Mixture of Conditional Gaussian Graphical Models for unlabelled heterogeneous populations in the presence of co-factors

Thomas Lartigue, Stanley Durrleman, Stéphanie Allassonnière

Statistics and Computing manuscript No.
(will be inserted by the editor)

Mixture of Conditional Gaussian Graphical Models for Unlabelled Heterogeneous Populations in the Presence of Co-factors

Thomas Lartigue[1,2] · **Stanley Durrleman**[1] · **Stéphanie Allassonnière**[3]

Received: date / Accepted: date

Abstract Conditional correlation networks, within Gaussian Graphical Models (GGM), are widely used to describe the direct interactions between the components of a random vector. In the case of an unlabelled Heterogeneous population, Expectation Maximisation (EM) algorithms for Mixtures of GGM have been proposed to estimate both each sub-population's graph and the class labels. However, we argue that, with most real data, class affiliation cannot be described with a Mixture of Gaussian, which mostly groups data points according to their geometrical proximity. In particular, there often exists external co-features whose values affect the features' average value, scattering across the feature space data points belonging to the same sub-population. Additionally, if the co-features' effect on the features is Heterogeneous, then the estimation of this effect cannot be separated from the sub-population identification. In this article, we propose a Mixture of Conditional GGM (CGGM) that subtracts the heterogeneous effects of the co-features to regroup the data points into sub-population corresponding clusters. We develop a penalised EM algorithm to estimate graph-sparse model parameters. We demonstrate on synthetic and real data how this method fulfils its goal and succeeds in identifying the sub-populations where the Mixtures of GGM are disrupted by the effect of the co-features.

Keywords Gaussian Graphical Models · Unlabelled Heterogeneous populations · Conditional GGM · Mixture Models · EM algorithm

1 Introduction

The conditional correlation networks are a popular tool to describe the co-variations between the components of a random vector. Within the Gaussian Graphical Model (GGM) framework, introduced in Dempster (1972), the random vector of interest is modelled as a gaussian vector $\mathcal{N}(\mu, \Sigma)$, and the conditional correlation networks can be recovered from the sparsity of the precision matrix $\Lambda := \Sigma^{-1}$. In this article, we consider the case of an unlabelled heterogeneous population, in which different sub-populations (or "classes") are described by different networks. Additionally, we take into account the presence of observed co-features (discrete and/or

Funding. The research leading to these results has received funding from the European Research Council (ERC) under grant agreement No 678304, European Union's Horizon 2020 research and innovation program under grant agreement No 666992 (EuroPOND) and No 826421 (TVB-Cloud), and the French government under management of Agence Nationale de la Recherche as part of the "Investissements d'avenir" program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute) and reference ANR-10-IAIHU-06 (IHU-A-ICM).
Conflicts of interest. We have no conflict of interest to disclose.
Code availability. Code for our algorithm, as well as a toy example that reproduces some of the results of this paper, publicly available at: https://github.com/tlartigue/Mixture-of-Conditional-Gaussian-Graphical-Models
Authors' contributions. Conceptualisation, T.L. and S.A.; methodology, T.L. and S.A.; software, T.L.; data curation, S.D. and T.L.; validation, T.L. and S.A.; visualisation, T.L.; result analysis, S.D., S.A. and T.L.; writing—original draft preparation; T.L.; writing—review and editing, T.L., S.D. and S.A.; supervision, S.D and S.A.. All authors have read and agreed to the published version of the manuscript.

Thomas Lartigue
thomas.lartigue@inria.fr
Present e-mail: thomas.lartigue@dzne.de

Stanley Durrleman
stanley.durrleman@inria.fr

Stéphanie Allassonnière
stephanie.allassonniere@u-paris.fr

[1] Aramis project-team, INRIA, ICM, Paris, France

[2] CMAP, CNRS, École polytechnique, I.P. Paris, France

[3] Centre de Recherche des Cordeliers, Université de Paris, Inserm, HEKA Project team, INRIA Paris, Sorbonne Université, F-75006, Paris, France

continuous) that have a heterogeneous (class-dependent) impact on the values of the features. The absence of known class labels turns the analysis of the population into an unsupervised problem. As a result, any inference method will have to tackle the problem of cluster discovery in addition to the parameter estimation. The former is a crucial task, especially since the relevance of the estimated parameters is entirely dependent on the clusters identified. The co-features, if their effects on the features are consequent, can greatly disrupt the clustering. Indeed, any unsupervised method will then be more likely to identify clusters correlated with the values of the co-features than with the hidden sub-population labels. This occurs frequently when analysing biological or medical features. To provide a simple illustration, if one runs an unsupervised method on an unlabelled population containing both healthy and obese patients, using the body fat percentage as a feature, then the unearthed clusters are very likely to be more correlated with the gender of the patients (a co-feature) rather than with the actual diagnostic (the hidden variable). Additionally, the fact that the effect of the gender on the average body fat is also dependent on the diagnostic (class-dependent effect) makes the situation even more complex.

The seminal work of Dempster (1972), and its later hierarchical extension introduced in Honorio and Samaras (2010) and Varoquaux et al. (2010), generated tremendous interest in GGM and hierarchical GGM analyses respectively. See Lartigue (2020) for an extensive literature review on these topics. Recently, unsupervised hierarchical GGM have received increasing attention, with works such as Gao et al. (2016) and Hao et al. (2017) adapting the popular supervised joint Hierarchical GGM methods of Mohan et al. (2014) and Danaher et al. (2014) to the unsupervised case. When the labels are known in advance, these joint Hierarchical GGM are useful models to estimate several sparse conditional correlation matrices and are modular enough to allow for the recovery of many different forms of common structure between classes. However, we argue that they are not designed for efficient cluster identification in the unsupervised scenario, and will very likely miss the hidden variable and find clusters correlated to the most influential co-features instead. Which in turn will result in the estimation of irrelevant parameters. Even when there are no pre-existing hidden variables to recover, and the unsupervised method is run "blindly", it is uninteresting to recover clusters describing the values to already known co-features. Instead, one would rather provide beforehand the unsupervised method with the information of the co-features' values and encourage it to recover new information from the data.

In order to take into account the effect of co-features on features, Yin and Li (2011) and Wytock and Kolter (2013) introduced the Conditional Gaussian Graphical Models (CGGM). Within this model, the average effect of the co-features is subtracted from the features using a "linear transition" matrix Θ, which leaves the features with only their residual behaviour. Both Yin and Li (2011) and Sohn and Kim (2012) worked with homogeneous populations, but the Hierarchical form of the CGGM was introduced by Chun et al. (2013) to study labelled heterogeneous populations, with heterogeneous effects of the co-features on the features. Recent works such as Huang et al. (2018) and Ou-Yang et al. (2019) have adapted the state of the art supervised joint Hierarchical GGM methods to the CGGM, in order to also recover elements of common structure across all classes. However, to the best of our knowledge, there has been no effort to make use of the CGGM in the unsupervised case to identify groups with distinct sparse correlation structures with common elements.

In this article, we introduce a Mixture of Conditional GGM that models the class-dependent effect of the co-features on the features with different transition parameters $(\Theta_k)_{k=1}^K$. We propose an Expectation-Maximisation (EM) procedure to estimate this model without prior knowledge of the class labels. This EM algorithm can be regularised with all the structure-inducing penalties introduced for the supervised joint Hierarchical CGGM. Hence, the recovered sparse precision matrices $(\Lambda_k)_{k=1}^K$ and corresponding conditional correlation graphs can present any of the desired form of common structure. Moreover, with an additional penalty term, we can also enforce sparse and common structure within the transition parameters Θ_k. We provide a very detailed computational scheme for our algorithm in the specific case of the Group Graphical Lasso (GGL) penalty of Danaher et al. (2014). Moreover, we make the code for our algorithm, as well as a toy example that reproduces some of the results of this paper, publicly available at: https://github.com/tlartigue/Mixture-of-Conditional-Gaussian-Graphical-Models.

Thanks to the inclusion of the co-features within the model, our EM algorithm is able to avoid trivial clusters correlated with the co-features' values, and instead unearths clusters providing new information on the population. Additionally, since our model takes into account heterogeneous effects of the co-features, our EM can handle the more complex scenarios, where the co-features act differently on the features in each sub-population.

A similar method was introduced by Kim (2016), who also uses a Conditional Gaussian Mixture model between co-features and features, as well as sparse GGM on the features. In their case however, the prediction of the features by the co-features takes centre stage. Indeed, the Mixture model acts as a Mixture of Experts, see Jordan and Jacobs (1994), that improves the prediction by allowing non-linear relations. Whereas in this article, the co-features are not meant to predict the features, but instead to make the clustering of the population more sensible. Another difference is that our algorithm performs at each M step a joint estimation of the model parameters in order to identify the common covariance structure between the different groups. Another domain of research, the "Finite Mixture Regression models" (FMR) DeSarbo and Cron (1988); Khalili and Chen (2007), exhibit some superficial similarities with the Mixture of CGGM, but are very different in scope. The FMR consist of several parallel linear regressions between co-features and features. Where the features are usually uni- or low-dimensional and their covariance structure is not in question. Prediction is once again the main focus. Any clustering derived from an FMR model has its different classes solely defined by different linear relations (different "slopes") between co-features and features. This is very different from our GGM approach, where a multidimensional feature vector is described with a hierarchical graphical model, and the co-feature are a tool to improve the clustering within the feature space.

We demonstrate the performance of our method on synthetic and real data. First with a 2-dimensional toy example, where we show the importance of taking into consideration the (heterogeneous) effects of co-features for the clustering. Then, in higher dimension, we demonstrate that our EM with Mixture of CGGM consistently outperforms, both in terms of classification and parameter reconstruction, the EM with a Mixture of GGM (used in Gao et al. (2016) and Hao et al. (2017)), as well as an improved Mixture of GGM EM, that takes into consideration a homogeneous co-feature effect. Finally, on real Alzheimer's Disease data, we show that our method is the better suited to recover clusters correlated with the diagnostic, from both MRI and Cognitive Score features.

2 Supervised Hierarchical GGM and CGGM

In this section, we summarise the whys and wherefores of Gaussian Graphical Modelling: the simple models for homogeneous populations, as well as the hierarchical models for heterogeneous populations. First, we explore the classical Gaussian Graphical Models techniques to describe a vector of features $Y \in \mathbb{R}^p$, then we discuss the Conditional Gaussian Graphical Models implemented in the presence of additional co-features $X \in \mathbb{R}^q$. For every parametric model, we call θ the full parameter, and p_θ the probability density function. Hence, in the example of a gaussian model $\theta = (\mu, \Sigma)$. For hierarchical models with K classes, we will have K parameters $(\theta_1, ..., \theta_K)$.

2.1 Basics of Hierarchical Gaussian Graphical Models

In the classical GGM analysis introduced by Dempster (1972), the studied features $Y \in \mathbb{R}^p$ are assumed to follow a Multivariate Normal distribution: $Y \sim \mathcal{N}(\mu, \Sigma)$. The average μ is often ignored and put to 0. With $\Lambda := \Sigma^{-1}$, the resulting distribution is:

$$p_\theta(Y) = (2\pi)^{-p/2} \left|\Lambda\right|^{1/2} exp\left(-\frac{1}{2} Y^T \Lambda Y\right) . \tag{1}$$

In this case $\theta = \Lambda$. Using the property that $corr(Y_u, Y_v | (Y_w)_{w \neq u,v}) = -\frac{(\Lambda)_{uv}}{\sqrt{(\Lambda)_{uu}(\Lambda)_{vv}}}$, the conditional correlation network is obtained using a sparse estimation of the precision matrix Λ. Heterogeneous populations, where different correlation networks may exist for each sub-population (or "class"), can be described with the Hierarchical version of the GGM (1). With K classes, Let $\theta := (\theta_1, ..., \theta_k)$ be the parameter for each class and $z \in \llbracket 1, K \rrbracket$ the categorical variable corresponding to the class label of the observation Y. With $\theta_k := \Lambda_k$ and z known, the Hierarchical density can be written:

$$\begin{aligned} p_\theta(Y|z) &= \sum_{k=1}^{K} \mathbb{1}_{z=k} p_{\theta_k}(Y) \\ &= \sum_{k=1}^{K} \mathbb{1}_{z=k} (2\pi)^{-p/2} \left|\Lambda_k\right|^{1/2} exp\left(-\frac{1}{2} Y^T \Lambda_k Y\right) . \end{aligned} \tag{2}$$

Mirroring the famous Graphical LASSO (GLASSO) approach introduced by Yuan and Lin (2007) and Banerjee et al. (2006) for homogeneous populations, many authors have chosen to estimate sparse $\widehat{\Lambda}_k$ as penalised Maximum Likelihood Estimator (MLE) of Λ_k. For $i = 1, ..., n$, let $Y^{(i)}$ be independent identically distributed (iid) feature vectors and $z^{(i)}$ their labels. These MLE are computed from the simple convex optimisation problem

$$\hat{\theta} = \underset{\theta}{\operatorname{argmin}} \; -\frac{1}{n} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathbb{1}_{z^{(i)}=k} \, ln \, p_{\theta_k}(Y^{(i)}) + pen(\theta) . \tag{3}$$

Where the convex penalty $pen(\theta)$ is usually designed to induce sparsity within each individual $\hat{\Lambda}_k$ as well as to enforce a certain common structure between the $\hat{\Lambda}_k$. This common structure is a desirable outcome when the different sub-populations are assumed to still retain core similarities. Following in the footsteps of Guo et al. (2011), most authors propose such a joint estimation of the matrices Λ_k. In the case of the penalised MLE estimation (3), the form of the resulting common structure is dependent on the penalty. For instance, Danaher et al. (2014) propose the "Fused Graphical LASSO"" and "Group Graphical LASSO" penalties that encourage shared values and shared sparsity pattern across the different Λ_k respectively. Likewise, Yang et al. (2015) propose another fused penalty to incentivise common values across matrices. With their node based penalties, Mohan et al. (2014) can encourage the recovery of common hubs in the graphs.

