



**HAL**  
open science

# A Robust and Flexible EM Algorithm for Mixtures of Elliptical Distributions with Missing Data

Florian Mouret, Alexandre Hippert-Ferrer, Frédéric Pascal, Jean-Yves  
Tourneret

► **To cite this version:**

Florian Mouret, Alexandre Hippert-Ferrer, Frédéric Pascal, Jean-Yves Tourneret. A Robust and Flexible EM Algorithm for Mixtures of Elliptical Distributions with Missing Data. *IEEE Transactions on Signal Processing*, 2023, 71, pp.1–25. 10.1109/TSP.2023.3267994 . hal-04317202

**HAL Id: hal-04317202**

**<https://ut3-toulouseinp.hal.science/hal-04317202v1>**

Submitted on 10 Apr 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

# A Robust and Flexible EM Algorithm for Mixtures of Elliptical Distributions with Missing Data

Florian Mouret<sup>1,2</sup>, Alexandre Hippert-Ferrer<sup>3</sup>, Frédéric Pascal<sup>3</sup>, and Jean-Yves Tournet<sup>2</sup>

<sup>1</sup>TerraNIS, Ramonville-Saint-Agne, 31520, France

<sup>2</sup>University of Toulouse, IRIT-INP-ENSEEIH / TésA, Toulouse, 31000, France

<sup>3</sup>Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des signaux et systèmes, Gif-sur-Yvette, 91190, France

This is a preprint version of the research paper published in *IEEE Transaction on Signal Processing* : F. Mouret, A. Hippert-Ferrer, F. Pascal and J. -Y. Tournet, "A Robust and Flexible EM Algorithm for Mixtures of Elliptical Distributions with Missing Data," in IEEE Transactions on Signal Processing, vol. 71, pp. 1669-1682, 2023, doi: [10.1109/TSP.2023.3267994](https://doi.org/10.1109/TSP.2023.3267994).

## Abstract

This paper tackles the problem of missing data imputation for noisy and non-Gaussian data. A classical imputation method, the Expectation Maximization (EM) algorithm for Gaussian mixture models, has shown interesting properties when compared to other popular approaches such as those based on  $k$ -nearest neighbors or on multiple imputations by chained equations. However, Gaussian mixture models are known to be non-robust to heterogeneous data, which can lead to poor estimation performance when the data is contaminated by outliers or follows non-Gaussian distributions. To overcome this issue, a new EM algorithm is investigated for mixtures of elliptical distributions with the property of handling potential missing data. This paper shows that this problem reduces to the estimation of a mixture of Angular Gaussian distributions under generic assumptions (*i.e.*, each sample is drawn from a mixture of elliptical distributions, which is possibly different for one sample to another). In that case, the complete-data likelihood associated with mixtures of elliptical distributions is well adapted to the EM framework with missing data thanks to its conditional distribution, which is shown to be a multivariate  $t$ -distribution. Experimental results on synthetic data demonstrate that the proposed algorithm is robust to outliers and can be used with non-Gaussian data. Furthermore, experiments conducted on real-world datasets show that this algorithm is very competitive when compared to other classical imputation methods.

## Keywords

EM algorithm, Elliptical distributions, Angular Gaussian distributions, Mixture Models, Missing data, Imputation

## 1 Introduction

Missing data is a recurrent problem in data analysis that has been studied for decades (Anderson, 1957; Dempster et al., 1977; Little and Rubin, 2002; van Buuren, 2018). Missing data appear in a wide range of applications including biomedical signal processing, medical imaging (Cismondi et al., 2013; Mirza et al., 2019) and remote sensing (Shen et al., 2015). As an example, in remote sensing applications, missing data can be due to acquisition problems, cloud coverage or poor atmospheric conditions. The missing data problem is also of critical importance in applications relying on techniques that are non-robust to the absence of data, which is often the case with classical machine learning approaches (*e.g.*, most of the regression or classification algorithms provided in the benchmark Python library *scikit-learn* (Pedregosa et al., 2011) cannot be used with missing data). Moreover, having access to imputed values can be interesting for the end-user. An example that will be investigated in this paper is crop monitoring based on remote sensing images, which requires to have access to timely and accurate information on the crop status (Moran et al., 1997; Mouret et al., 2022).

## 1.1 Related work

Missing value imputation (MVI) is a common solution to bypass the data incompleteness. Two main approaches are generally used for this task, namely *statistical* and *machine learning* techniques (Lin and Tsai, 2020). Some imputation strategies are very simple, *e.g.*, imputing the missing data by the mean or mode of the feature or by linear interpolation when working with time series. The simplicity of these methods and their straightforward implementation have motivated their use in various applications (Farhangfar et al., 2007). However, their performance can be limited in some practical applications, motivating the use of more sophisticated techniques for MVI. Methods based on the Expectation Maximization (EM) algorithm have been widely used for MVI (Lin and Tsai, 2020). As explained in (Ghahramani and Jordan, 1994a), the EM algorithm can be naturally extended to handle missing data, the problem of mixture estimation being itself a missing data problem. Similarly to clustering or classification tasks, a particular attention has been devoted to the EM algorithm for Gaussian Mixture Models (GMM) with missing data (Dempster et al., 1977; Ghahramani and Jordan, 1994a; Eiroa et al., 2014). However, GMM estimation is known to be non-robust to noisy data and outliers (Campbell, 1984; Tadjudin and Landgrebe, 2000; Roizman et al., 2020). Moreover, when the data has a non-Gaussian distribution (*i.e.*, with heavier or lighter tails than the Gaussian distribution), the performance of GMM estimation algorithms might decrease significantly (Fraley and Raftery, 2002). In the complete-data case, various strategies have been investigated to solve these issues ranging from robust parameter estimation to the use of non-Gaussian distributions such as multivariate  $t$ - or hyperbolic distributions (Campbell, 1984; Tadjudin and Landgrebe, 2000; Peel and McLachlan, 2000; Browne and McNicholas, 2015). Some of these strategies have been adapted to the missing data case, *e.g.*, using multivariate  $t$ -distributions Wang et al. (2004) or skew  $t$ -distributions Wei et al. (2019). However, their extension to the missing data case is generally not straightforward since new conditional expectations have to be computed during the expectation step of the EM algorithm.

## 1.2 Contributions of this work

A flexible EM algorithm (FEM) was recently investigated in Roizman et al. (2020, 2021), showing good properties for the clustering of noisy and non-Gaussian data. An outstanding property of this algorithm is its robustness to the underlying data distribution when assuming cluster-independent density generators, see (Roizman et al., 2020, Proposition 4). This property was used to build a versatile EM clustering algorithm characterized by a simple parameter tuning (*i.e.*, self-contained).

This paper proposes to extend the FEM algorithm to handle missing data. The resulting algorithm is able to perform an efficient MVI, which generally outperforms the classical EM for GMMs<sup>1</sup>. The main contributions of this work can be summarized as follows:

- A new EM algorithm for mixtures of elliptical distributions potentially affected by missing data is derived. The algorithm assumes that each sample is drawn from a mixture of elliptical distributions, which is possibly different for one sample to another. Under these generic assumptions, the complete likelihood is shown to be upper bounded by the likelihood of a mixture of Angular Gaussian (AG) distributions. Moreover, the conditional distribution of the complete likelihood used in the considered EM framework is shown to be a Student’s  $t$ -distribution.
- Algorithms for the implementation of the proposed EM algorithm are provided. In addition, we show that the proposed algorithm is *intuitive* in the sense that its derivation is very similar to the EM for GMM in the missing data case.
- Imputation results obtained on different synthetic and real world datasets are presented to evaluate the flexibility of the proposed approach.

## 1.3 Notations

In the following,  $a$  refers to a scalar quantity,  $\mathbf{a}$  to a vector and  $\mathbf{A}$  to a matrix. The notation  $\det(\mathbf{A})$  (resp.  $\text{tr}(\mathbf{A})$ ) refers to the determinant of matrix  $\mathbf{A}$  (resp. trace of matrix  $\mathbf{A}$ ). Moreover  $\mathbf{A}^T$  is the transpose of  $\mathbf{A}$ .

The rest of this paper is organized as follows. Section 2 presents the general context and background necessary to understand the FEM method. Section 3 derives the proposed algorithm

---

<sup>1</sup>The proposed approach could also be used for clustering, as in the case without missing data. However, this task is not considered in this paper.

for missing data with appropriate theoretical justifications. [Section 4](#) evaluates the performance of the proposed FEM algorithm to impute missing values in various datasets, using both synthetic and real world data. Note that this paper focuses on the data reconstruction task, the performance of the FEM algorithm for clustering was evaluated in [Roizman et al. \(2020\)](#) in the complete data case. [Section 5](#) finally draws some conclusions and presents some future work that would deserve to be conducted.

## 2 Background: an EM algorithm for mixtures of elliptical distributions

Elliptical Symmetric (ES) distributions refer to a broad family of distributions generalizing the multivariate normal distribution, such as the famous multivariate  $t$ -distribution ([Peel and McLachlan, 2000](#)) or the multivariate generalized Gaussian distributions ([Pascal et al., 2013](#)). They have been used in a wide range of applications to deal with heavy-tailed distribution or outliers ([Conte et al., 2002](#); [Tadjudin and Landgrebe, 2000](#)). Their flexibility and robustness have proven to be very interesting in tasks such as classification or clustering when compared to standard methods ([Hippert-Ferrer et al., 2022](#); [Roizman et al., 2020](#)), particularly for heterogeneous or noisy datasets. More information regarding elliptical distributions can be found in the pioneering paper from [Kelker \(1970\)](#), which introduced for the first time this generalization of the Gaussian distribution. A more recent survey in the complex case can be found in [Olhila et al. \(2012\)](#).

This section briefly recalls the FEM algorithm in its standard formulation, *i.e.*, without missing data, as proposed in [Roizman et al. \(2020\)](#) (for more details and justifications, the reader is invited to consult this reference). We focus on the case where the density generator is the same for each component, which allows us to derive generic estimators that do not depend on the underlying distribution of the data. In a second step, we extend this procedure to the missing data scenario, which is the main contribution of this work. The resulting algorithm is very intuitive and has the interesting property to be robust to outliers.

### 2.1 Data model and complete log-likelihood

Suppose that each sample  $\mathbf{x}_i \in \mathbb{R}^m$  of the dataset  $\mathcal{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_N\}$  (containing  $N$  samples of dimension  $m$ ) is drawn from a mixture of distributions with the following probability density function (pdf):

$$f_{i,\boldsymbol{\theta}}(\mathbf{x}_i) = \sum_{k=1}^K \pi_k f_{i,\boldsymbol{\theta}_k}(\mathbf{x}_i) \text{ with } \sum_{k=1}^K \pi_k = 1, \quad (1)$$

where  $\pi_k$  denotes the *a priori* probability of class  $k$ ,  $\boldsymbol{\theta} = (\boldsymbol{\theta}_1^T, \dots, \boldsymbol{\theta}_K^T)^T$  and  $f_{i,\boldsymbol{\theta}_k}$  is the pdf of  $\mathbf{x}_i$  (that is potentially different for each sample). Note that the parameters of cluster  $k$  are grouped into the vector  $\boldsymbol{\theta}_k$ . This paper assumes that  $f_{i,\boldsymbol{\theta}_k}$  is an ES distribution with mean vector  $\boldsymbol{\mu}_k$  and covariance matrix  $\tau_{ik} \boldsymbol{\Sigma}_k$  ([Kelker, 1970](#)), whose pdf can be written:

$$f_{i,\boldsymbol{\theta}_k}(\mathbf{x}_i) = A_{ik} \det(\boldsymbol{\Sigma}_k)^{-1/2} \tau_{ik}^{-m/2} g_{i,k} \left( \frac{(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)}{\tau_{ik}} \right), \quad (2)$$

where  $A_{ik}$  is a normalization constant and  $g_{i,k}$  is a density generator such that [Eq. 2](#) defines a pdf.  $\boldsymbol{\Sigma}_k$  is referred to as the scatter matrix, which defines the structure of the covariance of  $\mathbf{x}_i$  (in particular  $\boldsymbol{\Sigma}_k$  is equal to the covariance matrix of  $\mathbf{x}_i$  up to a scale factor). Finally,  $\tau_{ik}$  is known as the scale or nuisance parameter and is not of direct interest when estimating the other model parameters. After introducing the scale factor

$$s_{ik} = \frac{(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)}{\tau_{ik}},$$

the pdf  $f_{i,\boldsymbol{\theta}}$  can be expressed as follows:

$$f_{i,\boldsymbol{\theta}_k}(\mathbf{x}_i) = A_{ik} \det(\boldsymbol{\Sigma}_k)^{-1/2} [(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)]^{-m/2} s_{ik}^{m/2} g_{i,k}(s_{ik}). \quad (3)$$

In the rest of this paper, **we will suppose that the density generator is the same for each component** (but is possibly different from one sample to another), *i.e.*,  $g_{i,k} = g_i$ , leading to

$$f_{i,\boldsymbol{\theta}_k}(\mathbf{x}_i) = A_i \det(\boldsymbol{\Sigma}_k)^{-1/2} [(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)]^{-m/2} s_{ik}^{m/2} g_i(s_{ik}). \quad (4)$$

In this paper, we show that this specific structure for the density generator allows us to have a model that is as generic as possible with intuitive and relatively simple derivations. Indeed, assuming the pdf depends on sample  $i$  is more generic without any complexity added to the algorithm. Moreover, while preserving the generality of the estimated model, this assumption on the density generator provides an algorithm allowing the model parameters to be estimated without knowing the generator  $g_i$ , which is an interesting property of the FEM algorithm [Roizman et al. \(2020\)](#). Note that a precise knowledge of the underlying data distribution could improve the estimation of the model parameters ([Roizman et al., 2020](#)). However, the main interest of the FEM algorithm presented in this paper is that it can be used without any a priori on the data distribution (*e.g.*, if the  $\mathbf{x}$  observations are identically distributed or not).

