie, 2005) in the expert networks and intercept-only multinomial gating functions (the latter is better than allowing 99 predictors to enter the gating functions in the absence of regularization for this part of the model). We note that results from `MoEclust` are not presented here as this method (based on incremental forward model search) never selected the correct model with four clusters. Results from the 100 attempts are summarized in Table 4.

Table 4: TCGA data application, performance including a model selection step. Number of times, out of 100 applications to the TGCA data, that the methods selected two, three and four clusters based on BIC. Each time a different gene expression was used as response variable with the predictor matrix containing the remaining 99 gene expressions. The correct number of clusters is four corresponding to the four cancer types included in the dataset.

Methods and cluster selection			
Estimated number of clusters	$\hat{g} = 2$	$\hat{g} = 3$	$\hat{g} = 4$
GMM (`mclust`)	5	91	4
MoE (`flexmix`)	18	67	15
RJM-NJ	17	18	65
RJM-FL	16	17	67
RJM-RL	17	19	64

References

H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 1974.

A. Anandkumar, D. J. Hsu, F. Huang, and S. M. Kakade. Learning mixtures of tree graphical models. In *NIPS*, pages 1061–1069, 2012.

K. Bae and B. Mallick. Gene selection using a two-level hierarchical Bayesian model. *Bioinformatics*, 20(18):3423–3430, 2004.

J. D. Banfield and A. E. Raftery. Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49(3):803–821, 1993.

C. Carvalho, N. Polson, and J. Scott. The horseshoe estimator for sparse signal. *Biometrika*, 97(2):465–480, 2010.

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

F. Chamroukhi and B. T. Huynh. Regularized maximum-likelihood estimation of mixture-of-experts for regression and clustering. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2018.

U. J. Dang and P. D. McNicholas. Families of Parsimonious Finite Mixtures of Regression Models. In I. Morlini, T. Minerva, and M. Vichi, eds., *Advances in Statistical Models for Data Analysis*, pages 73–84. Springer International Publishing, 2015.

U. J. Dang, A. Punzo, P. D. McNicholas, S. Ingrassia, and R. P. Browne. Multivariate response and parsimony for Gaussian cluster-weighted models. *Journal of Classification*, 34(1):4–34, 2017.

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

L. D. D. Desboulets. A review on variable selection in regression analysis. *Econometrics*, 6 (4), 2018.

D. L. Donoho and J. M. Johnstone. Ideal spatial adaptation by wavelet shrinkage. *Biometrika*, 81(3):425–455, 1994.

M. A. T. Figueiredo. Wavelet-based image estimation: An empirical Bayes approach using Jeffreys' noninformative prior. *IEEE Transactions on Image Processing*, 10(9):1322–1331, 2001.

M. A. T. Figueiredo. Adaptive sparseness for supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(9):1150–1159, 2003.

M. Fop, T.B. Murphy, and L. Scrucca. Model-based clustering with sparse covariance matrices. *Statistics and Computing*, 29(4):791–819, 2019.

C. Fraley and A. E. Raftery. Model-based clustering, discriminant analysis and density estimation. *Journal of the American Statistical Association*, 97(458):611–631, 2002.

J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008a.

J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2008b.

S. Früwirth-Schnatter. *Finite Mixture and Markov Switching Models*. Springer, Heidelberg, 2005.

J. Griffin and P. Brown. Alternative prior distributions for variable selection with very many more variables than observations. Technical report, University of Warwick, 2005.

B. Grün and F. Leisch. FlexMix version 2: Finite mixtures with concomitant variables and varying and constant parameters. *Journal of Statistical Software*, 28(4):1–35, 2008.

Q. Gu and J. Han. Clustered support vector machines. In *Proceedings of the Sixteenth International Conference on Artificial Intelligence and Statistics (AISTATS)*, pages 307–315, 2013.

C. Heinze-Deml and N. Meinshausen. Conditional variance penalties and domain shift robustness. *Machine Learning*, 110:303–348, 2021.

S. Ingrassia, S. C. Minotti, and G. Vittadini. Local statistical modeling via a cluster-weighted approach with elliptical distributions. *Journal of Classification*, 29(3):363–401, 2012.

S. Ingrassia, A. Punzo, G. Vittadini, and S. C. Minotti. The generalized linear mixed cluster-weighted model. *Journal of Classification*, 32(1):85–113, 2015.

R. A. Jacobs. Bias/variance analyses of mixtures-of-experts architectures. *Neural Computation*, 9(2):369–383, 1997.

R. A. Jacobs, M.I. Jordan, S.I. Nowlan, and G.E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991.

I. T. Jolliffe. A note on the use of principal components in regression. *Journal of the Royal Statistical Society, Series C*, 31(3):300–303, 1982.