Remark 1 Within a hierarchical model, one can also take $\theta_k := (\mu_k, \Lambda_k)$, and adapt $p_{\theta_k}(Y)$ accordingly, since it is natural to allow each sub-population to have different average levels μ_k.

2.2 Conditional GGM in the presence of co-features

In some frameworks, additional variables, noted $X \in R^q$ and called "co-features" or "cofactors" can be observed alongside the regular features within the gaussian vector $Y \in \mathbb{R}^p$. In all generality, X can be a mix of finite, discrete and continuous random variables. In the GGM analysis, these co-features are not included as nodes of the estimated conditional correlation graph. Instead, they serve to enrich the conditioning defining each edge: in the new graph, there is an edge between the nodes Y_u and Y_v if and only if $cov(Y_u, Y_v | (Y_w)_{w \neq u,v}, X) \neq 0$. The Conditional Gaussian Graphical Models (CGGM) were introduced by Yin and Li (2011) and Sohn and Kim (2012) in order to properly take into account the effect of X on Y and easily identify the new conditional correlation network in-between the Y. They propose a linear effect, expressed by the conditional probability density function (pdf):

$$p_\theta(Y|X) := \frac{|\Lambda|^{\frac{1}{2}}}{(2\pi)^{\frac{p}{2}}} exp\left(-\frac{1}{2}(Y+\Lambda^{-1}\Theta^T X)^T \Lambda (Y+\Lambda^{-1}\Theta^T X)\right), \tag{4}$$

with $\Theta \in \mathbb{R}^{q \times p}$ and $\theta = \{\Lambda, \Theta\}$. In other words: $Y|X \sim \mathcal{N}\left(-\Lambda^{-1}\Theta^T X, \Lambda^{-1}\right)$. To simplify later equations, we define the function $L(Y, X, \Lambda, \Theta) := -\frac{1}{2}(Y + \Lambda^{-1}\Theta^T X)^T \Lambda (Y + \Lambda^{-1}\Theta^T X)$. Eq (4) can be re-written:

$$p_\theta(Y|X) = \frac{|\Lambda|^{\frac{1}{2}}}{(2\pi)^{\frac{p}{2}}} exp\Big(L(Y, X, \Lambda, \Theta)\Big).$$

Two main branches of CGGM exist, depending on whether the pdf of X is also modelled. In this work, we chose to impose no model on X. The lack of assumption on the density of X provides far more freedom than the joint gaussian assumption. In particular, X can have categorical and even deterministic components. This allows us to integrate any observed variables without restriction to the model.

To tackle heterogeneous populations, works such as Chun et al. (2013) have introduced the Hierarchical version of the CGGM pdf:

$$p_\theta(Y|X, z) := \sum_{k=1}^{K} \mathbb{1}_{z=k} \frac{|\Lambda_k|^{\frac{1}{2}}}{(2\pi)^{\frac{p}{2}}} exp\Big(L(Y, X, \Lambda_k, \Theta_k)\Big). \tag{5}$$

In particular, Huang et al. (2018) have adapted the penalised MLE (3) to the Hierarchical CGGM density for some of the most popular GGM penalties. With an iid observed sample $(Y^{(i)}, X^{(i)}, z^{(i)})_{i=1}^n$, the corresponding penalised CGGM MLE can be written;

$$\hat{\theta} = \underset{\theta}{\text{argmin}} - \frac{1}{n} \sum_{k=1}^{K} \sum_{i=1}^{n} \mathbb{1}_{z^{(i)}=k} \, ln \, p_{\theta_k}(Y^{(i)}|X^{(i)}) + pen(\theta). \tag{6}$$

Remark 2 To include a regular average value for Y, independent of the values of X, one can simply add a constant component equal to "1" in X.

3 Mixtures of CGGM for unlabelled heterogeneous population

In this section, we tackle the problem of an unlabelled heterogeneous population. We introduce a Mixture of Conditional Gaussian Graphical Model to improve upon the state of the art unsupervised methods by taking into consideration the potent co-features that can drive the clustering. We develop a penalised EM algorithm to both identify data clusters and estimate sparse, structured, model parameters. We justify that our algorithm is usable with a wide array of penalties and provide detailed algorithmic steps for the Group Graphical LASSO (GGL) penalty.

3.1 Presentation and motivation of the model

When the labels of a heterogeneous population are missing, supervised parameter estimation methods like (3) have to be replaced by unsupervised approaches that also tackle the problem of cluster discovery. When z is unknown, the Hierarchical model (2) can easily be replaced by a Mixture model with observed likelihood:

$$p_{\theta,\pi}(Y) = \sum_{k=1}^{K} \pi_k p_{\theta_k}(Y), \tag{7}$$

and complete likelihood:

$$p_{\theta,\pi}(Y, z) = \sum_{k=1}^{K} \mathbb{1}_{z=k} \pi_k p_{\theta_k}(Y). \tag{8}$$

Where $\pi_k := \mathbb{P}(z = k)$ and $\pi := (\pi_1, ..., \pi_K)$. Then, the supervised penalised likelihood maximisation (3) can be adapted into the penalised observed likelihood optimisation:

$$\hat{\theta}, \hat{\pi} = \underset{\theta,\pi}{\text{argmin}} - \frac{1}{n} \sum_{i=1}^{n} ln \left(\sum_{k=1}^{K} \pi_k \, p_{\theta_k} \left(Y^{(i)}\right) \right) + pen(\theta, \pi). \tag{9}$$

This is a non-convex problem, and authors such as Zhou et al. (2009) and Krishnamurthy (2011) have proposed EM algorithms to find local solutions to (9). For a more recent example, Hara et al. (2018) used such a Mixture of GGM with an l_1 regularisation on each class with each M step solved by Graphical LASSO, Banerjee et al. (2008); Friedman et al. (2008). Likewise, Fop et al. (2019) also consider problems of the form (9), solved with an EM algorithm, although their regularisation functions belong to another family entirely, that enforces sparsity on the covariance matrix instead of the precision matrix. However, all these works omit the common structure inducing penalties that are the signature of the supervised joint GGM methods. The works of Gao et al. (2016) and Hao et al. (2017) correct this by proposing EM algorithms that solve (9) for some of the joint-GGM penalties, such as the Fused and Group Graphical LASSO penalties.

By design, the EM algorithm must handle the cluster identification jointly with the mixture parameters estimation. The underlying assumption is that the different sub-populations can be identified as different clusters in the feature space. With real data, and especially medical data, this is generally untrue, as many factors other than the class label can have a larger impact on the position of the data points in the feature space. Even when there are no specific sub-populations to recover, and the EM is ran "blindly" in order to observe which data points are more naturally grouped together by the method, the unearthed clusters have every chance to be very correlated with very influential but trivial external variables, such as the age group or the gender. In order to guide the cluster discovery of the EM algorithm, we propose a Mixture of Conditional Gaussian Graphical Models with which the overbearing effect of trivial external variables can be removed. By placing all external observed variable into X, we define the Mixture of CGGM with its observed likelihood:

$$\begin{aligned} p_{\theta,\pi}(Y|X) &:= \sum_{k=1}^{K} \pi_k p_{\theta_k}(Y|X) \\ &= \sum_{k=1}^{K} \pi_k \frac{|\Lambda_k|^{\frac{1}{2}}}{(2\pi)^{\frac{p}{2}}} exp\Big(L(Y,X,\Lambda_k,\Theta_k)\Big) . \end{aligned} \tag{10}$$

Within this model, the position of each feature vector Y is corrected by its, class-dependent, linear prediction by the co-features X: $\mathbb{E}[Y|X,z=k] = -\Lambda_k^{-1}\Theta_k^T X$. In other words the "Mixture of Gaussians" type clustering is done on the residual vector $Y - \mathbb{E}[Y|X,z=k] = Y + \Lambda_k^{-1}\Theta_k^T X$. Hence, even if the co-features X have a class-dependent impact on the average level of the features Y, the Mixture of CGGM model is still able to regroup in the feature space the observations $Y^{(i)}$ that belong to the same class, $z^{(i)} = k$. We illustrate this dynamic in section 4.1.

Like the previous works on joint-GGM estimation, our goal is to estimate the parameters of model (10) with sparse precision matrices Λ_k and common structure across classes. Sparsity in the matrices Θ_k is also desirable for the sake of interpretation. Hence, we define the following penalised Maximum Likelihood problem:

$$\begin{aligned} \hat{\theta},\hat{\pi} = \underset{\theta,\pi}{\text{argmin}} \Bigg\{ -\frac{1}{n}\sum_{i=1}^{n} ln\left(\sum_{k=1}^{K}\pi_k\, p_{\theta_k}\left(Y^{(i)}|X^{(i)}\right)\right) \\ + pen(\theta,\pi) \Bigg\} . \end{aligned} \tag{11}$$

As with (9), this is a non-convex problem, and we define an EM algorithm to find local minima of the optimised function.

3.2 Penalised EM for the Mixture of CGGM

In this section, we provide the detailed steps of a penalised EM algorithm to find local solution of the non-convex penalised MLE (11) in order to estimate the parameters of the mixture model (10) with precision matrix sparsity as well as common structure. First we provide the different steps of the algorithm and justify that it can be run with a wide array of penalty functions. Then, we provide a detailed optimisation scheme for the Group Graphical Lasso (GGL) penalty specifically. The full code for this GGL-penalised CGGM EM can be found at https://github.com/tlartigue/Mixture-of-Conditional-Gaussian-Graphical-Models.

3.2.1 EM algorithm for Mixtures of CGGM

With n fixed $\left\{X^{(i)}\right\}_{i=1}^{n}$ and n iid observations $\left\{Y^{(i)}\right\}_{i=1}^{n}$ following the mixture density $p_{\theta,\pi}(Y|X)$ given in (10), the penalised observed negative log-likelihood to optimise is:

$$-\frac{1}{n}\sum_{i=1}^{n} ln\left(\sum_{k=1}^{K}\pi_k\, p_{\theta_k}\left(Y^{(i)}|X^{(i)}\right)\right) + pen(\theta,\pi) . \tag{12}$$

The EM algorithm is an iterative procedure updating the current parameter $(\theta^{(t)},\pi^{(t)})$ with two steps. For a Mixture model, the Expectation (E) step is:

$$\begin{aligned} p_{i,k}^{(t)} &:= \mathbb{P}_{\theta^{(t)},\pi^{(t)}}(z^{(i)} = k|Y^{(i)},X^{(i)}) \\ &= \frac{p_{\theta_k^{(t)}}(Y^{(i)}|X^{(i)})\pi_k^{(t)}}{\sum_{l=1}^{K} p_{\theta_l^{(t)}}(Y^{(i)}|X^{(i)})\pi_l^{(t)}} . \end{aligned}$$

More explicitly, by replacing $p_{\theta_k}(Y|X)$ by its formula (4):

$$(E) \quad p_{i,k}^{(t)} = \frac{|\Lambda_k|^{\frac{1}{2}}\, exp\left(\left(L(Y^{(i)},X^{(i)},\Lambda_k,\Theta_k)\right)\pi_k^{(t)}\right.}{\sum_{l=1}^{K}|\Lambda_l|^{\frac{1}{2}}\, exp\left(L(Y^{(i)},X^{(i)},\Lambda_l,\Theta_l)\right)\pi_l^{(t)}} . \tag{13}$$

Likewise, for Mixture models, the M step is:

$$\begin{aligned} (\theta,\pi)^{(t+1)} = \underset{\theta,\pi}{\text{argmin}} \Bigg\{ -\frac{1}{n}\sum_{k=1}^{K}\sum_{i=1}^{n} p_{i,k}^{(t)}\, ln\big(\pi_k p_{\theta_k}(Y^{(i)}|X^{(i)})\big) \\ + pen(\theta,\pi) \Bigg\} . \end{aligned}$$

Assuming that there is no coupling between π and θ in the penalty, i.e. $pen(\pi,\theta) = pen_\pi(\pi) + pen_\theta(\theta)$, then the two optimisations can be separated:

$$\begin{aligned}\theta^{(t+1)} = \underset{\theta}{\text{argmin}} \Bigg\{ - \frac{1}{n} \sum_{k=1}^{K} \sum_{i=1}^{n} p_{i,k}^{(t)} \, ln \, p_{\theta_k} (Y^{(i)} | X^{(i)}) \\ + pen_{\theta}(\theta) \Bigg\} , \\ \pi^{(t+1)} = \underset{\pi}{\text{argmin}} \Bigg\{ - \frac{1}{n} \sum_{k=1}^{K} \sum_{i=1}^{n} p_{i,k}^{(t)} \, ln \, \pi_k + pen_{\pi}(\pi) \Bigg\} .\end{aligned}$$