The complete log-likelihood for a mixture of elliptical distributions can be defined by introducing latent vectors  $\mathcal{Z} = \{z_1, \dots, z_N\}$  containing the cluster labels for the different observed vectors. More precisely, for each sample  $\mathbf{x}_i$ , the latent vector  $\mathbf{z}_i = (z_{i1}, \dots, z_{iK})^T$  is such that  $z_{ik} = 1$  if the vector  $\mathbf{x}_i$  belongs to the  $k$ th component of the model and  $z_{ik} = 0$  otherwise:

$$\log \mathcal{L}_c(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) = \sum_{i=1}^N \sum_{k=1}^K z_{ik} \log(\pi_k f_{i, \boldsymbol{\theta}_k}(\mathbf{x}_i)). \quad (5)$$

Based on [Eq. 4](#),  $\log \mathcal{L}_c(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z})$  can be rewritten as:

$$\log \mathcal{L}_c(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) = \sum_{i=1}^N \sum_{k=1}^K z_{ik} [\ell_{0k}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + \ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik})] \quad (6)$$

with

$$\begin{aligned} \ell_{0k}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) &= \log(\pi_k) + \log(A_i) + \frac{1}{2} \log(\det(\boldsymbol{\Sigma}_k^{-1})) \\ &\quad - \frac{m}{2} \log((\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)), \end{aligned} \quad (7)$$

$$\ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}) = \log\left(s_{ik}^{m/2} g_i(s_{ik})\right). \quad (8)$$

## 2.2 The M-step

Using the EM algorithm, where  $t$  denotes the current iteration, we can derive the new set of parameters  $\boldsymbol{\theta}^{(t+1)}$  based on the current set of parameters  $\boldsymbol{\theta}^{(t)}$  and  $p_{ik} = E[z_{ik} | \mathbf{x}_i, \boldsymbol{\theta}^{(t)}]$ , the probability that sample  $i$  has been generated by component  $k$ . The probabilities  $p_{ik}$  (also referred to as responsibilities) are computed in the E-step defined in the next section. We begin by the M-step since interesting results regarding the estimation of  $\tau_{ik}$  will allow the E-step to be simplified significantly. For brevity, denote as  $\boldsymbol{\theta}^{(t)} = \boldsymbol{\theta}$ , *i.e.*,  $\boldsymbol{\mu}_k^{(t)} = \boldsymbol{\mu}_k$ ,  $\boldsymbol{\Sigma}_k^{(t)} = \boldsymbol{\Sigma}_k$ ,  $\tau_{ik}^{(t)} = \tau_{ik}$  and  $\pi_k^{(t)} = \pi_k$  the current set of parameters.

- **Estimation of the nuisance parameters  $\tau_{ik}$ :** One can observe that  $\tau_{ik}$  is only related to the term  $\ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik})$  of the complete log-likelihood. For fixed  $(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ , the value of  $\tau_{ik}$  maximizing  $p_{ik} \log(s_{ik}^{m/2} g_i(s_{ik}))$  is:

$$\tau_{ik} = \frac{(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)}{a_{im}}, \quad (9)$$

with  $a_{im} = \arg \sup_t (t^{m/2} g_i(t))$  (see proof and complementary results in [Roizman et al. \(2020\)](#)), in particular, it has been shown that  $a_{im} \sim m$  when  $m$  is sufficiently large). After replacing  $\tau_{ik}$  by its estimate in  $s_{ik}$ , the following results can be obtained:

$$s_{ik} = \frac{(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)}{\tau_{ik}} = a_{im}, \quad (10)$$

$$\ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}) = \log(s_{ik}^{m/2} g_i(s_{ik})) = \log(a_{im}^{m/2} g_i(a_{im})). \quad (11)$$

This central result indicates that  $\ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik})$  does not depend on  $(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$  within the EM framework, and thus only depends on  $a_{im}$ . Consequently, estimating these parameters using the complete log-likelihood can be done using  $\ell_{0k}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$  only, which

reduces to estimate the parameters of a mixture of AG distributions. More formally, we denote as  $AG_m(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$  the AG distribution studied in (Ollila et al., 2012), with pdf

$$f_{AG}(\mathbf{x}_i) = B_i \det(\boldsymbol{\Sigma}_k)^{-1/2} [(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)]^{-m/2}, \quad (12)$$

where  $B_i$  is a normalization constant.

**Remark 1.** The estimation of  $\tau_{ik}$  reduces  $\boldsymbol{\Sigma}_k^{-1/2}(\mathbf{x}_i - \boldsymbol{\mu}_k)$  to lie on a hypersphere, which explains the apparition of the AG distribution. Indeed, the stochastic representation theorem (see e.g., Roizman et al. (2020)) for  $\mathbf{x}_i$  distributed according to an elliptical distribution can be stated as follows:

$$\mathbf{x}_i \stackrel{d}{=} \boldsymbol{\mu}_k + \sqrt{\mathcal{Q}_{ik}} \mathbf{A}_k \mathbf{u}_i$$

where  $\stackrel{d}{=}$  stands for “is distributed as”,  $\mathcal{Q}_{ik}$  is a positive random variable independent of  $\mathbf{u}_i$ ,  $\mathbf{A}_k$  is such that  $\mathbf{A}_k \mathbf{A}_k^T = \boldsymbol{\Sigma}_k$  and  $\mathbf{u}_i$  is a uniform random vector on the unit hypersphere. Since  $\mathcal{Q}_{ik}$  is distributed as  $(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)$  (Ollila et al., 2012), the following result is obtained:

$$\frac{\boldsymbol{\Sigma}_k^{-1/2}(\mathbf{x}_i - \boldsymbol{\mu}_k)}{\sqrt{\mathcal{Q}_{ik}}} \stackrel{d}{=} \mathbf{u}_i,$$

which shows that the normalized observation  $\mathbf{x}_i$  lies on the unit hypersphere.

- **Estimating the model parameters**  $(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ : maximizing the log-likelihood with respect to the model parameters leads to the following expressions defined through fixed-point equations:

$$\pi_k = \frac{1}{N} \sum_{i=1}^N p_{ik}, \quad (13)$$

$$\boldsymbol{\mu}_k = \sum_{i=1}^N \frac{w_{ik} p_{ik} \mathbf{x}_i}{\sum_{l=1}^N w_{lk} p_{lk}}, \quad (14)$$

$$\boldsymbol{\Sigma}_k = m \sum_{i=1}^N \frac{p_{ik} w_{ik} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T}{\sum_{l=1}^N p_{lk}}, \quad (15)$$

where  $w_{ik} = \frac{1}{(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)}$ . One can notice that Eq. 14 and Eq. 15 are classical expressions of the mean and covariance matrix in a robust estimation framework. More precisely,  $w_{ik}$  are weights reducing the influence of outlier samples (see for instance (Campbell, 1984; Tadjudin and Landgrebe, 2000)), where similar forms are obtained using the Huber function for the weight function). Typically, when  $w_{ik}$  is close to zero, the considered sample will have little influence on the estimation of the model parameters.

Finally, it should be noted that the scatter matrix  $\boldsymbol{\Sigma}_k$  is equal to the covariance matrix up to a scale factor. In this work, we choose to fix the trace of  $\boldsymbol{\Sigma}_k$  to  $m$ , as in Roizman et al. (2020). The aforementioned reference has shown that the scale of  $\boldsymbol{\Sigma}_k$  has no influence on the clustering results. In subsection 3.2, we will show that the scale of  $\boldsymbol{\Sigma}_k$  has also no influence either on the imputation results in the missing data case, which is an important result.

### 2.3 The E-step

In the E-step, one needs to compute  $E[\log \mathcal{L}_c(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) | \boldsymbol{\theta}^{(t)}]$ , which reduces to evaluate  $\hat{p}_{ik} = E[z_{ik} | \mathbf{x}_i, \boldsymbol{\theta}^{(t)}]$  when there is no missing data. By replacing unknown parameters by the current state of parameters at iteration  $t$ , i.e., by  $\boldsymbol{\theta}^{(t)}$ ,  $\hat{p}_{ik}$  can be computed as follows:

$$\hat{p}_{ik} = \frac{\pi_k f_{i, \boldsymbol{\theta}_k}(\mathbf{x}_i)}{\sum_{j=1}^K \pi_j f_{i, \boldsymbol{\theta}_j}(\mathbf{x}_i)}. \quad (16)$$

Using the result obtained in Eq. 10 leads to:

$$\hat{p}_{ik} = \frac{\pi_k \det(\boldsymbol{\Sigma}_k)^{-1/2} [(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)]^{-m/2}}{\sum_{j=1}^K \pi_j \det(\boldsymbol{\Sigma}_j)^{-1/2} [(\mathbf{x}_i - \boldsymbol{\mu}_j)^T \boldsymbol{\Sigma}_j^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_j)]^{-m/2}}. \quad (17)$$

It is important to note here that the result obtained in Eq. 17 shows that  $\hat{p}_{ik}$  does not depend on the density generator  $g$  when  $g_{ik} = g_i$ . This result is particularly important since knowing the precise data distribution (and the corresponding density generator) is often not possible in practical applications.

## 2.4 Algorithm

Algorithm 1 provides the pseudo-code for the FEM implementation in the complete data case. The reader is invited to consult Roizman et al. (2020) for a discussion regarding implementation details and numerical considerations. Regarding the initialization of  $\theta^{(0)}$ , it is possible to use random values. However, a more efficient way is to use a fast clustering algorithm (such as the K-means algorithm) to have a more relevant initial guess for the set of parameters. This is for instance the strategy adopted in the Python library *scikit-learn* for the estimation of GMM.