M. I. Jordan and R. A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214, 1994.

A. Khalili. New estimation and feature selection methods in mixture-of-experts models. *Canadian Journal of Statistics*, 38(4):519–539, 2010.

A. Khalili and C. Jiahua. Variable selection in finite mixture of regression models. *Journal of the American Statistical Association*, 102(479):1025–1038, 2007.

A. Khalili and S. Lin. Regularization in finite mixture of regression models with diverging number of parameters. *Biometrics*, 69(2):436–446, 2013.

S. Liverani, D. I. Hastie, L. Azizi, M. Papathomas, and S. Richardson. PReMiuM: An R package for profile regression mixture models using Dirichlet processes. *Journal of Statistical Software*, 64(7):1–30, 2015.

G.J. McLachlan and D. Peel. *Finite Mixture Models*. John Wiley, New York, USA, 2000.

D. P. McNicholas and T. B. Murphy. Parsimonious Gaussian mixture models. *Statistics and Computing*, 18(3):285–296, 2008.

N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(3):1436–1462, 2006.

X.-L. Meng and D. B. Rubin. Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika*, 80(2):267–278, 1993.

J. Molitor, M. Papathomas, M. Jerrett, and S. Richardson. Bayesian profile regression with an application to the National survey of children's health. *Biostatistics*, 11(3):484–498, 2010.

K. Murphy and T. B. Murphy. Gaussian parsimonious clustering models with covariates and a noise component. *Advances in Data Analysis and Classification*, 14(2):293–325, 2020.

T. Park and G. Casella. The Bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686, 2008.

N. G. Polson and J. G. Scott. Shrink Globally, Act Locally: Sparse Bayesian Regularization and Prediction. In J.M. Bernardo, M.J. Bayarri, J.O. Berger, A.P. David, D. Heckerman, A.F.M. Smith and M. West, eds., *Bayesian Statistics*, Vol. 9, pages 501–538. Oxford University Press, 2010.

N. G. Polson and J. G. Scott. On the half-cauchy prior for a global scale parameter. *Bayesian Analysis*, 7(4):887–902, 2012.

A. Punzo and S. Ingassia. Clustering bivariate mixed-type data via the cluster-weighted model. *Computational Statistics*, 31(3):989–1013, 2016.

B. Recht, R. Roelofs, L. Schmidt, and V. Shankar. Do CIFAR-10 classifiers generalize to CIFAR-10? *arXiv:1806.00451 [cs.LG]*, 2018.

G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464, 1978.

L. Scrucca, M. Fop, T. B. Murphy, and Raftery A. E. mclust 5: clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, 8(1):289–317, 2016.

N. Städler and S. Mukherjee. Penalized estimation in high-dimensional hidden Markov models with state-specific graphical models. *The Annals of Applied Statistics*, 7(4):2157–2179, 2013.

N. Städler, P. Bühlmann, and S. van de Geer. ℓ_1-penalization for mixture regression models. *Test*, 19:209–256, 2010.

N. Städler, F. Dondelinger, S. M. Hill, R. Akbani, Y. Lu, G. B. Mills, and S. Mukherjee. Molecular heterogeneity at the network level: high-dimensional testing, clustering and a TCGA case study. *Bioinformatics*, 33(18):2890–2896, 2017.

S. Subedi, A. Punzo, S. Ingassia, and P.D. McNicholas. Clustering and classification via cluster-weighted factor analyzers. *Advances in Data Analysis and Classification*, 7(1): 5–40, 2013.

M. A Sustik and B Calderhead. GLASSOFAST: An efficient GLASSO implementation. Technical Report TR-12-29:1-3, UTCS, 2012.

B. Taschler, F. Dondelinger, and S. Mukherjee. Model-based clustering in very high dimensions via adaptive projections. *arXiv:1902.08472v1 [stat.ML]*, 2019.

R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society B*, 58(1):267–288, 1996.

S. van Erp, D. L. Oberski, and J. Mulder. Shrinkage priors for Bayesian penalized regression. *Journal of Mathematical Psychology*, 89:31–50, 2019.

H. Wang. Bayesian graphical lasso models and efficient posterior computation. *Bayesian Analysis*, 7(4):867–886, 2012.

C. F. J. Wu. On the convergence properties of the EM algorithm. *The Annals of Statistics*, 11(1):95–103, 1983.

M. Yuan and Y. Lin. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19–35, 2007.

H. Zou. The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429, 2006.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B*, 67(2):301–320, 2005.