We denote the sufficient statistics $n_k^{(t)} := \sum_{i=1}^{n} p_{i,k}^{(t)}$, $S_{YY}^{k,(t)} := \frac{1}{n} \sum_{i=1}^{n} p_{i,k}^{(t)} Y^{(i)} Y^{(i)T}$, $S_{YX}^{k,(t)} := \frac{1}{n} \sum_{i=1}^{n} p_{i,k}^{(t)} Y^{(i)} X^{(i)T}$ and $S_{XX}^{k,(t)} := \frac{1}{n} \sum_{i=1}^{n} p_{i,k}^{(t)} X^{(i)} X^{(iT)}$. Then, the M step can be formulated as:

$$(M) \quad \begin{aligned}\theta^{(t+1)} = \underset{\theta}{\text{argmin}} \Bigg\{ \sum_{k=1}^{K} \Bigg(\Bigg\langle \Lambda_k, \frac{S_{YY}^{k,(t)}}{2} \Bigg\rangle + \Big\langle \Theta_k, S_{YX}^{k,(t)} \Big\rangle \\ + \Bigg\langle \Theta_k \Lambda_k^{-1} \Theta_k^T, \frac{S_{XX}^{k,(t)}}{2} \Bigg\rangle \\ - \frac{n_k^{(t)}}{n} ln(|\Lambda_k|) \Bigg) + pen_{\theta}(\theta) \Bigg\} , \\ \pi^{(t+1)} = \underset{\pi}{\text{argmin}} \Bigg\{ - \sum_{k=1}^{K} \frac{n_k^{(t)}}{n} \, ln \, \pi_k + pen_{\pi}(\pi) \Bigg\} .\end{aligned} \tag{14}$$

The E step in Eq (13) is in closed form. Likewise, the optimisation problem that defines $\pi^{(t+1)}$ in Eq (14) is very simple, and with any reasonable penalty pen_{π}, solving it is not issue. For instance, usual regularisation of the penalty take either the form $pen_{\pi}(\pi) = 0$ or $pen_{\pi}(\pi) \propto \sum_k \log(\pi_k)$, which both result in closed form solutions for $\pi^{(t+1)}$. The M step update of the model parameter θ is a harder optimisation problem. However, as is usually the case with EM algorithms, the update of θ in the M step (14) has the same form as the supervised MLE (6). As a result, when the supervised Hierarchical CGGM problem (6) is tractable, then our M step is tractable as well. Conveniently, this supervised CGGM estimation problem has already been studied and solved in Huang et al. (2018) for the Group Graphical Lasso (GGL) penalty of Danaher et al. (2014). Furthermore, Huang et al. (2018) show that the supervised negative log-likelihood is a convex function of θ. As a consequence, as long as the penalty pen_{θ} is convex and differentiable, then our M step (14) is a convex optimisation problem with a differentiable objective function. Since those problems are solvable with gradient descent algorithm, this means that the proposed EM algorithm is tractable for a very wide array of penalties pen_{θ}. In particular, this includes all the convex differentiable penalties, which covers a lot of popular GGM penalties for sparsity and joint structure, such as the ones from the works of Danaher et al. (2014), Mohan et al. (2014) and Yang et al. (2015) for instance.

In order to provide an algorithm with more specific and detailed steps, we consider in the rest of the section the special case of the GGL penalty. The GGL penalty was noticeably used in the supervised case by Huang et al. (2018), who proposed a proximal gradient algorithm. Likewise, we can use a proximal gradient algorithm to compute the M step (14) of our EM algorithm.

3.2.2 Proximal gradient algorithm to solve the M step with the GGL penalty

The GGL penalty, introduced in Danaher et al. (2014) and adapted to the hierarchical CGGM by Huang et al. (2018), can be written:

$$\begin{aligned}pen_{\theta}(\theta) := \sum_{1 \leq i \neq j \leq p} \left(\lambda_1^{\Lambda} \sum_{k=1}^{K} \left| \Lambda_k^{(ij)} \right| + \lambda_2^{\Lambda} \sqrt{\sum_{k=1}^{K} \left(\Lambda_k^{(ij)} \right)^2} \right) \\ + \sum_{\substack{1 \leq i \leq q \\ 1 \leq j \leq p}} \left(\lambda_1^{\Theta} \sum_{k=1}^{K} \left| \Theta_k^{(ij)} \right| + \lambda_2^{\Theta} \sqrt{\sum_{k=1}^{K} \left(\Theta_k^{(ij)} \right)^2} \right) .\end{aligned} \tag{15}$$

Unlike in Huang et al. (2018), where$\lambda_1^{\Lambda} = \lambda_1^{\Theta}$ and $\lambda_2^{\Lambda} = \lambda_2^{\Theta}$, we use different levels of penalisation for the parameters Λ and Θ, since both their scales and their desired sparsity level can be very different. This penalty borrows its design from the Group Lasso, see Yuan and Lin (2006), where the l_1 norm induces individual sparsity of each coefficient, and the l_2 induces simultaneous sparsity of groups of coefficients. In Eq. (15), for each pair (i, j) belonging to the relevant space, $\left\{ \Lambda_k^{(ij)} \right\}_{k=1}^{K}$ constitutes a group that can be entirely put to 0. This incites the algorithm to set a certain matrix coefficient to 0 over all K classes. These common zeros constitute the common structure sought after by the GGL approach. In our CGGM case, the same can be said for the group $\left\{ \Theta_k^{(ij)} \right\}_{k=1}^{K}$.

Regarding the theoretical analysis, we underline that the l_2 part of the penalty is not separable in a sum of K different penalties, which forces a joint optimisation problem to be solved, even in the supervised framework.

We detail here how to solve the M step (14) with $pen_{\theta}(\theta)$ defined as in Eq (15). We assume, as usual, that the optimisation in π is both independent from the optimisation in $\theta = \{\Lambda_k, \Theta_k\}_{k=1}^{K}$ and trivial. The function to minimise in θ at the M step is:

$$\begin{aligned}f(\theta) := \sum_{k=1}^{K} \Bigg(- \frac{n_k^{(t)}}{n} ln(|\Lambda_k|) + \Big\langle \Lambda_k, S_{YY}^{k,(t)} \Big\rangle + \Big\langle 2\Theta_k, S_{YX}^{k,(t)} \Big\rangle \\ + \Big\langle \Theta_k \Lambda_k^{-1} \Theta_k^T, S_{XX}^{k,(t)} \Big\rangle \Bigg) + pen_{\theta}(\theta) .\end{aligned}$$

As shown in Huang et al. (2018), this function is convex and infinite on the border of its set of definition and as a unique global minimum. We note $f(\theta) =: g(\theta) + pen_\theta(\theta)$ for the sake of simplicity. The proximal gradient algorithm, see Combettes and Pesquet (2011), is an iterative method, based on a quadratic approximation on $g(\theta)$, that benefits from theoretical convergence guarantees. If $\theta^{(s-1)}$ is the current state of the parameter within the proximal gradient iterations, then the next stage, $\theta^{(s)}$, is found by optimising the approximation:

$$\begin{aligned} f\left(\theta^{(s)}\right) &= f\left(\theta^{(s-1)} + \theta^{(s)} - \theta^{(s-1)}\right) \\ &\approx g\left(\theta^{(s-1)}\right) + \nabla g\left(\theta^{(s-1)}\right)^T . \left(\theta^{(s)} - \theta^{(s-1)}\right) \\ &\quad + \frac{1}{2\alpha}\left\|\theta^{(s)} - \theta^{(s-1)}\right\|_2^2 + pen_\theta\left(\theta^{(s)}\right) \\ &\equiv \frac{1}{2\alpha}\left\|\theta^{(s)} - \left(\theta^{(s-1)} - \alpha\nabla g\left(\theta^{(s-1)}\right)\right)\right\|_2^2 \\ &\quad + pen_\theta\left(\theta^{(s)}\right) . \end{aligned} \tag{16}$$

Where we removed in the last line the constants irrelevant to the optimisation in $\theta^{(s)}$ and α denotes the step size of the gradient descend. Note that we use the superscript (s) to indicate the current stage of the proximal gradient iteration, to avoid confusion with the superscript (t) used for the EM iterations (which are one level above). We underline that, in addition to $g(\theta)$ itself, the second order term in the Taylor development of $g(\theta)$ is also approximated. Using $\frac{1}{2\alpha}\left\|\theta^{(s)} - \theta^{(s-1)}\right\|_2^2$ instead of $\frac{1}{2}\left(\theta^{(s)} - \theta^{(s-1)}\right)^T .H_g\left(\theta^{(s-1)}\right) . \left(\theta^{(s)} - \theta^{(s-1)}\right)$ spares us from computing the Hessian $H_g\left(\theta^{(s-1)}\right)$ and simplifies the calculations to come. The approximated formulation in Eq (16) leads to the definition of the proximal optimisation problem:

$$prox_\alpha(x) := \underset{\theta}{\operatorname{argmin}} \frac{1}{2\alpha}\|\theta - x\|_2^2 + pen_\theta(\theta) . \tag{17}$$

So that the proximal gradient step can be written:

$$\theta^{(s)} = prox_{\alpha_s}\left(\theta^{(s-1)} - \alpha_s\nabla g\left(\theta^{(s-1)}\right)\right) . \tag{18}$$

Where the step size α_s is determined by line search. The usual proximal gradient heuristic is to take a initial step size α^0, a coefficient $\beta \in (0,1)$, and to reduce the step size, $\alpha \longleftarrow \beta\alpha$, as long as:

$$\begin{aligned} g\left(\theta^{(s-1)} - \alpha G_\alpha(\theta^{(s-1)})\right) >& g\left(\theta^{(s-1)}\right) \\ & - \alpha\nabla g\left(\theta^{(s-1)}\right)^T .G_\alpha(\theta^{(s-1)}) \\ & + \frac{\alpha}{2}\left\|G_\alpha(\theta^{(s-1)})\right\|_2^2 , \end{aligned}$$

with $G_\alpha\left(\theta^{(s-1)}\right) := \frac{\theta^{(s-1)} - prox_\alpha\left(\theta^{(s-1)} - \alpha\nabla g\left(\theta^{(s-1)}\right)\right)}{\alpha}$ the generalised gradient.

To apply the proximal gradient algorithm, we need to be able to solve the proximal (17) with the CGGM likelihood and the GGL penalty. Thankfully, Danaher et al. (2014) found an explicit solution to this problem in the GGM case, which Huang et al. (2018) adapted to the CGGM. The proximal optimisation is separable in Λ and Θ, and the solutions $\Lambda^{(prox)}$ and $\Theta^{(prox)}$ share the same formula. As a result, we use D as a placeholder name for either Λ or Θ, i.e. depending on the context either $D_k^{ij} = \Lambda_k^{ij}$ or $D_k^{ij} = \Theta_k^{ij}$. Let S be the soft thresholding operator: $S(x, \lambda) := sign(x)\, max(|x| - \lambda, 0)$, and $\widetilde{D}_{k,\alpha}^{ij} := D_k^{ij,(s-1)} - \alpha\frac{\partial g}{\partial D_k^{ij}}\left(\theta^{(s-1)}\right)$. The solution of (17), with $x = \theta^{(s-1)} - \alpha\nabla g\left(\theta^{(s-1)}\right)$, is given coefficient-by-coefficient in Eq (19):

$$\begin{aligned} D_k^{ij,(prox)} =& S\left(\widetilde{D}_{k,\alpha}^{ij}, \lambda_1^D\alpha\right) \\ & \times max\left(1 - \frac{\lambda_2^D\alpha}{\sqrt{\sum_k S(\widetilde{D}_{k,\alpha}^{ij}, \lambda_1^D\alpha)^2}}, 0\right) . \end{aligned} \tag{19}$$

Note that the partial derivatives $\frac{\partial g}{\partial D_k^{ij}}\left(\theta^{(s-1)}\right)$, necessary to get $\widetilde{D}_{k,\alpha}^{ij}$, are easily calculated in closed form from the likelihood formula. With the proximal problem (17) and the line search easily solvable, the proximal gradient steps can be iterated until convergence to find the global minimum of $f(\theta)$. With $f(\theta)$ optimised, the M step (14) is solved.

4 Experiments

In this section, we demonstrate the performances of our EM with Mixture of CGGM. First on a visual toy example in 2 dimension, then on a higher dimensional synthetic example and finally on real Alzheimer's Disease data. We compare the Mixture of CGGM to the regular Mixture of GGM which ignores co-features and to a Mixture of GGM that assumes a uniform linear effect of the co-features on the features.

4.1 An illustration of co-features with class-dependent effect

In this section, we present a simple visual example to illustrate the importance of taking into account heterogeneous co-feature effects. We show that even with a single binary co-feature, and with low dimensional features, the state of the art unsupervised GGM techniques are greatly disrupted by the co-features. Whereas our EM with Mixture of CGGM (which we call "Conditional EM" or "C-EM") achieves near perfect classification.