---

**Algorithm 1** Scheme of the FEM algorithm in the complete data case (Roizman et al., 2020).

---

**Input:** Data  $\{\mathbf{x}_i\}_{i=1}^n$ ,  $K$  the number of clusters

**Output:** Clustering labels  $\mathcal{Z} = \{z_i\}_{i=1}^n$  and model parameters  $\theta = \{\pi_1, \dots, \pi_k, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K\}$

- 1: Initialize  $\theta^{(0)}$  (randomly or using a clustering algorithm);
  - 2:  $t \leftarrow 1$ ;
  - 3: **while** not convergence **do**
  - 4:   **for**  $1 \leq k \leq K$ : **do** ▷ E-step
  - 5:     
$$p_{ik}^{(t)} = \frac{\pi_k^{(t-1)} \det(\boldsymbol{\Sigma}_k^{(t-1)})^{-1/2} [(\mathbf{x}_i - \boldsymbol{\mu}_k^{(t-1)})^T (\boldsymbol{\Sigma}_k^{(t-1)})^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k^{(t-1)})]^{-m/2}}{\sum_{j=1}^K \pi_j^{(t-1)} \det(\boldsymbol{\Sigma}_j^{(t-1)})^{-1/2} [(\mathbf{x}_i - \boldsymbol{\mu}_j^{(t-1)})^T (\boldsymbol{\Sigma}_j^{(t-1)})^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_j^{(t-1)})]^{-m/2}};$$
  - 6:   **end for**
  - 7:   **for**  $1 \leq k \leq K$ : **do** ▷ M-step
  - 8:     
$$\pi_k^{(t)} = \frac{1}{N} \sum_{i=1}^N p_{ik}^{(t)};$$
  - 9:     set  $\boldsymbol{\mu}'_k = \boldsymbol{\mu}_k^{(t-1)}$  and  $\boldsymbol{\Sigma}'_k = \boldsymbol{\Sigma}_k^{(t-1)}$ ;
  - 10:    **while** not convergence **do** ▷ Fixed-point loop
  - 11:     
$$w_{ik} = \frac{1}{(\mathbf{x}_i - \boldsymbol{\mu}'_k)^T (\boldsymbol{\Sigma}'_k)^{-1} (\mathbf{x}_i - \boldsymbol{\mu}'_k)};$$
  - 12:     
$$\boldsymbol{\mu}''_k = \sum_{i=1}^N \frac{w_{ik} p_{ik}^{(t)} \mathbf{x}_i}{\sum_{l=1}^N w_{lk} p_{lk}^{(t)}};$$
  - 13:     
$$\boldsymbol{\Sigma}''_k = m \sum_{i=1}^N \frac{w_{ik} p_{ik}^{(t)} (\mathbf{x}_i - \boldsymbol{\mu}'_k)(\mathbf{x}_i - \boldsymbol{\mu}'_k)^T}{\sum_{l=1}^N p_{lk}^{(t)}};$$
  - 14:     Update:  $\boldsymbol{\mu}'_k = \boldsymbol{\mu}''_k$  and  $\boldsymbol{\Sigma}'_k = \boldsymbol{\Sigma}''_k$ ;
  - 15:    **end while**
  - 16:    Update:  $\boldsymbol{\mu}_k^{(t)} = \boldsymbol{\mu}''_k$  and  $\boldsymbol{\Sigma}_k^{(t)} = \boldsymbol{\Sigma}''_k$ ;
  - 17:   **end for**
  - 18:    $t \leftarrow t + 1$ ;
  - 19: **end while**
  - 20: Set  $z_i$  as the index  $k$  that has the maximum  $p_{ik}$ ;
-

### 3 A generalization to incomplete data

This section explains how to extend the previous FEM algorithm to handle missing data. In this case, each sample can be decomposed into  $\mathbf{x}_i = (\mathbf{x}_i^{o_i}, \mathbf{x}_i^{m_i})$ , where  $\mathbf{x}_i^{o_i}$  and  $\mathbf{x}_i^{m_i}$  are the vectors of observed and missing variables respectively (we denote  $\mathcal{X}^o$  the sets of all observed variables and  $\mathcal{X}^m$  the set of all missing variables). More generally, the superscripts  $o_i$  and  $m_i$  denote the observed and missing components of sample  $i$ . These subscripts can be used for matrices too, *e.g.*,  $\Sigma_k^{o_i m_i}$  refers to the elements of the matrix  $\Sigma_k$  in the rows and columns specified by  $o_i$  and  $m_i$ . Thus, the covariance matrix for  $\mathbf{x}_i$  is defined as

$$\Sigma_k = \begin{pmatrix} \Sigma_k^{o_i o_i} & \Sigma_k^{o_i m_i} \\ \Sigma_k^{m_i o_i} & \Sigma_k^{m_i m_i} \end{pmatrix} \quad (18)$$

where  $\Sigma_k$  has the same structure for any vector  $\mathbf{x}_i$ , with submatrices  $\Sigma_k^{o_i o_i}$ ,  $\Sigma_k^{o_i m_i}$ ,  $\Sigma_k^{m_i o_i}$  and  $\Sigma_k^{m_i m_i}$  changing according to the number of missing data (that depends on  $i$ ). For brevity, we will denote  $o_i = o$  and  $m_i = m$  in the following, but the reader should keep in mind that these subscripts are sample-dependent.

Note that the proposed algorithm assumes data missing completely at random (MCAR) or missing at random (MAR) (see [Little and Rubin \(2002\)](#) for a detailed description of the different missing data mechanisms), which is a standard assumption for EM-based algorithms. This implies that the missing data mechanism is ignorable, *i.e.*, the missingness is independent of all the values (MCAR) or independent of the missing values (MAR). In practice, the MCAR or MAR assumptions apply to a wide range of data. As a first example, the presence of clouds in remote sensing images induce missing data that may be regarded as MAR since the spatial distribution of clouds is independent of the land cover ([Salberg, 2011](#)). As a result, any cloud detection task will not change the distribution of the observed data. A second example concerns missing pixels due to sensor failure. This setting is clearly MCAR since the missing-data mechanism is independent of the observed and missing data. In both MAR and MCAR settings, valid inferences can be obtained by ignoring the missing-data mechanism ([Little and Rubin, 2002](#)) instead of using an ad hoc procedure (data deletion, mean imputation, *etc.*). Consequently, the use of the EM algorithm is justified by the assumption that the probability that a value is missing does not depend on the missing-data value itself. Finally, it would be interesting to study the case where the data is missing not at random (MNAR), such as in [Sportisse et al. \(2021\)](#), which is left for future work.

#### 3.1 The E-step

The E-step for elliptical distributions with missing data requires to evaluate  $E[\log \mathcal{L}_c(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) | \boldsymbol{\theta}^{(t)}, \mathcal{X}^o]$ . Similarly to the complete data case, this can be done by focusing only on the terms  $\ell_{0k}$  defined in [Eq. 7](#), which means that the parameters  $\tau_{ik}$  can be ignored. This leads us to the following proposition.

**Proposition 1.** *Maximizing the complete log likelihood with respect to  $\boldsymbol{\theta}_k$  and  $\mathcal{X}^o$  is equivalent to maximising:*

$$E \left[ \log \mathcal{L}_0(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) | \boldsymbol{\theta}^{(t)}, \mathcal{X}^o \right] = \sum_{i=1}^N \sum_{k=1}^K E \left[ z_{ik} \ell_{0k}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) | \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right]$$

*Proof.* See [Appendix A.1](#) for details. □

This result is of crucial importance, since it implies that the estimation problem reduces to the estimation of a mixture of AG distributions ( $\log \mathcal{L}_0(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z})$  is the log-likelihood of an AG distribution, whose pdf is provided in [Eq. 12](#)). This leads to:

$$E \left[ \log \mathcal{L}_0(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) | \boldsymbol{\theta}^{(t)}, \mathcal{X}^o \right] = \sum_{i=1}^N \sum_{k=1}^K \left( E \left[ z_{ik} | \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right] \times E \left[ \ell_{0k}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) | z_{ik} = 1, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right] \right), \quad (19)$$



where the expression of  $E[z_{ik}|\boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o]$  is similar to the case without missing data, except that  $\hat{p}_{ik}$  is estimated using the observed data:

$$\hat{p}_{ik} = \frac{\pi_k \det(\boldsymbol{\Sigma}_k^{oo})^{-1/2} [(\mathbf{x}_i^o - \boldsymbol{\mu}_k^o)^T (\boldsymbol{\Sigma}_k^{oo})^{-1} (\mathbf{x}_i^o - \boldsymbol{\mu}_k^o)]^{-d^o/2}}{\sum_{j=1}^K \pi_j \det(\boldsymbol{\Sigma}_j^{oo})^{-1/2} [(\mathbf{x}_i^o - \boldsymbol{\mu}_j^o)^T (\boldsymbol{\Sigma}_j^{oo})^{-1} (\mathbf{x}_i^o - \boldsymbol{\mu}_j^o)]^{-d^o/2}}, \quad (20)$$

with  $d^o$  the number of observed features for each sample.

### 3.1.1 Conditional expectations

As can be observed in Eq. 19, additional terms coming from  $E[\ell_{0k}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) | z_{ik} = 1, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o]$  have to be computed with respect to the case without missing data. More precisely, we need to compute two new sufficient statistics,  $E[\mathbf{x}_i^m | z_{ik} = 1, \mathbf{x}_i^o, \boldsymbol{\theta}]$  and  $E[\mathbf{x}_i^m (\mathbf{x}_i^m)^T | z_{ik} = 1, \mathbf{x}_i^o, \boldsymbol{\theta}]$  which are first and second order conditional expectations of the missing variables for a sample  $\mathbf{x}_i$ , given that  $\mathbf{x}_i$  has been generated by the AG  $k$  (this is in fact similar to the GMM case, but with different expectations, see Ghahramani and Jordan (1994b) for a detailed example in that case). These sufficient statistics can be determined easily after identifying the conditional distribution  $f_{i, \boldsymbol{\theta}_k}(\mathbf{x}_i^m | \mathbf{x}_i^o)$ . Using Proposition 1, we have noted that the pdf  $f_{i, \boldsymbol{\theta}_k}(x_i)$  reduces to an AG distribution, which is an outstanding result. Based on this, the conditional mean and covariance of  $\mathbf{x}_i^m | \mathbf{x}_i^o, \boldsymbol{\theta}$  can be determined using the following proposition.

**Proposition 2.** *Suppose that  $\mathbf{x} \sim AG_m(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ , with  $\mathbf{x} = [\mathbf{x}_1^T, \mathbf{x}_2^T]^T \in \mathbb{R}^d$ , and  $d = d_1 + d_2$ . Then,  $\mathbf{x}_2 | \mathbf{x}_1 \sim t_\nu(\boldsymbol{\mu}_{2,1}, \boldsymbol{\Sigma}_{22,1})$ , where  $t_\nu(\boldsymbol{\mu}_{2,1}, \boldsymbol{\Sigma}_{22,1})$  is a multivariate  $t$ -distribution with  $\nu = d_1$  degrees of freedom, mean vector  $\boldsymbol{\mu}_{2,1} = \boldsymbol{\mu}_2 + \boldsymbol{\Sigma}_{21} \boldsymbol{\Sigma}_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1)$  and scale matrix  $\boldsymbol{\Sigma}_{22,1} = s_{22,1} (\boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21} \boldsymbol{\Sigma}_{11}^{-1} \boldsymbol{\Sigma}_{12})$  with  $s_{22,1} = \frac{1}{d_1} (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \in \mathbb{R}$ .*

*Proof.* See Appendix A.2. □

**Remark 2.** *This proposition is in agreement with some results obtained in the general case of elliptical distributions, denoted as  $E_d(\boldsymbol{\mu}, \boldsymbol{\Lambda})$  (see for instance (Bilodeau and Brenner, 1999, Chapter 13) for detailed proofs). More precisely, if  $\mathbf{x} = [\mathbf{x}_1^T, \mathbf{x}_2^T]^T \sim E_d(\boldsymbol{\mu}, \boldsymbol{\Lambda})$ , then  $\mathbf{x}_2 | \mathbf{x}_1 \sim E_{d_2}(\boldsymbol{\mu}_{2,1}, \boldsymbol{\Lambda}_{22,1})$  with  $\boldsymbol{\mu}_{2,1} = \boldsymbol{\mu}_2 + \boldsymbol{\Lambda}_{21} \boldsymbol{\Lambda}_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1)$  and  $\boldsymbol{\Lambda}_{22,1} = \boldsymbol{\Lambda}_{22} - \boldsymbol{\Lambda}_{21} \boldsymbol{\Lambda}_{11}^{-1} \boldsymbol{\Lambda}_{12}$ . In that case, the covariance matrix of  $\mathbf{x}_2 | \mathbf{x}_1$  is  $\text{cov}[\mathbf{x}_2 | \mathbf{x}_1] = w(\mathbf{x}_1) \boldsymbol{\Lambda}_{22,1}$ , with  $w$  a function depending on  $\mathbf{x}_1$ . This paper shows that within the EM framework for elliptical distributions with missing data, the estimation of the model parameters  $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \pi_k)$  of any elliptical distribution reduces to consider an Angular Gaussian distribution, when assuming that the density generator is the same for each component. Further work based on this result could be interesting, but such investigations are out of the scope of this paper.*