Under the Mixture of Gaussians (MoG) model, the observed data, $Y \sim \sum_{k=1}^K \pi_k \mathcal{N}(\mu_k, \Sigma_k)$, belongs to K classes which can directly be represented as K clusters in the feature space

$\mathbb{R}^p$. Each cluster centered around a centroid at position μ_k and with an ellipsoid shape described by Σ_k. However, when there exists conditioning variables $X \in \mathbb{R}^q$ that have an effect on Y, this geometric description becomes more complex. Typically, the value of Y could depend linearly on the value of X, with $\mathbb{E}\left[Y|X, z = k\right] = \beta_k^T X$ for some $\beta_k \in \mathbb{R}^{p \times q}$. In this case, the average position in class k is not a fixed μ_k but a function of X. If X contains categorical variables, this creates as many different centroid positions as there are possible category combinations in X. The number of these *de facto* clusters geometrically increases with the dimension q, which deters from simply running a clustering method with an increased number of clusters K' to identify all of them. Moreover, if X contains continuous variables, there is a continuum of positions for the centroid, not a finite number of *de facto* clusters. If X mixes the two types of variables, the two effects coexist. This shatters any hope to run a traditional MoG-based EM clustering algorithm, since its success is heavily dependent on its ability to identify correctly the K distinct cluster centroids μ_k.

Since the X are observed, a possible solution is to run the linear regression $\hat{Y} = \hat{\beta} X$ beforehand, and run the EM algorithm on the residual $Y - \hat{Y}$ to remove the effect of X. This is what we call the "residual EM" or "residual Mixture of GGM". However this does not take into account the fact that this effect can be different for each class k, $\beta_1 \neq \beta_2 \neq ... \neq \beta_K$. Since the label is not known beforehand in the unsupervised context, the linear regression $\hat{Y} = \hat{\beta} X$ can only be run on all the data indiscriminately, hence is insufficient in general. On the other hand, the hierarchical CGMM (5), which verifies: $\mathbb{E}\left[Y|X, z = k\right] = -\Lambda_k^{-1} \Theta_k^T X$, is designed to capture heterogeneous co-feature effects. We design a simple experiment to substantiate this intuition.

In this example, $Y \in \mathbb{R}^2$, $X \in \{-1, 1\}$ and $z \in \{1, 2\}$. $Y|X, z$ follows the hierarchical conditional model of (5). In this simple case, this can be written as $Y = \left(\beta_1 X + \epsilon_1\right) \mathbb{1}_{z=1} + \left(\beta_2 X + \epsilon_2\right) \mathbb{1}_{z=2}$. With $\mathbb{P}\left(z = 1\right) = \mathbb{P}\left(z = 2\right) = 0.5$, $\epsilon_1 \sim \mathcal{N}(0, \Lambda_1^{-1})$ and $\epsilon_2 \sim \mathcal{N}(0, \Lambda_2^{-1})$. A typical iid data sample $(Y_{i=1}^{(i)})^n$ is represented on the left sub-figure of Fig. 1. The hidden variable z is represented by the colour (blue or orange). The covariance of the cluster changes with z. The observable co-feature X is represented by the shape of the data point (dot or cross). The centroid of the cluster is translated differently according to the values of X and z. It is clear from the figure that a Mixture of Gaussians model with $K = 2$ cannot properly separate the blue and orange points in two clusters. Indeed, on the right sub-figure of Fig. 1, we observe the final state of an EM that fits a Mixture of Gaussians on Y. The two recovered clusters are more correlated with the co-feature X than the hidden variable z. However, this method did not take advantage of the knowledge of the co-feature X. As previously mentioned, one could first subtract the effect of X from Y before running the EM. On the left sub-figure of Fig. 2, we represent the residual data $\tilde{Y} := Y - \hat{\beta} X$. Where $\hat{\beta}$ is the Ordinary Least Square estimator of the linear regression between X and Y over all the dataset ($\hat{\beta} \approx \frac{\beta_1 + \beta_2}{2}$ if n is large enough). Since the linear effect between X and Y is not uniform over the dataset, but class dependent, the correction is imperfect, and the two class clusters remain hardly separable. This is why the residual EM, that fits a Mixture of GGM on $\tilde{Y}$ is also expected to fail to identify clusters related to the hidden variable. Which is shown by the right sub-figure of Fig. 2, where we see a typical final state of the residual EM.

On the leftmost sub-figure of Fig. 3, we display the proper correction for the co-features' effect $\tilde{Y}' = Y - \beta_1 X \mathbb{1}_{z=1} - \beta_2 X \mathbb{1}_{z=2} = \epsilon_1 \mathbb{1}_{z=1} + \epsilon_2 \mathbb{1}_{z=2}$. Under this form, a Mixture of Gaussian can separate the data by colour. This is precisely the kind of translation that each data point undergoes within a Hierarchical CGGM. Hence a Mixture of CGGM can succeed in identifying the hidden variable z, provided that it estimates correctly the model parameters. To illustrate this point, the two next sub-figures in Fig. 3 represent the same final state of the EM fitting a Mixture of CGGM on Y. The middle sub-figure represents $\tilde{Y}'$ as well as the two estimated centered distributions $\mathcal{N}(0, \widehat{\Lambda}_k^{-1})$ for $k = 1, 2$. We can see the two formally identified clusters after removing the effect of X. The rightmost sub-figure represents the original data Y as well as the four estimated distributions $\mathcal{N}(\pm \widehat{\Sigma}_k \widehat{\Theta}_k^T, \widehat{\Lambda}_k^{-1})$ for $k = 1, 2$. The four *de facto* clusters present in the data Y before removing the effect of X are well estimated by the method.

We confirm these illustrative results by running several simulations. Under the previously introduced framework, we generate 50 datasets with $n = 500$ data points each. For each simulation, we make 10 random initialisations from which we run the three EMs: with GGM, residualised GGM or CGGM. Table 4.1 summarises the results of these simulations. We follow the errors made by the estimated class probabilities or "soft labels", $\widehat{\mathbb{P}}(z_i = k)$, which we call the "soft misclassification error", as well as the error made by the "hard labels", $\mathbb{1}_{\widehat{z}_i = k}$, which we call the "hard misclassification error". They can be expressed as $\frac{1}{2n} \sum_{i,k} \left| \mathbb{1}_{z_i = k} - \widehat{\mathbb{P}}(z_i = k) \right|$ and $\frac{1}{2n} \sum_{i,k} \left| \mathbb{1}_{z_i = k} - \mathbb{1}_{\widehat{z}_i = k} \right|$ respectively. We see that the Mixture of CGGM performs much better, with less than 10% of misclassification in average, while the two GGM methods are both above 40% of error, fairly close to the level of a random uniform classifier, 50%.

4.2 Experiments in high dimension

In this section, we perform a quantitative analysis of the algorithms in a higher dimension framework, where the matrix parameters Λ and Θ are more naturally interpreted as sparse networks. We confirm that the Mixture of Conditional Gaus-

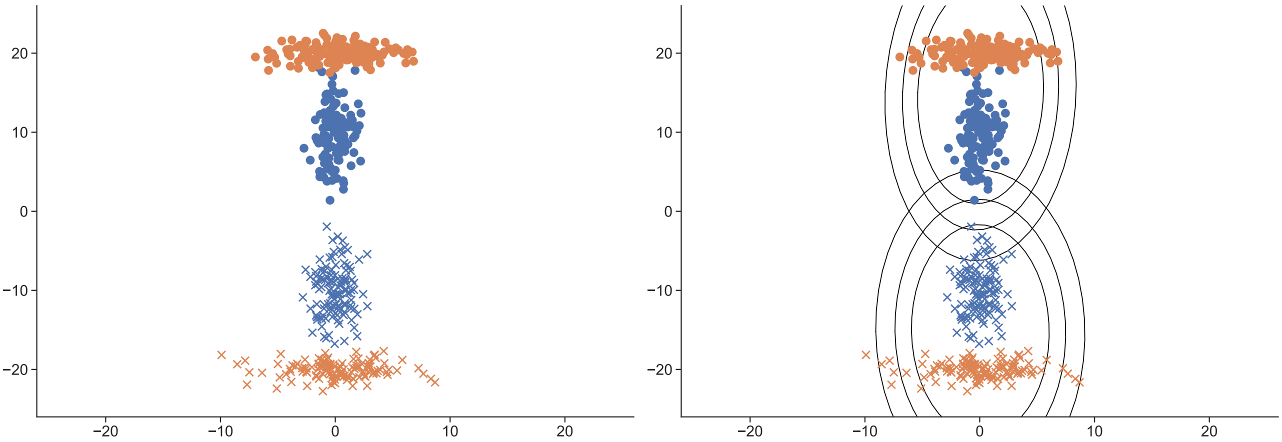

Fig. 1 (Left) Observed data Y in the 2D space. The observed conditioning variable X is binary. Data points with $X = -1$ are represented as crosses, and the ones with $X = 1$ are represented as dots. In addition, there is an unknown "class" variable z. Class 1 is in blue, class 2 in orange. $Y\,|X, z$ follows the hierarchical conditional model. As a result, the two classes (orange and blue) are hard to separate in two clusters. (Right) Typical clusters estimated by an EM that fits a GGM mixture on Y

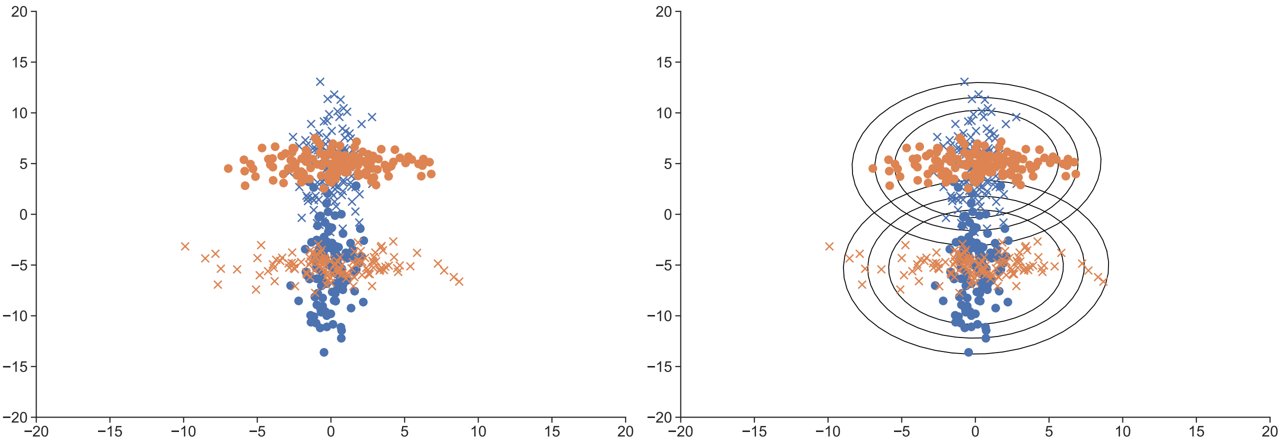

Fig. 2 (Left) Residual $\tilde{Y} = Y - \hat{\beta}Y$ data after taking into account the estimated effect of X. Since the effect had different intensities on class 1 and 2, only the average effect was subtracted, and two classes are still not well separated. (Right) Typical clusters estimated by the "residual EM", that fits a GGM mixture on $\tilde{Y}$

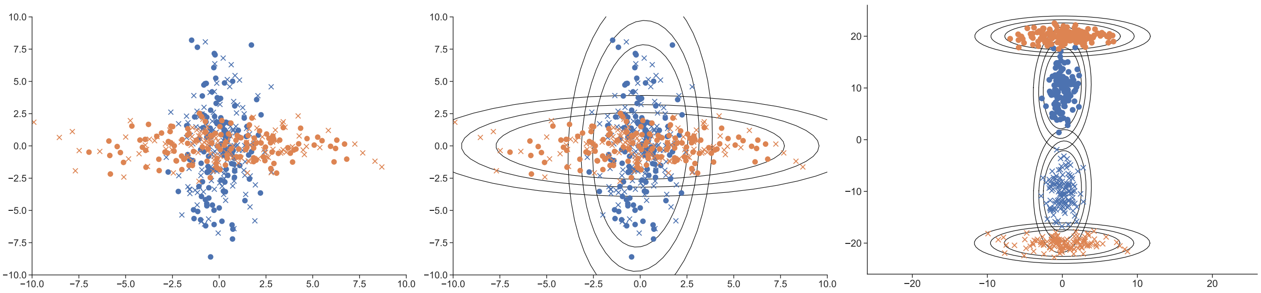

Fig. 3 (Left) Observations $\tilde{Y}' = Y - \beta_1 X \mathbb{1}_{z=1} - \beta_2 X \mathbb{1}_{z=2}$ exactly corrected for the class-dependent effect of X. In this state the two classes appear as two distinct clusters. The Conditional-EM is designed to transform the data in this manner. (Middle) One possible representation of the CEM results. The corrected observations $\tilde{Y}'$ are displayed alongside centered normal distributions with the two estimated covariance matrices: $\mathcal{N}(0, \widehat{\Lambda}_k^{-1})$. (Right) Another possible representation of the same CEM results. The original observations Y are displayed, alongside with the four *de facto* estimated distributions $\mathcal{N}(\pm\widehat{\Sigma}_k\widehat{\Theta}_k^T, \widehat{\Lambda}_k^{-1})$.

Table 1 Average and standard deviation of the misclassification error achieved on the 2-dimensional example with the EMs on the Mixture of GGM, the Mixture of GGM with residualised data, and the Mixture of CGG. The two GGM methods are close to the threshold of random classification (0.50), while the Mixture of CGGM is in average below 10% of error.

	EM GGM	EM res. GGM	EM CGGM
soft misclassif.	0.41 (0.11)	0.47 **(0.05)**	**0.08** (0.17)
hard misclassif.	0.41 (0.12)	0.46 **(0.06)**	**0.07** (0.17)

sian Graphical Models is better suited to take into account the heterogeneous effects of co-features on the graph.

For this experiment, the observed data follows a mixture model with $K = 3$ classes. Each class k has the probability weight $\pi_k = \frac{1}{3}$. An observation $(Y, X) \in \mathbb{R}^p \times \mathbb{R}^q$ belonging to the class k is described by the distribution: $Y|X \sim \mathcal{N}\left(-\Lambda_k^{-1}\Theta_k^T X, \Lambda_k^{-1}\right)$. In this example, the dimensions of Y and X are respectively $p = 10$ and $q = 5$. X contains four independent Rademacher variables ($= +1$ or -1) and a constant variable always equal to 1. The precision matrix $\Lambda_k \in \mathbb{R}^{p \times p}$ and the transition matrices $\Theta_k \in \mathbb{R}^{q \times p}$ are both sparse. For Λ, over the 45 off-diagonal coefficients, Λ_1 has 7 non-zero coefficients, Λ_2 has 16, and Λ_3 has 2. Resulting in an average 18.5% rate of sparsity among the off-diagonal coefficients. The sparsity pattern of the Λ_k are represented on Fig. 5, within the row of figures labelled "True". For Θ, over 50 possible coefficients, Θ_1 has 11 non-zero ones, Θ_2 has 14, and Θ_3 has 4. Resulting in an average sparsity rate of 19.3%. Additionally, there is a repeating pattern in the Θ_k: the 4 random components of X each affect (at least) the corresponding component among the first 4 of Y. That is to say that $\forall k, \forall i \leq 4, (\Theta_k)^{i,i} \neq 0$. The real Θ_k are represented on Fig. 6, within the row of figures labelled "True".

We run 20 simulations. A simulation consists of $n = 100$ generated data points. On these data points, we run the compared methods, all initialised with the same random parameters. For all simulations, we make 5 of these runs, each with a different random initialisation. We compared the same three algorithms as in section 4.1: the EM for the Mixture of GGM, the EM for the Mixture of GGM with average effect of X subtracted, and the EM applied to the Mixture of CGGM.

We follow three metrics to assess the method's success in terms of cluster recovery. The classification error (both soft and hard labels versions) and an "ABC-like" metric. The "ABC-like" metric is meant to assess how well each of the estimated solutions is able to replicate the observed data. Since each solution is the parameter of a probability distribution, at the end of each EM, we generate new data following this proposed distribution. Then, for each synthetic data point, we compute the distance to the closest neighbour among the real data points. These minimal distances constitute our "ABC-like" metric. Finally, we also compute the execution time of each EM, knowing that they all have the same stopping criteria. We represent on Fig. 4 the empirical distribution of these four metrics and we quantify with Table 2 the key statistics (mean, standard deviation, median) that characterise them. With $K = 3$ and balanced classes, a uniform random classifier would guess the wrong label 66.7% of the time. We observe that the two Mixture of GGM method are dangerously close to this threshold, with more than 50% hard misclassification. The EM on the Mixture of CGGM (C-EM) on the other hand, achieves a much better classification with less than 15% hard misclassifcation. This demonstrate that, even when faced with a more complex situation, in higher dimension, the Mixture of CGGM is better suited to correct for the effect of the co-features and discover the right clusters of data points. This also underlines once more the importance of allowing different values of the effect of X on Y for each class. Indeed, the residual Mixture of GGM - which took into account the average effect of X on Y over the entire population - was unable to achieve better performances than the EM that did not even use the co-features X. In terms of reconstruction of the observed data by the estimated model (ABC-like metric), we see that the synthetic data points generated from the estimated Mixture of CGGM model have closer nearest neighbours than the data points generated by the other estimated models. In addition to all these observations, the C-EM is also faster than the other two methods, reaching the convergence threshold faster.

Table 2 Average, standard deviation and median (below) of the four followed performance metrics over the 30 × 5 simulations. The best values are in **bold**. We can see that the classification performances with the Mixture of CGGM are much better than the two methods with Mixtures of GGM, and with faster computation times.

	EM GGM	EM GGM resid.	EM CGGM
soft misclassif.	0.56 **(0.03)** 0.57	0.51 **(0.03)** 0.51	**0.17** (0.05) **0.16**
hard misclassif.	0.55 (0.04) 0.56	0.50 **(0.03)** 0.49	**0.14** (0.06) **0.13**
ABC-like metric	5.57 **(0.09)** 5.58	4.64 (0.22) 4.64	**4.13** (0.14) **4.14**
runtimes	115 **(61)** 93	253 (137) 256	**58** (91) **16**

In addition to the cluster recovery, we can also assess the parameter reconstruction of each method. Since the three clustering methods estimate different parametric models over the data, they do not actually try to estimate the same parameters.

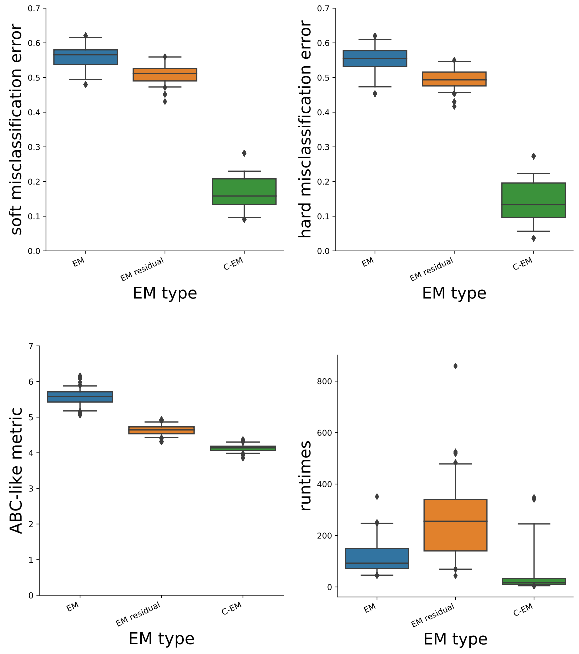

Fig. 4 Empirical distribution of several performance metrics measured over many simulations. The sample is made of 20 simulations with 5 different initialisations each. Three methods are compared. The EM and EM residual algorithms estimate a Mixture of GGM. The C-EM algorithm estimates a Mixture of CGGM. The C-EM is much better performing and faster. (Upper left) Soft mis-classification error $\left|\mathbb{1}_{z_i=k} - \widehat{\mathbb{P}}(z_i = k)\right|$. (Upper right) Hard mis-classification error $\left|\mathbb{1}_{z_i=k} - \mathbb{1}_{\widehat{z}_i=k}\right|$. (Bottom left) ABC-like metric. (Bottom right) Run time.

Regardless, all the methods still estimate a certain precision matrix Λ_k (conditional or not on the X depending on the model) of each sub-population that they identify. Hence, we also record the differences between the true Λ_k and their estimations by the methods. The two metrics followed are the Kullback–Leibler (KL) divergence Kullback and Leibler (1951) between the gaussian distribution $f_{\Lambda_k} \sim \mathcal{N}(0, \Lambda_k^{-1})$ and $f_{\widehat{\Lambda}_k} \sim \mathcal{N}(0, \widehat{\Lambda}_k^{-1})$, and the l_2 difference given by the Froebenius norm: $\left\|\Lambda_k - \widehat{\Lambda}_k\right\|_F^2$. In Table 3, we can check that, according to these metrics, the $\widehat{\Lambda}_k$ estimated by with the Mixture of CGGM are indeed a much better fit for the real Λ_k than the estimated matrices from the other models. This is expected, since the real Λ_k actually correspond to the CGGM model.

To illustrate the different level of success concerning the conditional correlation graph recovery, we display on Fig. 5 the conditional correlation matrix (i.e. the conditional correlation graph with weighted edges) estimated by each method. The three columns of figures correspond to the three sub-populations. The first two rows of figures are the matrices estimated by the two Mixtures of GGM methods, with and without residualisation with the co-features. The third row of figures correspond to the matrices estimated by the Mixture of CGGM. The final row displays the real conditional correlation matrices. We observe that the two Mixture of GGM recover way too many edges, with no particular fit with the real matrix. By contrast, the matrices from the CGGM Mixture exhibit the proper edge patterns, with few False Positive and False Negative. This is not an easy feat to achieve, since the method was run from a random initialisation on a totally unsupervised dataset, with heavily translated data points all over the 10 dimensional space. Moreover, the matrices in Fig. 5 all result from the inversion of the empirical covariance matrix, which is neither a very geometrical nor a very stable operation.

In Fig. 6, we represent the regression parameter $\widehat{\Theta}_k$ estimated by with the Mixture of CGGM alongside the real Θ_k. Once again, we see that the sparsity pattern is very well identified, with no False Positive. Moreover, in this case, there are also almost no False Negative, and all the edge intensities are correct. This is not a surprise. Indeed, the parameter Θ plays a huge role in the correct classification of the data, since it serves to define the expected position of each data point in the feature space (playing the part of the "average" parameter in Mixtures of GGM). Hence, a good estimation of Θ is mandatory to reach a good classification. Since the EM with Mixture of CGGM achieved very good classification results, it was expected that Θ would be well estimated.

Table 3 Average and standard deviation of the metrics describing the reconstruction of each precision matrix Λ_k. The matrices are consistently better reconstructed with the mixture of CGGM.

metric	class	EM GGM	EM res. GGM	EM CGGM
$KL(f_\Lambda, f_{\hat{\Lambda}})$	1	11.0 (3.0)	7.5 (6.8)	**0.8 (0.2)**
	2	10.3 (2.2)	8.5 (5.0)	**1.9 (0.3)**
	3	13.6 (2.5)	5.2 (2.3)	**3.4 (1.1)**
$\left\|\Lambda - \hat{\Lambda}\right\|_F^2$	1	39.2 (48.4)	44.2 (114)	**2.2 (0.8)**
	2	15.1 (12.2)	102 (73.9)	**6.6 (0.9)**
	3	14.2 (13.8)	15.1 (25.7)	**5.8 (4.0)**

4.3 Experiments on real data

In this section, we confirm our experimental observations with a real, high dimensional, Alzheimer's Disease dataset. We illustrate that the EM with Mixture of CGGM is better suited to identify clusters correlated with the diagnostic than the Mixture of GGM methods. We bring to light the effect of co-features such as the gender and age on the medical features.

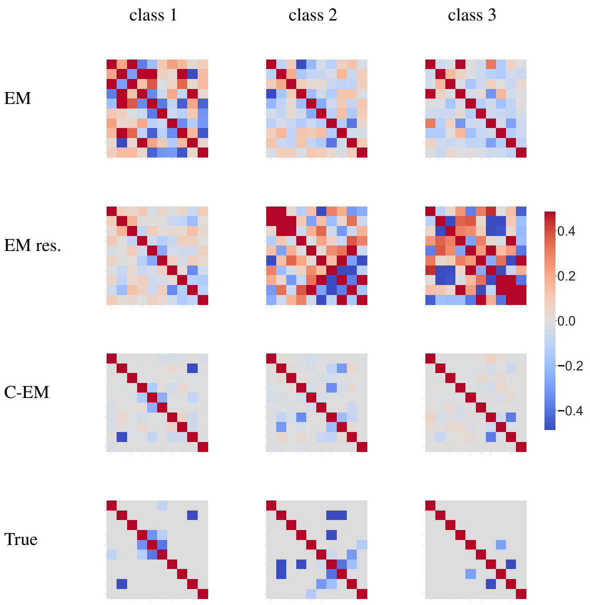

Fig. 5 Comparison between the several estimated and the true conditional correlation matrices for each sub-population. The three columns of figures correspond to the three sub-populations. The first two rows of figures are the matrices estimated by the two Mixture of GGM methods, with and without residualisation with the co-features. The third row of figures correspond to the matrices estimated by the Mixture of CGGM. The final row displays the real conditional correlation matrices. Unlike the two GGM-based methods, the Mixture of CGGM recovers correct edges with very few False Positives.

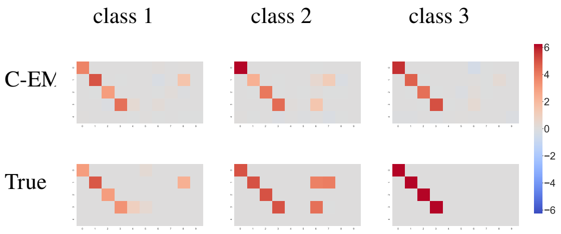

Fig. 6 Reconstruction of the Θ_k by the EM on the Mixture of CGGM. The three columns of figures correspond to the three sub-populations. Almost all the edges are right, with no False Positive and almost no False Negative. Moreover, the intensities are also mostly correct.