The expectation of the log-likelihood can then be derived as follows:

$$\begin{aligned} E \left[ \log \mathcal{L}_0(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) | \boldsymbol{\theta}^{(t)}, \mathcal{X}^o \right] &= \sum_{i=1}^N \sum_{k=1}^K \hat{p}_{ik} \left( \log(\pi_k) + \log(A_i) - \frac{1}{2} \log(\det(\boldsymbol{\Sigma}_k)) \right. \\ &\quad \left. - \frac{m}{2} \log \left[ \text{tr}((\boldsymbol{\Sigma}_k)^{-1} \tilde{\boldsymbol{\Sigma}}_{ik}) \right] \right. \\ &\quad \left. - \frac{m}{2} \log \left[ (\tilde{\mathbf{x}}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\tilde{\mathbf{x}}_i - \boldsymbol{\mu}_k) \right] \right), \end{aligned} \quad (21)$$

where  $\tilde{\boldsymbol{\Sigma}}_{ik}$  and  $\tilde{\mathbf{x}}_i$  are defined hereafter. Eq. 21 is obtained by computing the following sufficient statistics thanks to the conditional distributions found in Proposition 2:

$$\mathbb{E}[\mathbf{x}_i^m | z_{ik} = 1, \mathbf{x}_i^o, \boldsymbol{\theta}] = \boldsymbol{\mu}_{ik}^m = \boldsymbol{\mu}_k^m + \boldsymbol{\Sigma}_k^{mo} (\boldsymbol{\Sigma}_k^{oo})^{-1} (\mathbf{x}_i^o - \boldsymbol{\mu}_k^o), \quad (22)$$

$$\mathbb{E}[\mathbf{x}_i^m (\mathbf{x}_i^m)^T | z_{ik} = 1, \mathbf{x}_i^o, \boldsymbol{\theta}] = \boldsymbol{\Sigma}_{ik}^{mm} = C_i^o \times (\boldsymbol{\Sigma}_k^{mm} - \boldsymbol{\Sigma}_k^{mo} (\boldsymbol{\Sigma}_k^{oo})^{-1} \boldsymbol{\Sigma}_k^{mo}), \quad (23)$$

where  $\boldsymbol{\Sigma}_k^{oo}$ ,  $\boldsymbol{\Sigma}_k^{mo}$ ,  $\boldsymbol{\Sigma}_k^{om}$  and  $\boldsymbol{\Sigma}_k^{mm}$  are properly defined in Eq. 18 for each sample and  $C_i^o = \frac{(\mathbf{x}_i^o - \boldsymbol{\mu}_k^o)^T (\boldsymbol{\Sigma}_k^{oo})^{-1} (\mathbf{x}_i^o - \boldsymbol{\mu}_k^o)}{d^o - 2}$ .

As for the classical EM for GMM with missing data (Ghahramani and Jordan, 1994a; Eirola et al., 2014), the missing components of each  $\mathbf{x}_i$  are replaced by the conditional mean  $\boldsymbol{\mu}_{ik}^m$ , along with the computation of the quantity  $\tilde{\boldsymbol{\Sigma}}_{ik}$  given by

$$\tilde{\mathbf{x}}_i = (\mathbf{x}_i^o, \boldsymbol{\mu}_{ik}^m) \quad (24)$$

$$\tilde{\boldsymbol{\Sigma}}_{ik} = \begin{pmatrix} \mathbf{0}^{oo} & \mathbf{0}^{om} \\ \mathbf{0}^{mo} & \boldsymbol{\Sigma}_{ik}^{mm} \end{pmatrix} \quad (25)$$

where  $\mathbf{0}^{oo}$ ,  $\mathbf{0}^{om}$  and  $\mathbf{0}^{mo}$  are matrices of zeros of appropriate dimensions. Note that the tilde symbol is used here to highlight the terms related to the conditional expectations computed in Eq. 22 and Eq. 23, similarly to Eirola et al. (2014).

### 3.2 The M-step

Maximizing the log-likelihood leads to the following expressions for  $\boldsymbol{\mu}_k$ ,  $\boldsymbol{\Sigma}_k$  and  $\pi_k$  defined thanks to (classical) fixed-point equations:

$$\boldsymbol{\mu}_k = \sum_{i=1}^N \frac{w_{ik} p_{ik} \tilde{\mathbf{x}}_i}{\sum_{l=1}^N w_{lk} p_{lk}}, \quad (26)$$

$$\boldsymbol{\Sigma}_k = m \sum_{i=1}^N \frac{p_{ik}}{\sum_{l=1}^N p_{lk}} \left( \frac{\tilde{\boldsymbol{\Sigma}}_{ik}}{\text{tr}(\boldsymbol{\Sigma}_k^{-1} \tilde{\boldsymbol{\Sigma}}_{ik})} + w_{ik} (\tilde{\mathbf{x}}_i - \boldsymbol{\mu}_k)(\tilde{\mathbf{x}}_i - \boldsymbol{\mu}_k)^T \right), \quad (27)$$

$$\pi_k = \frac{1}{N} \sum_{i=1}^N p_{ik}, \quad (28)$$

where  $w_{ik} = \frac{1}{(\tilde{\mathbf{x}}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\tilde{\mathbf{x}}_i - \boldsymbol{\mu}_k)}$ .

These expressions are intuitive, in the sense that they follow the same logic as the EM for GMM with missing data. Indeed, in the M-step, missing values are replaced by their imputed values in  $\tilde{\mathbf{x}}_i$ , and covariances matrices are updated using an additional term taking into account the missing values. One can also notice that those expressions are fixed-point equations as often in robust approaches (*e.g.*, for  $M$ - or FEM estimators).

Finally, note that the scale of  $\boldsymbol{\Sigma}_k$  has no influence on the estimation of missing values (*i.e.*, the scale of  $\boldsymbol{\Sigma}_k$  does not change the result in Eq. 22). This is an important result since, as explained in subsection 2.2, the scatter matrix  $\boldsymbol{\Sigma}_k$  is equal to the covariance matrix up to a scale factor.

### 3.3 Proposed algorithm

The proposed generalized FEM algorithm is detailed in Algorithm 2. As for Algorithm 1, a more efficient way to initialize  $\boldsymbol{\theta}^{(0)}$  when compared to a random initialization is to use the K-means algorithm. In that case, a first imputation of the missing values is needed, for instance using imputations based on the mean or  $k$ -nearest neighbors. In the presence of missing data, the EM algorithm comes with a higher computational cost (it is true for GMM as well) mainly because one needs to evaluate  $(\boldsymbol{\Sigma}_k^{oo})^{-1}$  for each sample with missing data. See Delalleau et al. (2018) for an interesting discussion regarding this issue.

---

**Algorithm 2** Scheme of the generalized FEM algorithm in the the incomplete data case.

---

**Input:** Data  $\{\mathbf{x}_i\}_{i=1}^n$ ,  $K$  the number of clusters

**Output:** Clustering labels  $\mathcal{Z} = \{z_i\}_{i=1}^n$ , model parameters  $\boldsymbol{\theta} = \{\pi_1, \dots, \pi_k, \boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_1, \dots, \boldsymbol{\Sigma}_K\}$  and imputed samples  $\tilde{\mathbf{x}}_i$

- 1: For each sample, identify observed and missing components  $o$  and  $m$ ;
  - 2: Initialize  $\boldsymbol{\theta}^{(0)}$ ;
  - 3:  $t \leftarrow 1$
  - 4: **while** not convergence **do**
  - 5:   **for**  $1 \leq k \leq K$ : **do** ▷ **E-step**
  - 6:      $p_{ik}^{(t)} = \frac{\pi_k^{(t-1)} \det(\boldsymbol{\Sigma}_k^{oo(t-1)})^{-1/2} [(\mathbf{x}_i^o - \boldsymbol{\mu}_k^{o(t-1)})^T (\boldsymbol{\Sigma}_k^{oo(t-1)})^{-1} (\mathbf{x}_i^o - \boldsymbol{\mu}_k^{o(t-1)})]^{-d^o/2}}{\sum_{j=1}^K \pi_j \det(\boldsymbol{\Sigma}_j^{oo(t-1)})^{-1/2} [(\mathbf{x}_i^o - \boldsymbol{\mu}_j^{o(t-1)})^T (\boldsymbol{\Sigma}_j^{oo(t-1)})^{-1} (\mathbf{x}_i^o - \boldsymbol{\mu}_j^{o(t-1)})]^{-d^o/2}}$  ▷ Compute observed responsibilities:
  - 7:      $\boldsymbol{\mu}_{ik}^{m(t)} = \boldsymbol{\mu}_k^{m(t-1)} + \boldsymbol{\Sigma}_k^{mo(t-1)} (\boldsymbol{\Sigma}_k^{oo(t-1)})^{-1} (\mathbf{x}_i^o - \boldsymbol{\mu}_k^{o(t-1)})$  ▷ Compute conditional expectations:
  - 8:      $\boldsymbol{\Sigma}_{ik}^{mm(t)} = \frac{(\mathbf{x}_i^o - \boldsymbol{\mu}_k^{o(t)})^T (\boldsymbol{\Sigma}_k^{oo(t)})^{-1} (\mathbf{x}_i^o - \boldsymbol{\mu}_k^{o(t)})}{d^o - 2} \times (\boldsymbol{\Sigma}_k^{mm(t)} - \boldsymbol{\Sigma}_k^{mo(t)} (\boldsymbol{\Sigma}_k^{oo(t)})^{-1} \boldsymbol{\Sigma}_k^{mo(t)})$
  - 9:     Fill in:  $\tilde{\mathbf{x}}_{ik}^{(t)} \leftarrow [x_i^o, \boldsymbol{\mu}_{ik}^{m(t)}]$  and  $\tilde{\boldsymbol{\Sigma}}_{ik}^{(t)} \leftarrow \begin{pmatrix} \mathbf{0}^{oo} & \mathbf{0}^{om} \\ \mathbf{0}^{mo} & \boldsymbol{\Sigma}_{ik}^{mm(t)} \end{pmatrix}$
  - 10:   **end for**
  - 11:   **for**  $1 \leq k \leq K$ : **do** ▷ **M-step**
  - 12:      $\pi_k^{(t)} = \frac{1}{N} \sum_{i=1}^N p_{ik}^{(t)}$
  - 13:     set  $\boldsymbol{\mu}'_k = \boldsymbol{\mu}_k^{(t-1)}$  and  $\boldsymbol{\Sigma}'_k = \boldsymbol{\Sigma}_k^{(t-1)}$
  - 14:     **while** not convergence **do** ▷ Fixed-point loop
  - 15:        $w_{ik} = \frac{1}{(\tilde{\mathbf{x}}_{ik}^{(t)} - \boldsymbol{\mu}'_k)^T (\boldsymbol{\Sigma}'_k)^{-1} (\tilde{\mathbf{x}}_{ik}^{(t)} - \boldsymbol{\mu}'_k)}$ ,
  - 16:        $\boldsymbol{\mu}''_k = \sum_{i=1}^N \frac{w_{ik} p_{ik}^{(t)} \tilde{\mathbf{x}}_{ik}^{(t)}}{\sum_{i=1}^N w_{ik} p_{ik}^{(t)}}$ ,
  - 17:        $\boldsymbol{\Sigma}''_k = m \sum_{i=1}^N \frac{p_{ik}^{(t)}}{\sum_{i=1}^N p_{ik}^{(t)}} \left( \frac{\tilde{\boldsymbol{\Sigma}}_{ik}^{(t)}}{\text{tr}((\boldsymbol{\Sigma}'_k)^{-1} \tilde{\boldsymbol{\Sigma}}_{ik}^{(t)})} + w_{ik} (\tilde{\mathbf{x}}_i^{(t)} - \boldsymbol{\mu}'_k) (\tilde{\mathbf{x}}_i^{(t)} - \boldsymbol{\mu}'_k)^T \right)$ ;
  - 18:       Update:  $\boldsymbol{\mu}'_k = \boldsymbol{\mu}''_k$  and  $\boldsymbol{\Sigma}'_k = \boldsymbol{\Sigma}''_k$
  - 19:     **end while**
  - 20:     Update:  $\boldsymbol{\mu}_k^{(t)} = \boldsymbol{\mu}''_k$  and  $\boldsymbol{\Sigma}_k^{(t)} = \boldsymbol{\Sigma}''_k$
  - 21:   **end for**
  - 22:    $t \leftarrow t + 1$
  - 23: **end while**
  - 24: Set  $z_i$  as the index  $k$  that has the maximum  $p_{ik}$ .
  - 25: **Final imputation:**  $\tilde{\mathbf{x}}_i^m = \sum_{k=1}^K p_{ik}^{(t)} \tilde{\mathbf{x}}_{ik}^{(t)}$
-

## 4 Numerical results

This section evaluates the proposed method for the reconstruction of missing data coming from various datasets used in biomedical analysis and remote sensing.

### 4.1 Experimental setup

In the following experiments, we assume that the missing data mechanism can be ignored, *i.e.*, we consider the missing values to be MCAR or MAR (see discussion in [Section 3](#)). The imputation performance is quantitatively evaluated using the Mean Absolute Percentage Error (MAPE), which is convenient to use and interpret and can be computed with features having different magnitudes. The MAPE is defined as follows:

$$\text{MAPE} = \frac{100}{N_m} \sum_{i=1}^{N_m} \frac{|f_i - \hat{f}_i|}{|f_i|}, \quad (29)$$

where  $N_m$  is the number of missing features,  $f_i$  is the actual value of the  $i$ th feature and  $\hat{f}_i$  its estimation (also known as imputation or reconstruction). Other metrics, such as the mean absolute error (MAE) or the root mean squared error (RMSE) were also tested but are not reported here since they lead to similar conclusions (with a simpler interpretation for MAPE). Finally, note that when the features are very close to zero, the MAPE metric should be carefully used (see for instance the discussion on the abalone dataset).

The FEM algorithm is compared to five other imputation methods, namely the  $k$ -nearest neighbor (KNN) imputation ([Troyanskaya et al., 2001](#)), the Multiple Imputation by Chained Equations (MICE) ([van Buuren and Groothuis-Oudshoorn, 2011](#)), the MissForest imputation algorithm ([Stekhoven and Buhlmann, 2011](#)), the GMM method ([Dempster et al., 1977](#)) and a robust version of GMM ([Mouret et al., 2022](#)). The robust GMM algorithm uses an outlier detection algorithm (namely, the isolation forest algorithm ([Liu et al., 2012](#))) within the EM algorithm to reduce the influence of outlier samples in the estimation of the mixture model (see the original paper [Mouret et al. \(2022\)](#) for more details and derivations). For KNN, MICE and MissForest algorithms, we used the Python library *scikit-learn* ([Pedregosa et al., 2011](#)) (version 0.24.2), whereas we have implemented our own EM algorithm for GMM, robust GMM and FEM. A minimal parameter tuning was considered for the different algorithms. To that extent, KNN, MICE, MissForest algorithms were used with their default parameters: the number of neighbors was set to 5 for the KNN algorithm, the MICE algorithm uses the IterativeImputer with BayesianRidge estimators and the MissForest uses the IterativeImputer with ExtraTree regressor. The parameters of the GMM algorithms were adjusted as in [Mouret et al. \(2022\)](#) (in brief, only a small regularization of the covariances matrices is applied to avoid instabilities) and the FEM algorithm was used without any tuning (*i.e.*, no regularization was used for the covariance matrices). Finally, the number of components  $K$  used for the GMM and FEM depends on the datasets (it is fixed when the number of classes is known, otherwise it is estimated using the Bayesian Information Criterion, see results on the abalone dataset for details).

In a first experiment, two synthetic datasets are considered to evaluate a change in the underlying data distribution (*i.e.*, Gaussian and non-Gaussian distributions). In a second step, imputation tasks are conducted on different real world datasets. We mostly used benchmark datasets coming from the University of California at Irvine (UCI) database<sup>2</sup>. To that extent, the results presented here can be easily reproduced. The name, number of attributes and number of samples of each dataset are summarized in [Table 1](#). For all these datasets, 50 Monte Carlo (MC) simulations were conducted by varying the percentage of missing data and the percentage of outliers added to the dataset. The outliers are generated using uniform distributions on intervals defined by the minimum and maximum of each feature.

### 4.2 Synthetic data

The proposed imputation algorithm is first tested on synthetic data. The conducted experiments can be summarized as follows: 1) generation of synthetic data with  $N = 2000$  (number of samples) and  $m = 10$  (number of features), 2) scaling of the generated data (this allows us to use the MAPE metric without problems), 3) random generation of missing values (the missing elements of the feature matrix are chosen randomly), 4) imputation (and metric computation). The datasets are

---

<sup>2</sup><https://archive.ics.uci.edu>

Table 1: Numbers of features and samples of the datasets used to evaluate the different imputation techniques. UCI is added in parenthesis for the datasets coming from the UCI database.

Dataset	Features	Samples	Reference
Synthetic (AR(1) time series)	10	2000	-
Mice protein expression (UCI)	82	1080	<a href="#">Higuera et al. (2015)</a>
Abalone (UCI)	8	4176	<a href="#">Nash et al. (1994)</a>
Statlog - Landsat Satellite (UCI)	36	4435 (train)	-
Rapeseed crops - Sentinel satellites	106	2218	<a href="#">Mouret et al. (2021)</a>

generated according to a mixture of  $K = 3$  distributions (all classes are equally likely represented). The generation of these datasets is summarized in [Table 2](#) (which provides the distributions used for the datasets) and [Table 3](#) (which presents the generation of the synthetic samples). Note that after having generated the synthetic samples, the dataset is scaled so that the minimum value of the whole dataset is equal to 1 and the 98th<sup>3</sup> percentile of the whole dataset is equal to 100 (this allow us to avoid problems with the MAPE metrics when values are close to zero).

Table 2: Summary of the parameters used to generate the synthetic samples, where  $\mathcal{U}$  is the uniform distribution.

$$\begin{aligned}
 & \text{Parameters} \\
 & \hline
 & \mu_{kf} \sim \mathcal{U}[0, 1], \forall f \text{ with } f \text{ feature \#} \\
 & \hline
 & \phi_k \sim \mathcal{U}[0.1, 0.9] \\
 & \sigma_k^2 \sim \mathcal{U}[0.0005, 0.005] \\
 & \Sigma_k = \frac{\sigma_k^2}{1-\phi_k^2} \begin{pmatrix} 1 & \phi_k & \phi_k^2 & \dots & \phi_k^{m-1} \\ \phi_k & 1 & \phi_k & \dots & \phi_k^{m-1} \\ \phi_k^2 & \phi_k & 1 & \dots & \phi_k^{m-1} \\ \dots & \dots & \dots & \dots & \dots \\ \phi_k^{m-1} & \dots & \phi_k^2 & \phi_k & 1 \end{pmatrix}
 \end{aligned}$$

Table 3: Distributions used to generate the two synthetic datasets, which make use of the parameters provided in [Table 2](#), where  $t_5$  is the multivariate  $t$ -distribution with 5 degrees of freedom.

Name	Generating distribution	Classes
Gaussian dataset	$\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$	3
Student dataset	$t_5(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$	3

The various synthetic datasets can be viewed as AR(1) time series, which are grouped into 3 different clusters. Two representative examples generated according to the Gaussian and Student’s  $t$ -distributions are displayed in [Figure 1](#). One can observe that the Student dataset logically contains more samples with extreme values. Because of these extreme values, the range of the MAPE is different for these two types of datasets (since the data have been scaled).

Imputation results obtained for the two types of datasets are provided in [Figure 2](#) for a percentage of missing values equal to 50%, without outliers (more experiments conducted with different percentages of missing data are provided for the real world datasets). For the Gaussian datasets, the GMM and FEM methods provide the best results, with a slight advantage for the FEM approach. This illustrates the versatility of the FEM algorithm, which is competitive against the GMM imputations, even when it is used for a dataset adapted to Gaussian methods. When considering the Student datasets, FEM still provides the best results and outperforms all the other tested approaches. In that case, the GMM approaches can provide good imputations depending on the dataset generation, but can also lead to poor results. Moreover, the robust GMM does not improve the imputation results for this type of data (this is probably due to a non-optimal tuning of the algorithm parameters used for robust estimation). In both scenarios, the MissForest provides competitive results, but with a higher MAPE than the one obtained using the FEM ap-

<sup>3</sup>This values was chosen so that most of the data (except outliers) are scaled in the range [1, 100].

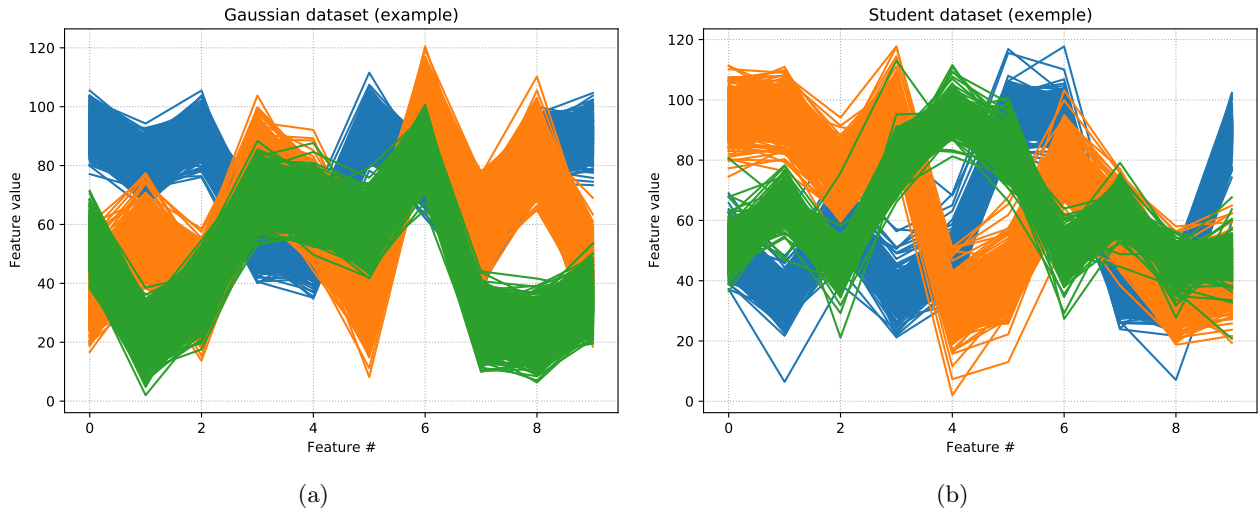


Figure 1: Synthetic datasets generated according to Table 3: a) multivariate Normal distribution, b) multivariate  $t$ -distribution. Each color corresponds to a different class.

proach. Finally, for both types of datasets, MICE and KNN approaches are not competitive, with a MAPE that can be almost twice higher than with FEM imputations.

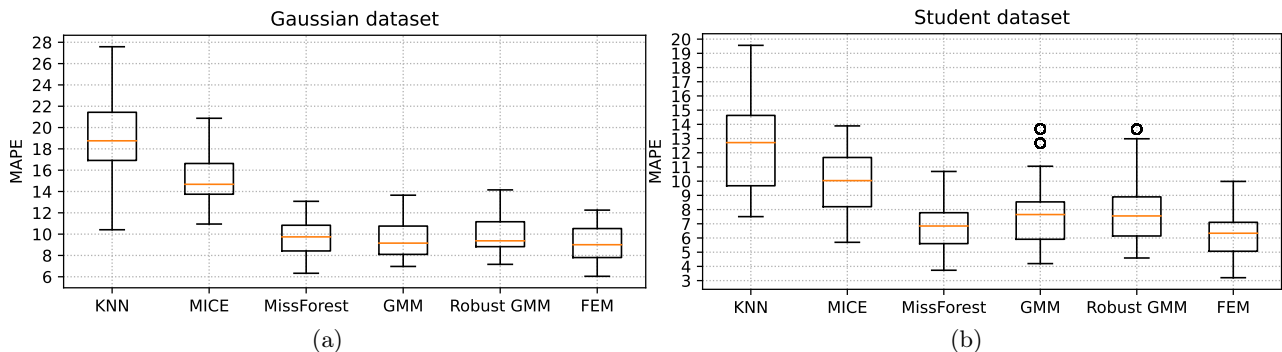


Figure 2: MAPE obtained after 50 MC simulations a) Gaussian and b) Student synthetic datasets (generated randomly for each simulation). The percentage of missing features is 50% for this example.

### 4.3 Real-world experiments

We now proceed with experiments conducted on real-world data sets.