Our dataset is composed of 708 Alzheimer's Disease patients ("AD" patients) and 636 healthy patients ("Control" patients). Each patient makes several visits to the clinic (between 6 and 11 usually) at different moments of their lives. During each visit, the volumes of 10 predetermined cortical regions are measured with MRI, and tests are conducted to evaluate 20 different cognitive faculties (memory, language...). Instead of using this complex raw data, a longitudinal model is then estimated using the methodology of Schiratti et al. (2015). Such a model describes the evolution (or non-evolution) over time of the measured variables with a geodesic trajectory that lives within a Riemannian manifold. Using all patients, a "baseline" trajectory is first estimated. Then, each individual patient's trajectory is described by their deviation from this baseline. In our example, this deviation is fully encoded within 32 parameters (1 per observed variable + 2 additional parameters). The first model parameter is ξ, the time acceleration, which describes if, overall, the disease progresses faster or slower for the patient than for the baseline. The second model parameter is τ, the time shift, which describes whether the disease starts sooner or later for the patient than for the baseline. Finally, there is one parameter w_i for each of the 30 measured variables ($i = 1, ..., 10$ for the MRI regions and $i = 11, ..., 30$ for the cognitive scores). Each of these describes whether the variable in question degenerates faster or slower than the baseline, after having taken into account the patient's overall disease acceleration ξ.

For our experiments, these $p = 32$ parameters are the features in-between which we will draw graphical models: $Y := (\xi, \tau, (w_i)_{i=1,...,30})$. To illustrate the purpose of our approach: within this framework, a positive edge between two features w_i and w_j indicates that the corresponding features tend to co-degenerate together within the population. This is a crucial information for the understanding of the disease.

We include in the data three co-features that describe each patient and are often relevant in Alzheimer's Disease studies: the gender, the age at the first visit ("age baseline"), and the number of years of education. With the addition of the constant co-feature = 1, the vector of co-features is 4-dimensional, $X \in \mathbb{R}^4$. The $n = 1344$ (708 AD + 636 Control) patients constitute the unlabelled heterogeneous population. The "AD" patients were diagnosed with the Alzheimer's Disease, either from the start or after a few visits. Before running any method, The data (Y, X) is centred and normalised over the entire population.

We run unsupervised methods with $K = 2$ on this dataset in order to separate the data into 2 sub-populations and estimate a graphical model for each of them. As before, the three compared algorithms are EM, EM residual and C-EM. There is only one dataset available, hence, in order to check the stability of the results over several different runs, we implement a bootstrap procedure that only uses 70% of the data each time. We generate 10 such bootstrapped dataset. We initialise the algorithms with a KMeans on the $Y^{(i)}$ data points. Since KMeans is not deterministic, we try 3 different KMeans initialisations for each bootstrapped dataset. Like previously, for the sake of fairness, the EM and C-EM are always provided with the same initialisation, and the residual EM is initialised with a KMeans on the residual of Y after subtracting the prediction by the X, a more relevant initialisation for this method. We make all these runs with four different feature sets. First with no space shift variable $Y = \{\xi, \tau\}$, $p = 2$, then we add only the MRI space shifts $Y = \left\{\xi, \tau, (w_i)_{i=1,...,10}\right\}$, $p = 12$, then only the Cognitive Scores space shifts $Y = \left\{\xi, \tau, (w_i)_{i=1,...,10}\right\}$, $p = 22$, and finally, with all the features $Y = \left\{\xi, \tau, (w_i)_{i=1,...,30}\right\}$, $p = 22$. This is a total of 10 *bootstrapped datasets* × 3 *initialisations* ×

4 *feature sets* $= 120$ runs for each method. The classification results are summarised in Table 4. With two balanced classes, the classification error of a uniform random classifier is 50%. On the smallest dataset, $p = 2$, we can see that the discovered cluster are somewhat correlated with the diagnostic, with classification errors below 30%. The Mixture of GGM on the uniformly residualised data and the Mixture of CGGM achieve similar levels of error, they are both better than the regular Mixture of GGM. When the MRI features are added, all the discovered cluster become more correlated with the diagnostic. The regular Mixture of GGM achieves in average 16% of hard classification error, the residualised Mixture of GGM is at 11% of error, and the Mixture of CGGM even below, at 7%. The results with only the Cognitive Scores are very similar, simply a bit worse for every method. However, when both the MRI and Cognitive Scores feature are included, the performances of both GGM mixtures decrease, with both higher average error and higher variance. On the other hand, the Mixture of CGGM achieves here its best level of performance. This stability of the Mixture of CGGM's performance as the size of the feature set increases indicates that our model is the best suited to properly identify clusters correlated with the diagnostic in high dimension.

We analyse the estimated Mixture of CGGM parameters on the full feature set $p = 32$. First, since $\mathbb{E}[Y|X, z = k] = -\Lambda_k^{-1}\Theta_k^T X$ in the CGGM, we display on Fig. 7 the two estimated $\hat{\beta}_k := -\widehat{\Lambda}_k^{-1}\widehat{\Theta}_k$ (averaged over the bootstrap), as well as their difference. They play the role of linear regression coefficients in the model. The last column is the constant coefficient, while the first three are the gender, age baseline and years of education coefficients respectively. Since the data is centered, negative and positive values correspond to below average and above average values respectively. The cluster $k = 1$ is the one very correlated with the Control patients sub-population. Similarly, the cluster $k = 2$ is the one very correlated with the AD patients.

The most noticeable difference between the two $\hat{\beta}_k$ are the constant vectors, who have opposite effects on all features. In particular, the "AD cluster" is very correlated with high ξ, low τ, an earlier atrophy of the ventricles, as well as high space shift for the two logical memory tests (immediate and delayed). The exact opposite being true for the "Control cluster". These are the expected effects: a high ξ corresponds to a quickly progressing disease, and a low τ to an early starting disease.

The non-constant linear regression coefficients are also different between the clusters, although these differences are often in intensity and not in sign. In order to visualise more clearly the differences in intensity, we represent on the left-most sub-figure of Fig. 7, with the same conventions, the difference $\hat{\beta}_2 - \hat{\beta}_1$. In particular, within the AD cluster, we observe stronger positive effect of the Age at the first visit on the space shifts corresponding to the Amygdala, entorhinal cortex, hippocampus and parahippocampus cortex atrophies. On the contrary, there is a stronger positive effect of the education level on all the space shifts of MRI atrophies for the Control patients. The age at the first visit has a stronger negative impact on w_i corresponding to the scores self reported memory, language and visual spatial capacity for the AD patients, and a stronger negative impact on the two logical memory scores for the control patients. These differences, although less intense than the differences between the two constant terms, consequently impact the clustering, since without these three columns, the Mixture of CGGM would be equivalent to the less performing Mixture of GGM.

With the same conventions and scale, we display on Fig. 8, the linear regression coefficients estimated by maximum likelihood on the Control and AD patients, with the oracle knowledge of the diagnostic this time. Fig. 7 and 8 are very similar, in particular when it comes to their more potent coefficients. This shows that the C-EM, in addition to identifying clusters very correlated with the hidden diagnosis, also managed to recover the correct linear tendencies between features and co-features.

Finally, we display on Fig. 9 the average conditional correlation graphs estimated for the two clusters estimated by the Mixture of CGGM. Their only noticeable difference is the negative conditional correlation between ξ and τ in the "Control cluster", which is reversed in the "AD cluster". For the AD patients, this means that a disease that appears later tends to also progress faster, which is in line with medical observations. Apart from this edge, the rest of the connections are almost identical in-between clusters. This suggests that the, cluster dependent, prediction $\mathbb{E}_{\hat{\theta}_k}[Y^{(i)}|X^{(i)}, z = k] = -\widehat{\Sigma}_k\widehat{\Theta}_k^T X^{(i)}$ takes into account enough of the cluster-specific effects so that the remaining unexplained variance has almost the same form in both clusters. Hence, the conditional correlations pictured in these graphs correspond to very general effects, such as the positive correlations between related cognitive tests or areas of the cortex.

More strikingly, there are no conditional correlation between ξ or τ and any of the space shifts w_i. This as consequent medical implications, since it suggests that the earliness (τ) and speed (ξ) of the disease are conditionally independent from the succession of degradation that the patient's imagery and cognitive scores undergo. In other words, these graphs support the idea that the disease is the same regardless of whether it appears early/late and progresses fast/slowly.

As previously, we also estimated these conditional correlations graphs with the oracle knowledge of the diagnosis, they are displayed on Fig. 10. As with the transition matrices, the oracle covariance graphs exhibit the same links as the ones recovered by the C-EM. This once again highlights that the Mixture of CGGM is an appropriate model to handle both cluster identification and structure recovery.

For the sake of comparison, the model parameters (both $\hat{\beta}$ and the graph) estimated with a single class CGGM are displayed in Appendix.

5 Discussion and Conclusion

5.1 Discussion

In this work, we assumed that the number K of classes was known and fixed. This corresponds for instance to the case where there is prior medical expertise that provides in advance the number of classes. However, as with all unsupervised methods, the question of the empirical estimation of K is of great importance. Several approaches are possible. Since we have a Maximum Likelihood-type Estimator, a classical technique would be to use theoretical model selection criterion that penalises the number of degrees of freedom in the model. Famous examples include the Akaike information criterion (AIC), Akaike (1974), the Bayesian information criterion, Schwarz et al. (1978), or the corrected AIC (AICc), Hurvich and Tsai (1989). There are $df(K, p, q) = K(\frac{p(p+1)}{2} + qp)$ degrees of freedom in a Mixture of CGGM with K classes. After running an EM to completion for a value of K and getting the estimates $\hat{\theta}^K, \hat{\pi}^K$, the loss function associated with class number K is:

$$-\sum_{i=1}^{n} ln\left(\sum_{k=1}^{K} \hat{\pi}_k^K \, p_{\hat{\theta}_k^K}\left(Y^{(i)}|X^{(i)}\right)\right) + n \, pen(\hat{\theta}^K, \hat{\pi}^K) + \text{crit}(df(K, p, q), n)\,. \tag{20}$$

Where $\text{crit}(df, n) = df$ for AIC, $\text{crit}(df, n) = df \frac{ln\, n}{2}$ for BIC and $\text{crit}(df, n) = df + \frac{df\,(df+1)}{n-df-1}$ for AICc. A preferred $\hat{K}$ is then chosen as the one to minimise (20) among a predefined grid of different K. This is notably the approach adopted by Kim (2016). Since this model selection technique requires one full run of the EM algorithm for each value of K, it can be computationally demanding. Moreover, the empirical validity of such theoretical criteria is not always guaranteed for all problems, see for instance Lee and Ghosh (2009) for an such an empirical analysis.

As a result, for EM algorithms like ours, one could consider a more data-driven selection procedure such as the *Robust EM*, see Figueiredo and Jain (2002); Yang et al. (2012). *Robust EM* starts with a high number of classes that decreases and converges along the optimisation thanks to an entropy penalty term. However, this method makes each run of the EM more computationally demanding and can lead to very long execution times.

The relative model selection performances and computational burdens of these two different procedure will be studied in future works.

One can also question the heterogeneity of the effect of the co-features, and wonder how the model fares in the presence of homogeneous co-feature effects. Indeed, in our empirical study we compare three models : (i) the Mixture of GGM which does not take into account the co-features; (ii) the Mixture of GGM with prior residualisation, which assume an homogeneous effect of the co-features across all hidden sub-populations; and (iii) the Mixture of CGGM, which allows the co-feature to have heterogeneous effects on the hidden sub-population. However, one can wonder how is the Mixture of CGGM able to describe, and detect, the situation where the/several co-features have a homogeneous effect on the/some features? Within the Mixture of CGGM, the heterogeneity assumption is translated as extra degrees of freedom. The homogeneous case is included in this model: it corresponds to the scenario where coefficients have equal values across classes. When it comes to the estimation procedure, one way to enforce truly equal values is to replace the Group Graphical Lasso (GGL) penalty used in our examples by the Fused Graphical Lasso (FGL) penalty. Like GGL, FGL is one of the penalties proposed for the supervised Hierarchical GGM estimation in Danaher et al. (2014). As such, it is one of the many penalties with which our EM is tractable. This penalty - thanks to terms of the form: $\sum_{l<k} \left|\Theta_l^{(ij)} - \Theta_k^{(ij)}\right|$ - is designed to equalise the coefficients whose estimated values across different classes are close. This allows the model to detect homogeneous effects of the co-features.

Another, more hands on, possibility that does not require the use of fused-type penalties is to conduct statistical testing during the optimisation procedure. In order to detect homogeneity, such tests would look for significantly close coefficients. Then, if applicable, constrain them to be equal from this point on, hence removing this degree of freedom from the model. Thanks to this reduced complexity of the resulting model, such an step could help make the procedure more stable in the presence of co-features with homogeneous effect.

Those approaches will also be tested as part of future works in order to evaluate the performances of Mixtures of CGGM in the case of homogeneous co-feature effects.