#### 4.3.1 Mice protein expression dataset

The mice protein dataset contains expression levels of various protein or protein modification measures in the cerebral cortex of 8 classes of control and down syndrome mice. This type of data is subject to the missing data problem, which is interesting in the context of this study. In particular, the dataset considered here already contains a small percentage of missing values (around 1%). During the experiments, missing values were simulated completely at random. The number of clusters was fixed to 8 for the GMM and FEM algorithms.

The imputation results obtained with the different methods are summarized in Figure 3. More precisely, Figure 3(a) evaluates the influence of the percentage of missing data whereas Figure 3(b) studies the influence of adding outliers in the dataset. It can be observed that, overall, the FEM and MissForest algorithm outperform all the other tested algorithms. More precisely, for percentages of missing values lower than 35%, the FEM imputation is optimal whereas for higher percentages of missing data the MissForest algorithm provides the best imputation. The GMM approaches are competitive only when the percentage of missing data is lower than 30%, with an important

degradation for higher levels of missing data (in that case the FEM algorithm is much more competitive than the other EM-based algorithms). Moreover, GMM are particularly sensitive to the presence of outliers (for a better visualization, the MAPE obtained for GMM is not fully displayed in **Figure 3(b)** since some values are close to 80%). For this dataset, the KNN and MICE algorithms perform poorly (it is especially true for the MICE algorithm). Finally, **Figure 3(c)** shows a boxplot representation of the MAPE obtained for different MC runs (missing data is set to 40% without outliers). It can be observed that the MissForest algorithm always provides a better MAPE in that configuration and that the MAPE obtained using the FEM algorithm is very close.

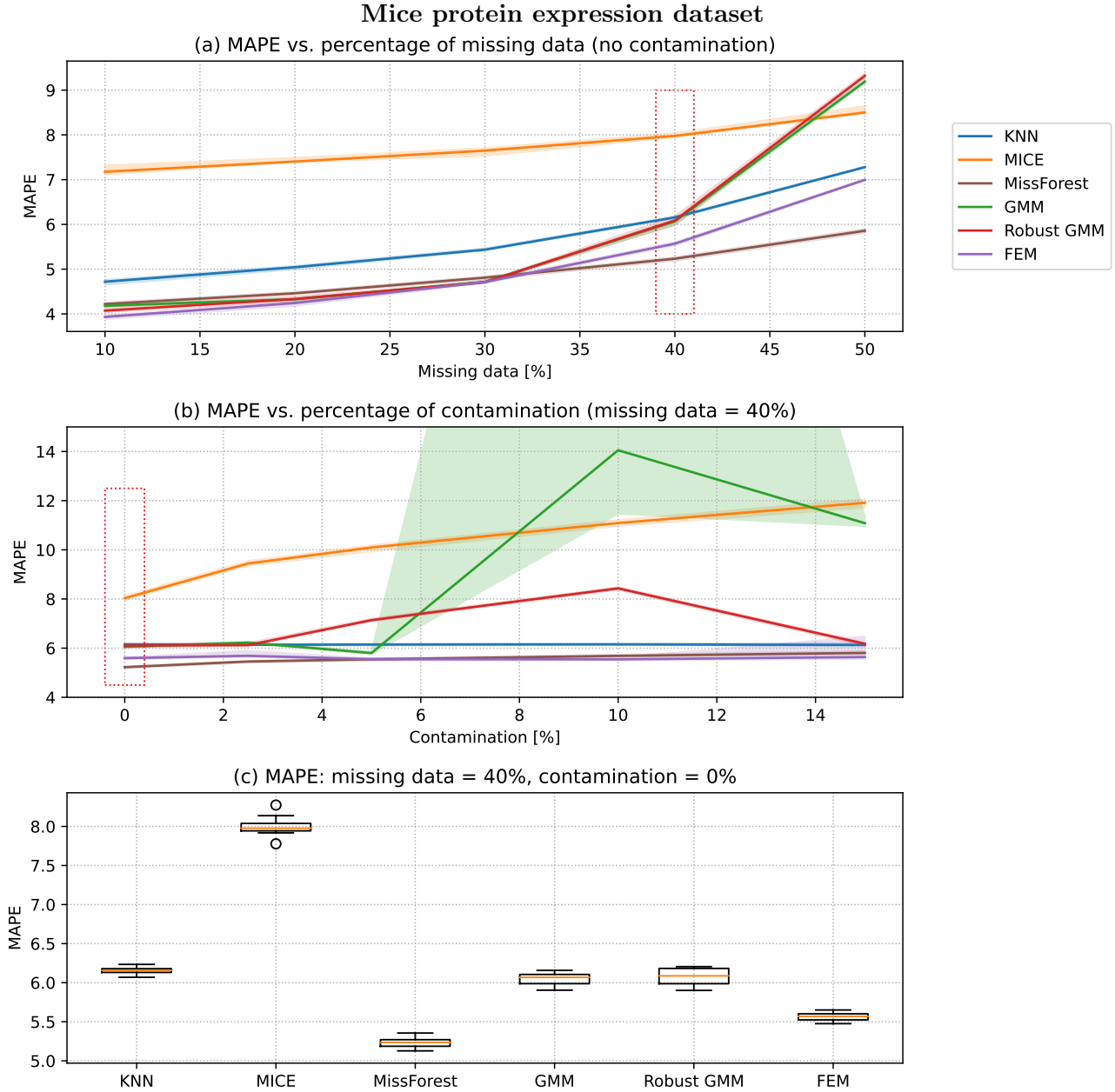


Figure 3: MAPE obtained for the mice protein expression dataset by varying (a) the percentage of missing data (MCAR) and (b) the percentage of outliers in the dataset. The results are obtained after 50 MC simulations (the plain line corresponds to the median and the shaded area is filled between the first and third quartiles). Figure (c) shows a boxplot of the MC runs when the quantity of missing data is set to 40% and there is no outlier, which corresponds to the red rectangle displayed in (a) and (b).

### 4.3.2 Abalone dataset

The abalone dataset consists of various physical measurements (length, diameter, etc.) made on abalone. Note that the feature “sex” is a categorical feature taking the value M, F or I (Infant), which was not considered in this experiment. A simple way to handle this feature would be to convert it into integers. However, this conversion would provide inconsistent results for the MAPE or other reconstruction metrics. Note also that the feature “rings”, which is an integer directly related to the age of the abalone, was kept in the dataset. Because some features can be very close to zero, a scaling of each feature in the range  $[1, 100]$  was made in order to use the MAPE metrics in a relevant way. This scaling makes sense since the aim of these experiments is to compare the different imputation methods. Finally, the optimal number of clusters to be chosen for EM approaches was estimated for each simulation using the Bayesian Information Criterion (BIC), as recommended for instance in [Bouveyron and Brunet-Saumard \(2014\)](#). It is defined as  $BIC = -2 \log(L) + p \log(N)$  (the lower the better), where  $L$  corresponds to the likelihood of a given model,  $p$  is the number of parameters and  $N$  is the number of samples used to fit the GMM parameters. When estimating the number of classes, we observed that the FEM algorithm tends to use efficiently a higher number of components when compared to GMM approaches, which on the contrary tend to be very unstable when the number of components is too high. Further investigations on that topic could be interesting but are out of the scope of this paper.

The results obtained on the abalone dataset are summarized in [Figure 4](#). In brief, most of the conclusions obtained for the mice dataset can be transposed to the abalone dataset. More precisely, the imputations obtained with the FEM algorithm have overall a lower MAPE when compared to the other algorithms. In particular, the FEM algorithm performs well when the percentage of missing data is high or when the data is contaminated by outliers, with a very low dispersion in its results. As an exception, the MissForest provides better results when the percentage of missing data is equal to 50%. This algorithm is, however, more sensitive than the FEM algorithm to the presence of outliers. In absence of outliers, GMM imputations are close to those obtained with the FEM algorithm, but are always sub-optimal (the MAPE is consistently higher of around 0.5%). Finally, the MICE algorithm outperforms the KNN imputation method for this dataset. This example confirms that MICE and KNN algorithms are very sensitive to the considered dataset.

### 4.3.3 Statlog - Landsat satellite data

This classic database contains multispectral pixel values (4 spectral bands) acquired in a  $3 \times 3$  neighborhood region using the Landsat satellite. Each sample is characterized by a total of  $3 \times 3 \times 4 = 36$  features, which belong to a land cover category (6 categories in total, *e.g.*, red soil, cotton crop, etc.). For this dataset, the missing data was simulated by removing all the values of some pixels (*i.e.*, all the spectral bands are missing), which could for instance correspond to a sensor failure. Experimental results obtained on this dataset are summarized in [Figure 5](#). Except for the KNN algorithm, all methods provide decent results when the percentage of missing data is lower than 20% (with a slight and consistent advantage for the FEM algorithm). However, for higher percentages or in the presence of outliers, results obtained with the FEM algorithm are significantly better (with a low dispersion). This confirms the results obtained on the datasets previously tested. As a last remark, we would like to clarify that the relatively poor results obtained with the robust GMM approach might be explained by a non-optimal tuning of the outlier detection mechanisms, whereas the FEM algorithm is not impacted at all by outliers. This illustrates the advantage of the FEM approach, which does not need additional tuning to take into account the presence of outlier samples.



### Abalone dataset

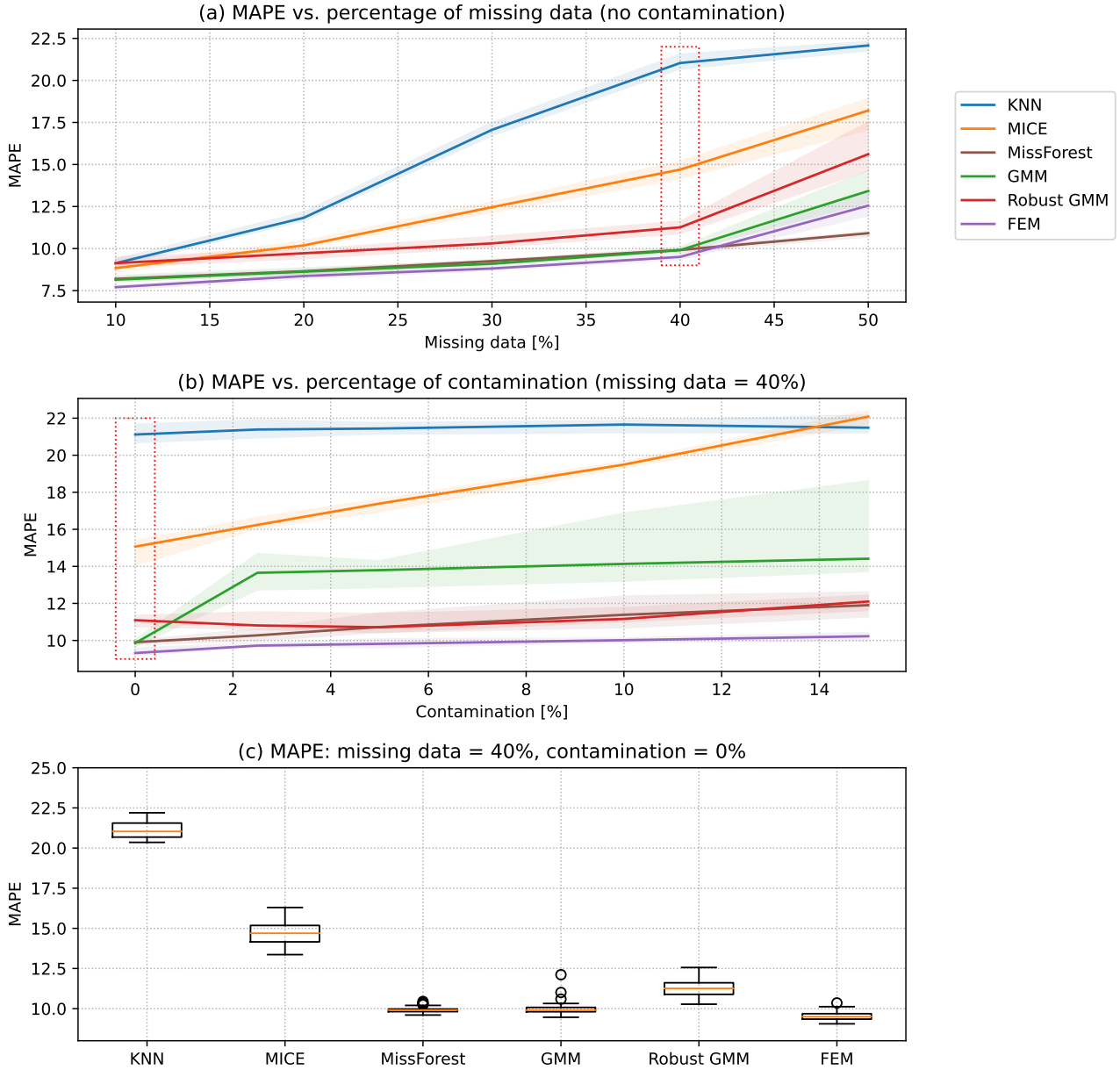


Figure 4: MAPE obtained on the abalone dataset by varying the (a) percentage of missing data (MCAR) and (b) the percentage of outliers in the dataset. The results have been obtained after averaging 50 MCs simulations (the plain line corresponds to the median and the shaded area is filled between the first and third quartiles). Figure (c) displays a boxplot of the MC runs when the quantity of missing data is 40% without outliers added to the dataset, which corresponds to the red rectangles displayed in (a) and (b).

### Landsat dataset

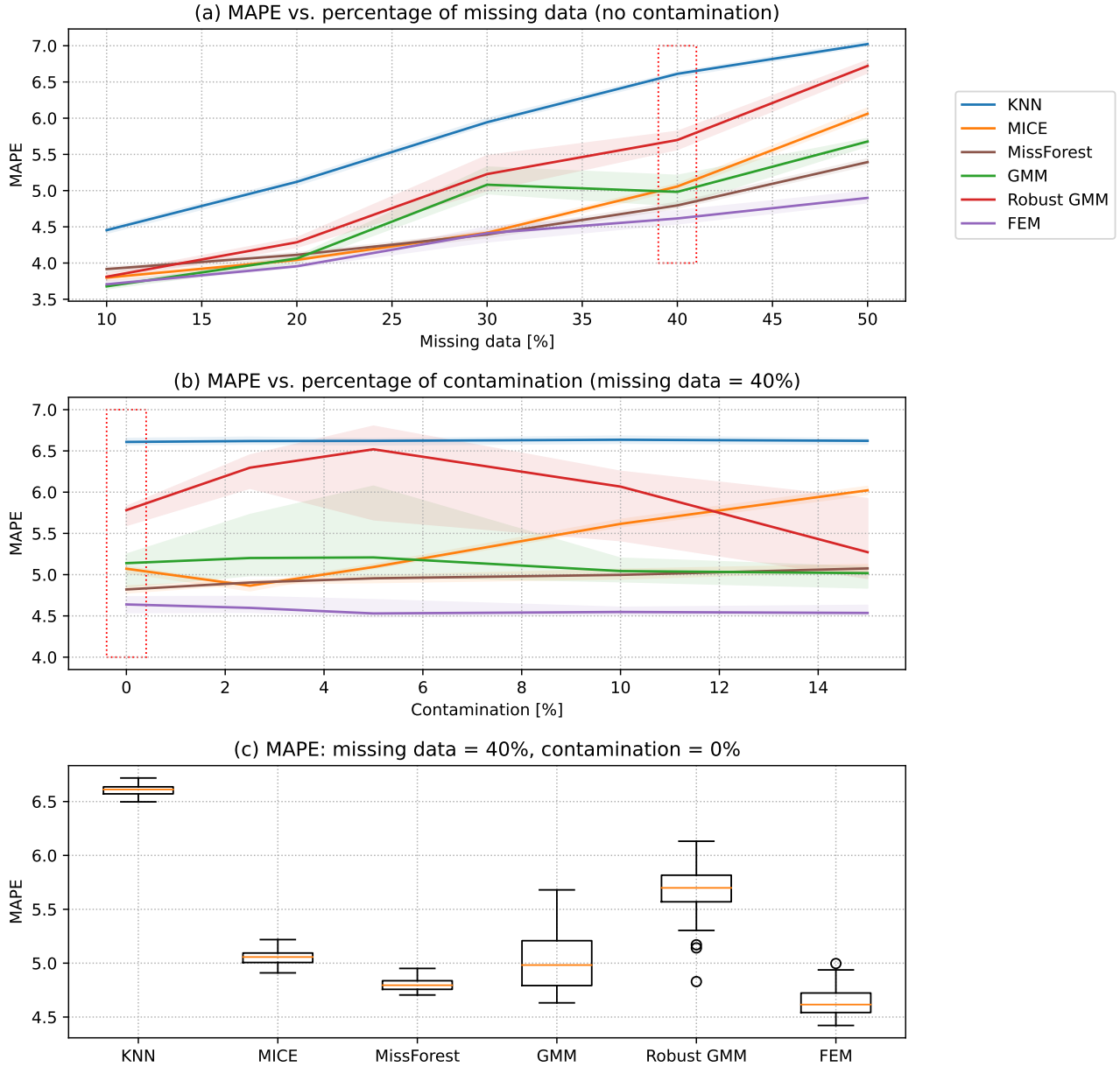


Figure 5: MAPE obtained for the Landsat dataset by varying (a) the percentage of missing data (MCAR) and (b) the percentage of outliers in the dataset. The results have been obtained after averaging the metrics of 50 MCs simulations (the plain line corresponds to the median and the shaded area is filled between the first and third quartiles). Figure (c) shows a boxplot of the MC runs when the quantity of missing data is 40% without outliers in the dataset, which corresponds to the red rectangles plotted in (a) and (b).

#### 4.3.4 Rapeseed crop monitoring - Sentinel satellite data

This section considers remote sensing time series computed for the monitoring of 2218 rapeseed parcels. The time series are obtained using Sentinel-1 (S1) and Sentinel-2 (S2) satellites, which provide synthetic aperture radar (SAR) and multispectral images, respectively. The dataset is subject to missing data, especially because clouds affect multispectral images, which is a known issue in remote sensing (Shen et al., 2015). The time series to be imputed are the median and interquartile range (IQR) of statistics (computed at the parcel-level) of the Normalized difference vegetation index (NDVI), which is a popular agronomic indicator used in remote sensing for agricultural applications. The feature matrix used in these experiments also contains features coming from S1 images, which are not subject to missing data. More precisely, these features are the median (computed at the parcel-level) of the VV and VH backscattering coefficients (see (Mouret et al., 2021) for more details regarding the construction of this feature matrix). To summarize, each rapeseed parcel is characterized by 13 values (each value corresponds to a specific time instant in the growing season of interest) of median NDVI, 13 values of IQR NDVI, 40 values of median VV backscattering and 40 values of VH backscattering (*i.e.*, a total of 106 features).

For this dataset, the missing values were not completely added at random to have more realistic experiments. Indeed, missing data occurs on cloudy days and only affect the multispectral features (here, the NDVI statistics computed at the parcel-level). More precisely, two parameters control the missing data mechanism: the percentage of multispectral images affected by missing data (*i.e.*, the number of cloudy multispectral images), and the percentage of crop parcels with missing data (*i.e.*, generally, only a part of the image is covered by clouds). For each multispectral image with missing data, we fixed the percentage of affected parcels to 50%. The number of mixture components used in GMM and FEM is unknown and was fixed using BIC as for the abalone dataset.

Results computed using 50 MC simulations are summarized in Figure 6, when looking separately at the median NDVI (a,b) and the IQR NDVI (c,d). Some general observations are first provided. The imputation results are more scattered than with the other datasets (one explanation is that some periods of the growing season are more difficult to reconstruct, see Mouret et al. (2022) for more details). However, very good reconstructions of the median NDVI are possible, even with a high percentage of S2 images with missing data, in part due to the use of additional S1 data. Regarding the IQR NDVI, the high values of the MAPE can be explained by 1) the fact that IQR NDVI values are close to zero (*i.e.*, a small imputation error implies a large MAPE) and 2) the fact that IQR NDVI can change abruptly through time and is less correlated to S1 data.

When looking specifically at each algorithm, it appears that, overall, the robust GMM algorithm is the best suited for this dataset, confirming previous results found in Mouret et al. (2022). It can be observed that, as for the other datasets, methods based on the EM algorithm outperform the KNN and MICE methods. The MissForest algorithm performs again well even if it is impacted by outliers, confirming previous results. Finally, the FEM algorithm provides results that are very close to the GMM imputations. This is interesting since the FEM algorithm was used without any deep tuning of its parameters, whereas the robust GMM was mainly designed and tested for this task with an accurate parameter tuning.

## Rapeseed crop monitoring - Sentinel satellite data

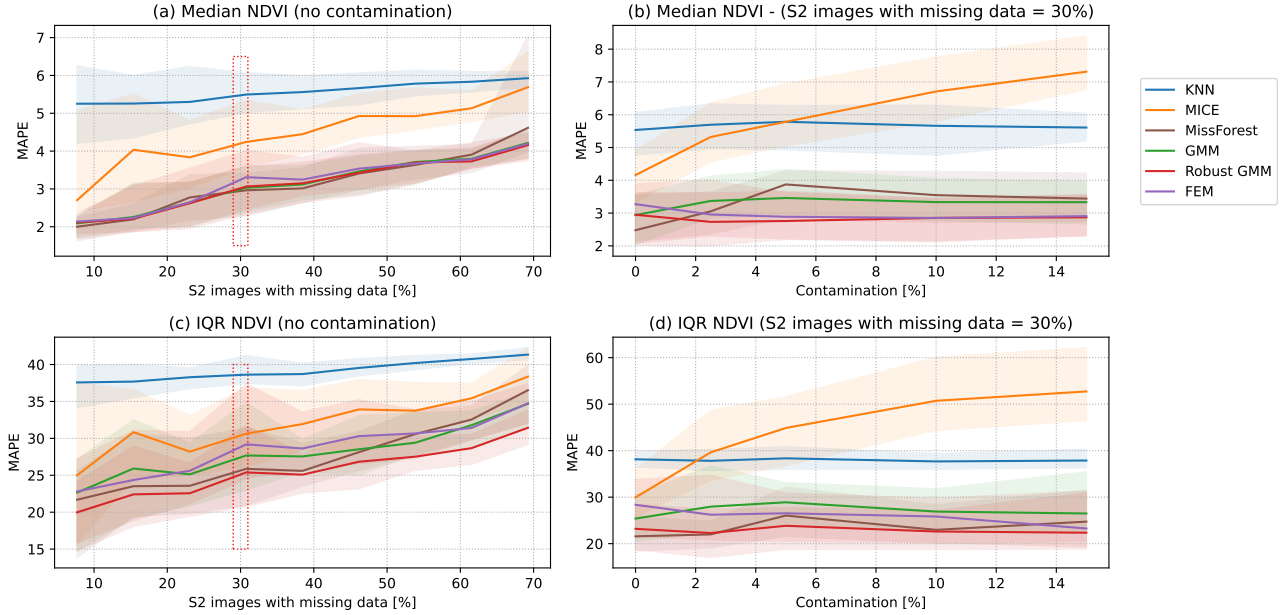


Figure 6: MAPE obtained for the rapeseed dataset by varying (a, c) the percentage of S2 images affected by missing data and (b, d) the percentage of outliers in the dataset (red boxes in (a,c) correspond to the percentage of missing S2 images used for (b,d)). When an S2 image has missing data, 50% of the parcels have their corresponding features missing. The results are obtained after averaging the metrics of 50 MC simulations (the plain line corresponds to the median and the shaded area is filled between the first and third quartiles). Figures (a,b) are obtained using the median NDVI of the parcels whereas (c,d) corresponds to the IQR NDVI of the parcels.

### 4.4 Influence of different types of outliers

Two additional experiments are conducted using the Abalone dataset to 1) evaluate the impact of outliers with a low percentage of missing data (10%) and 2) evaluate the impact of another outlier generation mechanism. More precisely, we used a mechanism similar to the one proposed in Hippert-Ferrer et al. (2022) by adding outliers generated as Gaussian noise (*i.e.*,  $z_i \sim \mathcal{N}(\mu, \text{diag}\{\sigma\})$ , with  $\mu$  a vector whose elements are the mean of each feature and  $\text{diag}\{\sigma\}$  a diagonal matrix whose elements are the variances of each feature). The obtained results are summarized in Figure 7.