5.2 Conclusion

We introduced the Mixture of Conditional Gaussian Graphical Models in order to guide the cluster discovery when estimating different Gaussian Graphical Models for an unlabelled heterogeneous population in the presence of co-features. We motivated its usage to deal with the potential inhomogeneous and class-dependent effect of the co-features on the observed data that would otherwise disrupt the clustering effort. To estimate our Mixture model, we proposed

Table 4 Recovery of the diagnostic labels (AD or control) with unsupervised methods on real longitudinal data. The three compared methods are the EM, EM residual (both GGM) and the C-EM (CGGM). Four different feature sets are tried: only $\{\tau, \xi\}$, adding the MRI space shift coefficients w_i, adding the Cognitive Score (CS) space shift coefficients w_i, and adding both the MRI and CS space shift coefficients. The table presents the average and standard deviation of the misclassification error over 10 bootstrap iteration, with 5 different KMeans initialisation each. The best results are in **bold**.

	metric	EM	EM resid.	C-EM
no CS, no MRI	soft misclassif.	0.31 (0.02)	0.22 (0.03)	**0.21 (0.01)**
$p = 2$	hard misclassif.	0.31 (0.03)	**0.18** (0.05)	0.19 **(0.01)**
only MRI	soft misclassif.	0.15 (0.01)	0.13 (0.01)	**0.08 (0.01)**
$p = 12$	hard misclassif.	0.12 (0.01)	0.10 (0.01)	**0.07 (0.01)**
only CS	soft misclassif.	0.17 (0.02)	0.15 (0.02)	**0.09 (0.01)**
$p = 22$	hard misclassif.	0.14 (0.03)	0.13 (0.03)	**0.08 (0.01)**
CS and MRI	soft misclassif.	0.24 (0.09)	0.17 (0.04)	**0.08 (0.01)**
$p = 32$	hard misclassif.	0.21 (0.10)	0.15 (0.05)	**0.07 (0.01)**

a penalised EM algorithm ("Conditional EM" or "C-EM") compatible with a wide array of penalties. Moreover, we provided detailed algorithmic steps in the specific case of the popular Group Graphical LASSO penalty, and made the corresponding code publicly available. Then, we demonstrated the interest of the method with experiments on synthetic and real data. First, we showed on a toy example - with a 2-dimensional feature space and a 1-dimensional co-feature - that the regular Mixture of GGM methods were inadequate to deal with even the most simple in-homogeneous co-feature. We confirmed on a more complex simulation, in higher dimension, that Mixtures of CGGM could identify much better the clusters in the feature space, and recover the actual GGM structure of the data. Finally, we tested all the methods on a real data set, with longitudinal model parameters describing the evolution of several Alzheimer's Disease patients. We demonstrated that our method was the best at identifying the diagnostic with an unlabelled dataset. We unearthed some in-homogeneous effect of co-features on the longitudinal parameter and recovered the conditional correlation graphs by cluster. These graphs hint at a conditional independence between the earliness and speed of the disease and the order in which the many degradation appear. This hypothesis will be tested in future studies.

Appendix: Single class CGGM on the real data

In this appendix, we take a look at the parameters (averaged over several bootstrap folds) estimated by fitting single CGGM on the real data. On Fig. 11, we display both the estimated $\hat{\beta} = -\widehat{\Sigma}\widehat{\Theta}$ between X and Y and the estimated conditional correlation graph in-between the components of Y. The constant term in $\hat{\beta}$ is 0 since the data is overall centred. Other than that, the coefficient intensities appear to be weaker than in the multi-class parameters. The conditional correlation graph on the other hand displays the negative correlation between disease earliness τ and speed ξ that was characteristic of the Control patients on Fig. 9 and 10. This is despite the Controls ($n = 636$) being slightly less numerous than the AD ($n = 708$) patients on this dataset.

References

Akaike H (1974) A new look at the statistical model identification. IEEE transactions on automatic control 19(6):716–723

Banerjee O, Ghaoui LE, d'Aspremont A, Natsoulis G (2006) Convex optimization techniques for fitting sparse gaussian graphical models. In: Proceedings of the 23rd international conference on Machine learning, ACM, pp 89–96

Banerjee O, Ghaoui LE, d'Aspremont A (2008) Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. Journal of Machine learning research 9(Mar):485–516

Chun H, Chen M, Li B, Zhao H (2013) Joint conditional gaussian graphical models with multiple sources of genomic data. Frontiers in genetics 4:294

Combettes PL, Pesquet JC (2011) Proximal splitting methods in signal processing. In: Fixed-point algorithms for inverse problems in science and engineering, Springer, pp 185–212

Danaher P, Wang P, Witten DM (2014) The joint graphical lasso for inverse covariance estimation across multiple classes. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 76(2):373–397

Dempster AP (1972) Covariance selection. Biometrics pp 157–175

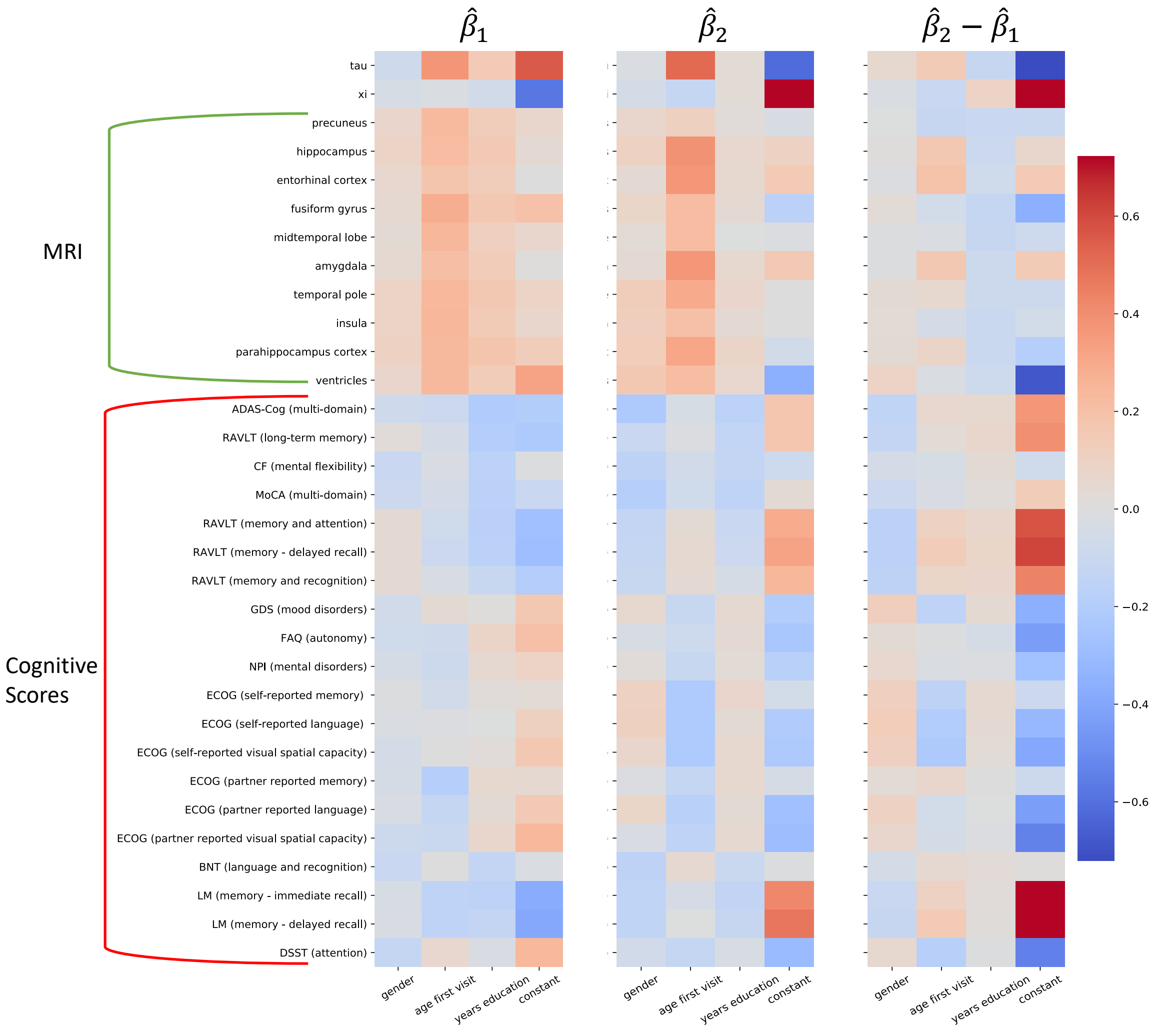

Fig. 7 Average of $\widehat{\beta}_k := -\widehat{\Sigma}_k \widehat{\Theta}_k^T$ over 10 bootstrap sampling of the data. For each bootstrapped dataset, $\widehat{\Sigma}_k$ and $\widehat{\Theta}_k$ are estimated with the C-EM. We make 3 of these C-EM runs, each starting from a different KMeans initialisation of the labels. This results in $10 \times 3 = 30$ different estimates of $\widehat{\beta}_k$ to average over. (Left) $\widehat{\beta}_1$, the cluster $k = 1$ is always very correlated with the **Control patients** sub-population (less than 10% deviation). (Middle) $\widehat{\beta}_2$, the cluster $k = 2$ is likewise very correlated with the **AD patients**. In each figure, the last column is the constant coefficient. The largest inter-cluster differences are between the two constant terms. However there are some noticeable difference on the other regression coefficients as well.(Right) Average $\widehat{\beta}_2 - \widehat{\beta}_1$ over the 30 bootstrap runs of the C-EM. Here, the **differences in intensity** between AD ($k = 2$) and Control ($k = 1$) clusters are more explicit.

DeSarbo WS, Cron WL (1988) A maximum likelihood methodology for clusterwise linear regression. Journal of classification 5(2):249–282

Figueiredo MAT, Jain AK (2002) Unsupervised learning of finite mixture models. IEEE Transactions on pattern analysis and machine intelligence 24(3):381–396

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

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

Gao C, Zhu Y, Shen X, Pan W (2016) Estimation of multiple networks in gaussian mixture models. Electronic journal of statistics 10:1133

Guo J, Levina E, Michailidis G, Zhu J (2011) Joint estimation of multiple graphical models. Biometrika 98(1):1–15

Hao B, Sun WW, Liu Y, Cheng G (2017) Simultaneous clustering and estimation of heterogeneous graphical models. The Journal of Machine Learning Research 18(1):7981–8038

Hara Y, Suzuki J, Kuwahara M (2018) Network-wide traffic state estimation using a mixture gaussian graphical model and graphical lasso. Transportation Research Part C: Emerging Technologies 86:622–638

Honorio J, Samaras D (2010) Multi-task learning of gaussian graphical models. In: ICML, Citeseer, pp 447–454

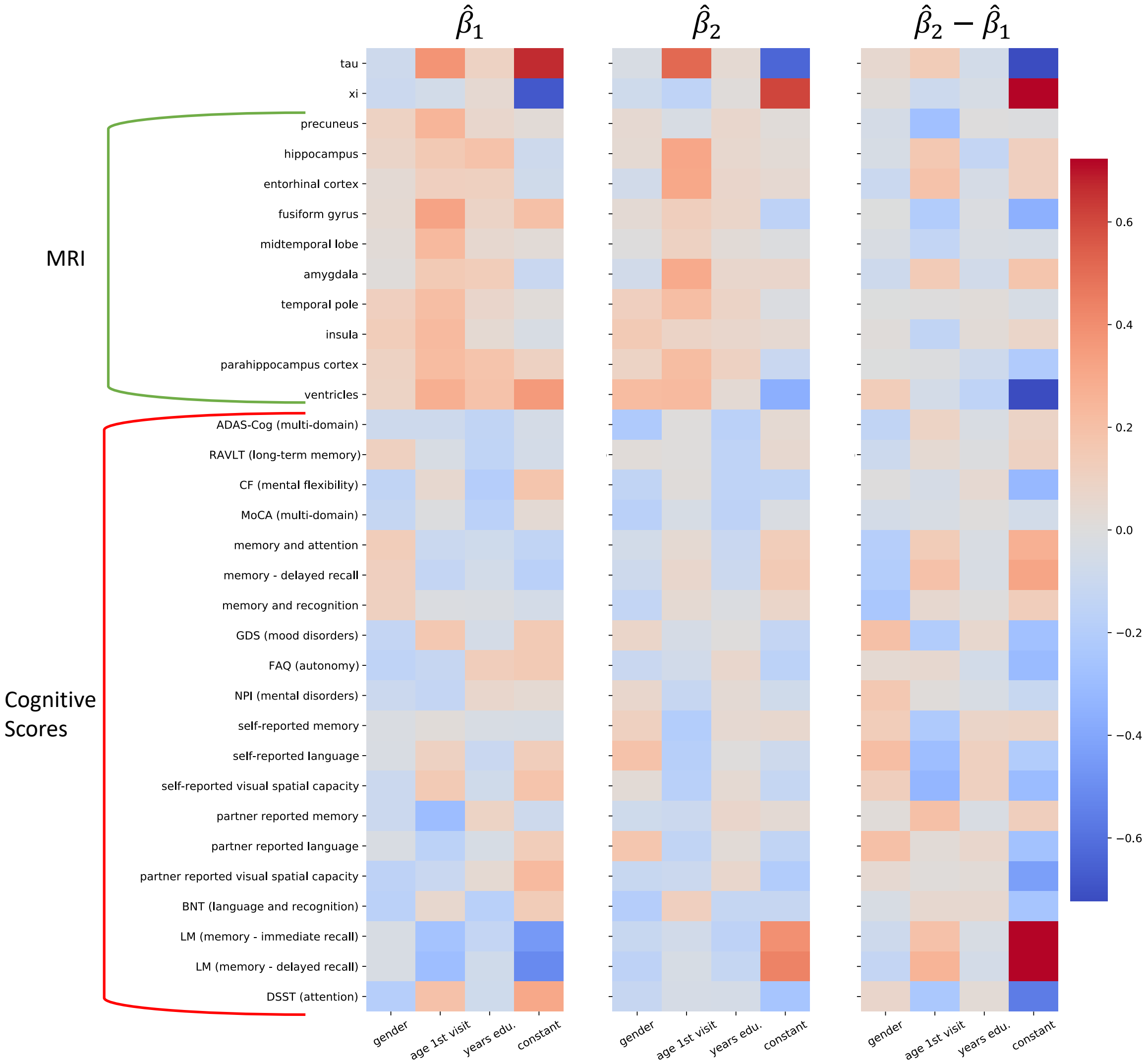

Fig. 8 Oracle MLE $\widehat{\beta}_k := -\widehat{\Sigma}_k \widehat{\Theta}_k^T$ estimated with the knowledge of the real labels (supervised estimation). A comparison with Fig. 7 shows that the C-EM properly recoverd these structures. (Left) $\widehat{\beta}_1$, MLE of the **Control patients** sub-population. (Middle) $\widehat{\beta}_2$, MLE of the **AD patients**. (Right) $\widehat{\beta}_2 - \widehat{\beta}_1$, their **difference in intensity**.

Huang F, Chen S, Huang SJ (2018) Joint estimation of multiple conditional gaussian graphical models. IEEE transactions on neural networks and learning systems 29(7):3034–3046

Hurvich CM, Tsai CL (1989) Regression and time series model selection in small samples. Biometrika 76(2):297–307

Jordan MI, Jacobs RA (1994) Hierarchical mixtures of experts and the em algorithm. Neural computation 6(2):181–214

Khalili A, Chen J (2007) Variable selection in finite mixture of regression models. Journal of the american Statistical association 102(479):1025–1038

Kim M (2016) Sparse inverse covariance learning of conditional gaussian mixtures for multiple-output regression. Applied Intelligence 44(1):17–29

Krishnamurthy A (2011) High-dimensional clustering with sparse gaussian mixture models. Unpublished paper pp 191–192

Kullback S, Leibler RA (1951) On information and sufficiency. The annals of mathematical statistics 22(1):79–86

Lartigue T (2020) Mixture of gaussian graphical models with constraints. PhD thesis, Institut Polytechnique de Paris

Lee H, Ghosh SK (2009) Performance of information criteria for spatial models. Journal of statistical computation and simulation 79(1):93–106

Mohan K, London P, Fazel M, Witten D, Lee SI (2014) Node-based learning of multiple gaussian graphical models. The Journal of Machine Learning Research 15(1):445–488

Ou-Yang L, Zhang XF, Hu X, Yan H (2019) Differential network analysis via weighted fused conditional gaussian graphical model. IEEE/ACM transactions on computa-

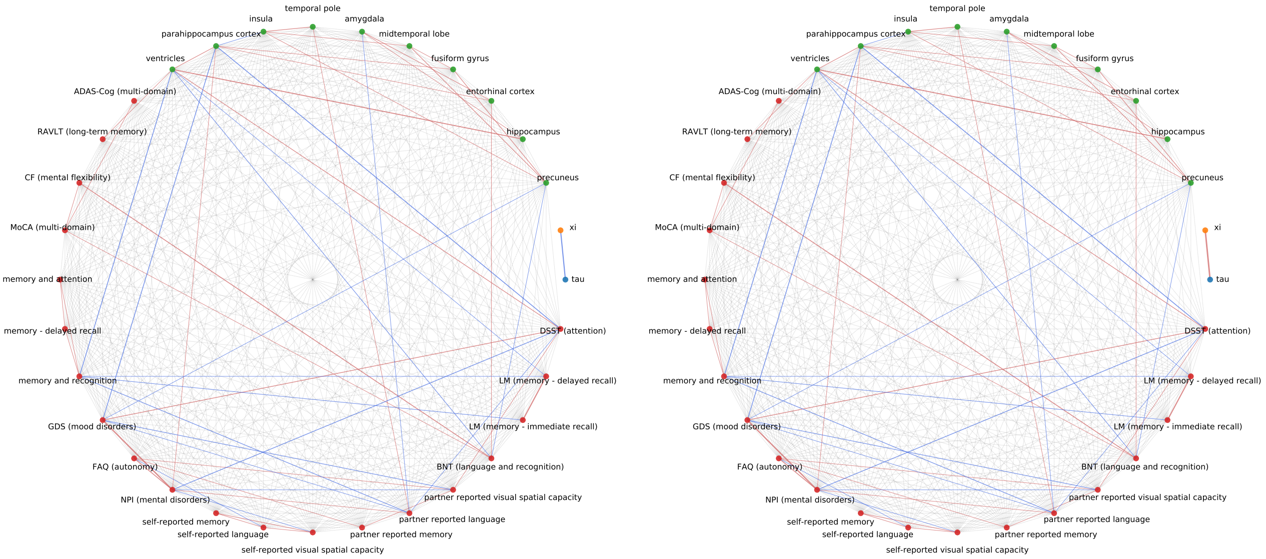

Fig. 9 (Left) Conditional correlation graph of the estimated cluster most correlated with the **"Control" diagnosis**. (Right) Conditional correlation graph of the estimated cluster most correlated with the **"AD" diagnosis**.

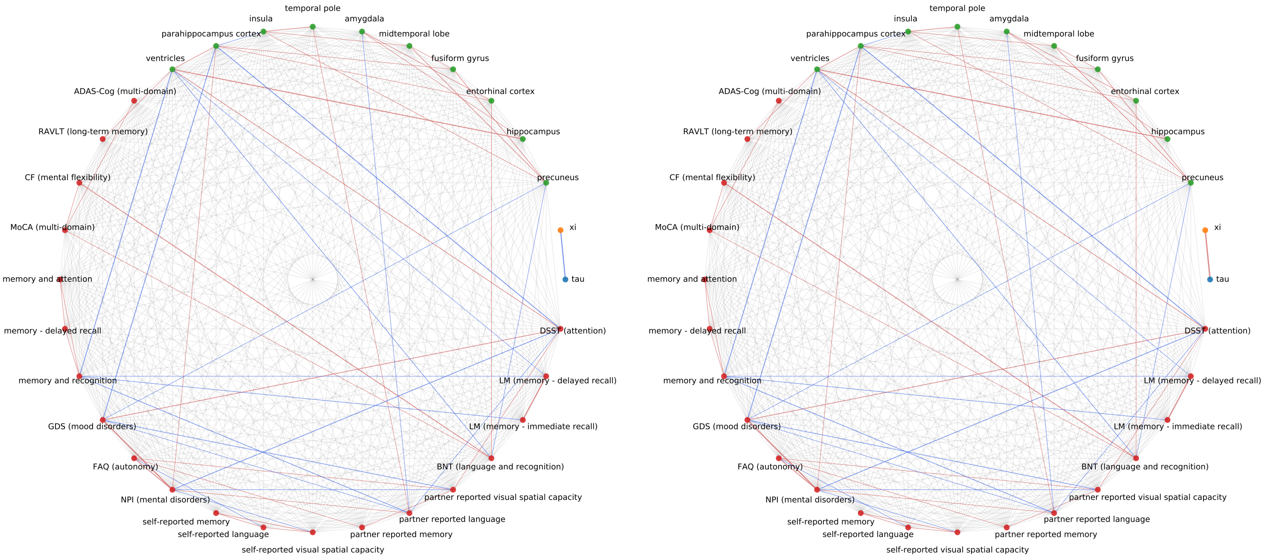

Fig. 10 (Left) Conditional correlation graph estimated in a supervised way on the **"Control" patients**. (Right) Conditional correlation graph estimated in a supervised way on the **"AD" patients**.

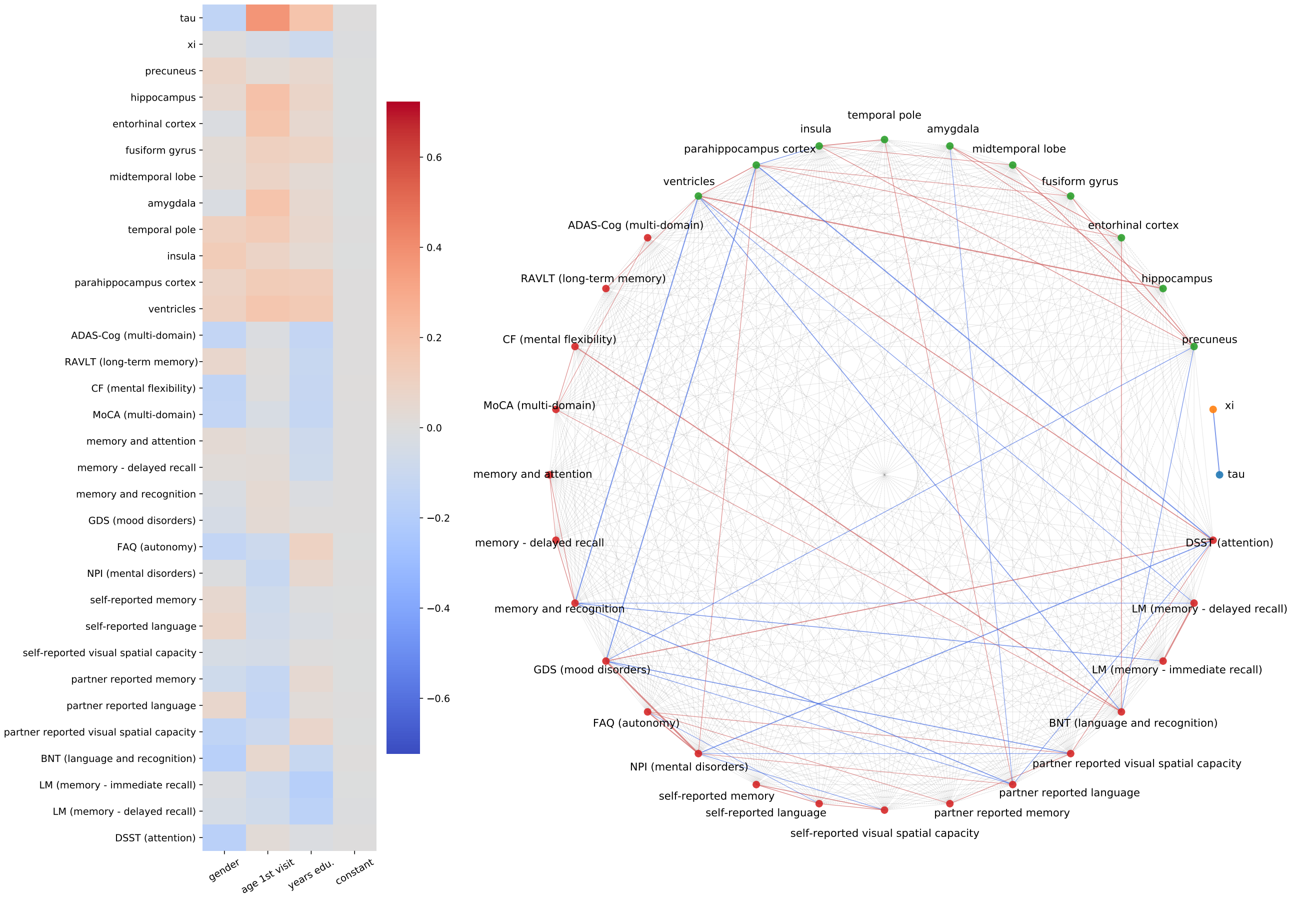

Fig. 11 Parameters estimated with a simple CGGM on all data. (Left) $\hat{\beta} = -\widehat{\Sigma}\widehat{\Theta}$. (Right) Conditional correlations graph.

tional biology and bioinformatics

Schiratti JB, Allassonniere S, Routier A, Colliot O, Durrleman S, Initiative ADN, et al. (2015) A mixed-effects model with time reparametrization for longitudinal univariate manifold-valued data. In: International Conference on Information Processing in Medical Imaging, Springer, pp 564–575

Schwarz G, et al. (1978) Estimating the dimension of a model. The annals of statistics 6(2):461–464

Sohn KA, Kim S (2012) Joint estimation of structured sparsity and output structure in multiple-output regression via inverse-covariance regularization. In: Artificial Intelligence and Statistics, pp 1081–1089

Varoquaux G, Gramfort A, Poline JB, Thirion B (2010) Brain covariance selection: better individual functional connectivity models using population prior. In: Advances in neural information processing systems, pp 2334–2342

Wytock M, Kolter Z (2013) Sparse gaussian conditional random fields: Algorithms, theory, and application to energy forecasting. In: International conference on machine learning, pp 1265–1273

Yang MS, Lai CY, Lin CY (2012) A robust em clustering algorithm for gaussian mixture models. Pattern Recognition 45(11):3950–3961

Yang S, Lu Z, Shen X, Wonka P, Ye J (2015) Fused multiple graphical lasso. SIAM Journal on Optimization 25(2):916–943

Yin J, Li H (2011) A sparse conditional gaussian graphical model for analysis of genetical genomics data. The annals of applied statistics 5(4):2630

Yuan M, Lin Y (2006) Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68(1):49–67

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

Zhou H, Pan W, Shen X (2009) Penalized model-based clustering with unconstrained covariance matrices. Electronic journal of statistics 3:1473