Overall, two main conclusions can be drawn. First, even with a low percentage of missing data, imputation results obtained with the FEM algorithm are very competitive. Note that even with a small amount of missing data, the impact of outliers can be important depending on some algorithms (*e.g.*, classical GMM). Secondly, changing the outlier generation mechanism has an impact on the imputation results. In particular, the robust GMM algorithm is more impacted by Gaussian white noise, while it is not the case with uniform white noise. Overall, these results confirm that the FEM algorithm is almost not impacted by outliers when changing the percentage of missing data and provide competitive results when compared to the state-of-the-art.

## Abalone dataset

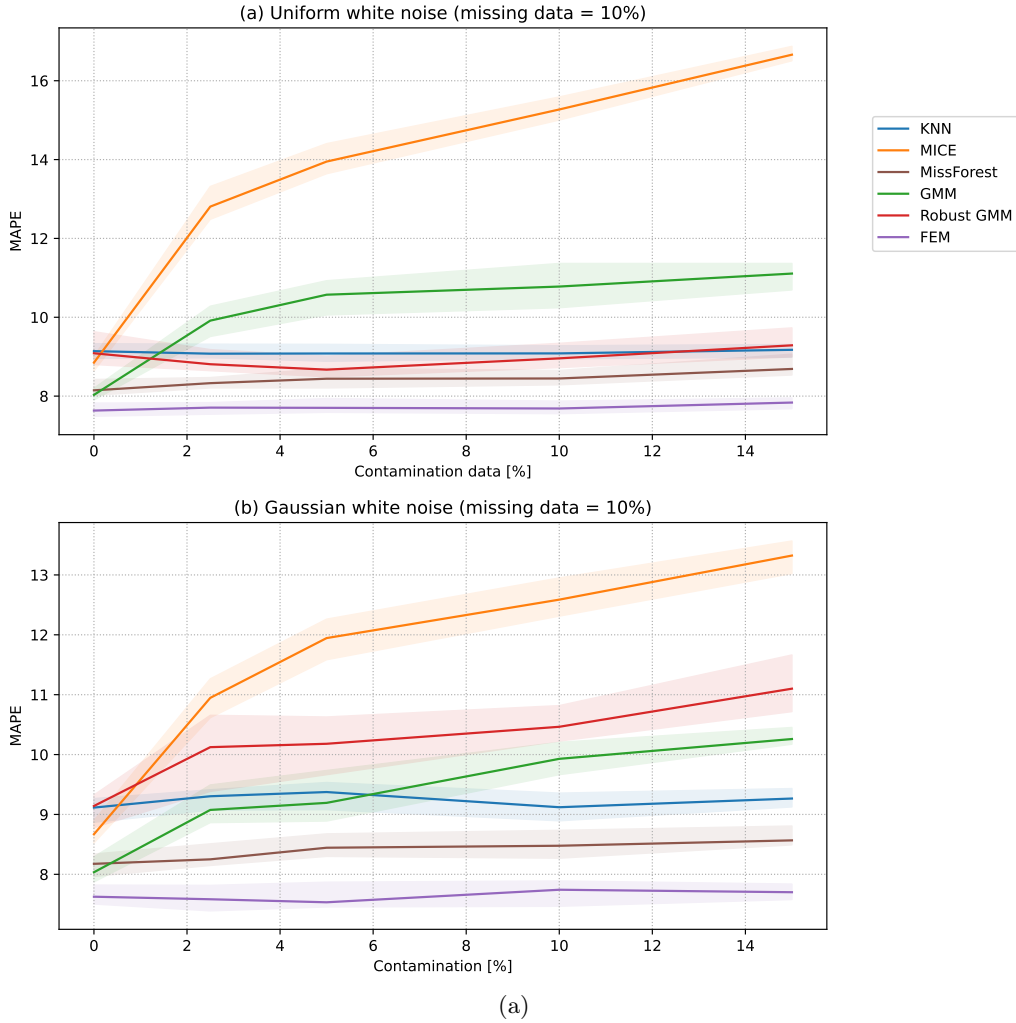


Figure 7: MAPE obtained for the Abalone dataset by varying the percentage of outliers in the dataset, with a percentage of missing data equal to 10%. Outliers are generated according to (a) a uniform distribution taking values between the minimum and maximum of the features and (b) a Gaussian noise (*i.e.*,  $z_i \sim \mathcal{N}(\boldsymbol{\mu}, \text{diag}\{\boldsymbol{\sigma}\})$ ), with  $\boldsymbol{\mu}$  a vector whose element are the mean of the features and  $\text{diag}\{\boldsymbol{\sigma}\}$  a diagonal matrix whose element are the variances of the features. The results have been obtained after averaging the metrics of 50 Monte Carlo simulations (the plain line corresponds to the median and the shaded area is filled between the first and third quartiles).

## 5 Conclusion

This paper proposed to extend the flexible EM (FEM) algorithm of Roizman et al. (2020, 2021) to handle missing data. The algorithm is flexible in the sense that it is 1) robust to outliers and 2) adapted to any mixture of elliptical distributions (*i.e.*, the data distribution is not necessarily Gaussian). As a consequence, the FEM algorithm can be used for a wide range of datasets, unlike the classical EM for Gaussian mixture models, which is impacted by noise and non-Gaussian distributions. The main theoretical contribution of this paper is to derive, in the presence of missing data, an EM algorithm which assumes that the data has been generated from a mixture of (unknown) elliptical distributions having the same density generator. As in the complete-data case, the FEM algorithm derived in the presence of missing data is intuitive and can be used with little parameter tuning.

The main focus of this paper is the imputation of missing data. Imputation results obtained using the FEM algorithm were compared with 5 other benchmark algorithms, based on KNN, MICE, MissForst, GMM and robust GMM. From all the experiments presented in this study, two main conclusions can be drawn. First, it was observed that the FEM algorithm was competitive for all considered datasets, generally outperforming all other tested methods when the percentage of missing data is high or when outliers are contaminating the dataset. To that extent, the experimental results confirm the theoretical robust properties of the algorithm and illustrate the flexibility of the proposed algorithm when compared to the other tested methods, which may fail depending on the considered dataset.

Using the FEM algorithm for outlier detection and classification tasks (potentially with missing values) is an interesting prospect. Other perspectives are related to the regularization of the scatter matrix, which can be complicated to estimate with high dimensional data (Bouveyron and Brunet-Saumard, 2014). Various regularization strategies used for GMM could be investigated for FEM models, such as the approaches proposed in Bouveyron et al. (2007) to regularize the eigenvalues of the covariance matrices or the  $l - 1$  constrained graphical lasso algorithm (Friedman et al., 2008), which has been extended to GMM with missing data in Ruan et al. (2011).

## Declarations

- Funding: This document is the result of a research project funded by TerraNIS SAS and ANRT (convention CIFRE no. 2018/1349)
- Conflict of interest/Competing interests: The authors have no competing interest to declare that is relevant to the content of this article.
- Ethics approval: not applicable
- Consent to participate: not applicable
- Consent for publication: not applicable
- Availability of data and materials: We mostly used benchmark datasets coming from the University of California at Irvine (UCI) database (<https://archive.ics.uci.edu>).
- Code availability: the implementation in Python of the algorithm is available at the repository [https://github.com/fmouret/flexible\\_em\\_imputation](https://github.com/fmouret/flexible_em_imputation).
- Authors' contributions: All authors contributed to the study conception and design. The first draft of the manuscript was written by Florian Mouret and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

## Appendix A Proofs

### A.1 Independence of $E[\ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}) | z_{ik}, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o]$ from the model parameters $(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$

This appendix shows the independence of  $E[\ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}) | z_{ik}, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o]$  from the model parameters  $(\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ . From (6), the expectation of the complete log-likelihood knowing the model parameters and the observed variables can be written as follows:

$$E \left[ \log \mathcal{L}_c(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) | \boldsymbol{\theta}^{(t)}, \mathcal{X}^o \right] = \sum_{i=1}^N \sum_{k=1}^K E \left[ z_{ik} \ell_{0k}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) + z_{ik} \ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}) | \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right], \quad (30)$$

where, similarly to the complete data case, we define  $\ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}) = \log(s_{ik}^{d/2} g_i(s_{ik}))$  with  $s_{ik} = \frac{(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)}{\tau_{ik}}$ . Using the fact that  $z_{ik}$  is a binary indicator, (30) can be decomposed as follows:

$$\begin{aligned} E \left[ \log \mathcal{L}_c(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) | \boldsymbol{\theta}^{(t)}, \mathcal{X}^o \right] &= \sum_{i=1}^N \sum_{k=1}^K E \left[ z_{ik} | \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right] E \left[ \ell_{0k}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) | z_{ik} = 1, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right] \\ &\quad + \sum_{i=1}^N \sum_{k=1}^K E \left[ z_{ik} | \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right] E \left[ \ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}) | z_{ik} = 1, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right]. \end{aligned} \quad (31)$$

Since the first term of this expression does not depend on  $\tau_{ik}$ , the maximization of (31) w.r.t.  $\tau_{ik}$  reduces to maximize the following function:

$$E \left[ \ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}) | z_{ik} = 1, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right] = E \left[ \log \left( s_{ik}^{d/2} g_i(s_{ik}) \right) | z_{ik} = 1, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right]. \quad (32)$$

Consider the one dimensional function  $f$  defined by  $f(t) = t^{d/2} g(t)$ ,  $t \in \mathbb{R}$ . The maximum of this function is denoted as

$$t^* = \arg \sup_t \{f(t)\},$$

which is a constant independent of  $\pi_k, \boldsymbol{\mu}_k$  and  $\boldsymbol{\Sigma}_k$ . Moreover, the supremum of the function  $f$  is well-defined and is denoted as  $\sup\{f(t^*)\}$ . Using the fact that  $\log$  is an increasing function, one has

$$\log \left( s_{ik}^{d/2} g_i(s_{ik}) \right) \leq \log[\sup\{f(t^*)\}],$$

which implies

$$E \left[ \ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}) | z_{ik} = 1, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right] \leq \log[\sup\{f(t^*)\}].$$

Moreover, define

$$\tau_{ik}^* = \frac{(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Sigma}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)}{t^*}.$$

Replacing  $\tau_{ik}$  by  $\tau_{ik}^*$  in (32) leads to

$$E \left[ \ell_{ik}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k, \tau_{ik}^*) | z_{ik} = 1, \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right] = \log[\sup\{f(t^*)\}]. \quad (33)$$

This shows that the conditional expectation is maximized for  $\tau_{ik} = \tau_{ik}^*$  and that this maximum does not depend on the model parameters  $\pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$  and the missing data. Thus, maximizing  $E \left[ \log \mathcal{L}_c(\boldsymbol{\theta}; \mathcal{X}, \mathcal{Z}) | \boldsymbol{\theta}^{(t)}, \mathcal{X}^o \right]$  w.r.t.  $\boldsymbol{\theta}$  and  $\mathcal{X}^o$  is equivalent to maximizing the AG part of the log-likelihood, i.e.,  $\sum_{i=1}^N \sum_{k=1}^K E \left[ z_{ik} \ell_{0k}(\mathbf{x}_i; \pi_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) | \boldsymbol{\theta}^{(t)}, \mathbf{x}_i^o \right]$ , which concludes the proof.

## A.2 Conditional distribution of an Angular Gaussian distribution

This appendix derives the conditional density  $f_{i, \boldsymbol{\theta}_k}(\mathbf{x}_2 | \mathbf{x}_1)$ . For brevity we denote  $\mathbf{x}_i = \mathbf{x}$ ,  $\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}$  and  $\boldsymbol{\mu}_k = \boldsymbol{\mu}$  in the following. By definition of a conditional pdf, we have

$$f_{\boldsymbol{\theta}}(\mathbf{x}_2 | \mathbf{x}_1) = \frac{f_{\boldsymbol{\theta}}(\mathbf{x}_1, \mathbf{x}_2)}{f_{\boldsymbol{\theta}}(\mathbf{x}_1)}, \quad (34)$$

i.e.,

$$f_{\boldsymbol{\theta}}(\mathbf{x}_2 | \mathbf{x}_1) \propto \frac{\det(\boldsymbol{\Sigma})^{-1/2} [(\mathbf{x} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})]^{-d/2}}{\det(\boldsymbol{\Sigma}_{11})^{-1/2} [(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1)]^{-d_1/2}} \quad (35)$$

where  $d_1$  is the number of features in  $\mathbf{x}_1$ .

The determinant of the matrix  $\boldsymbol{\Sigma}$  can be decomposed as

$$\det(\boldsymbol{\Sigma}^{-1/2}) = [\det(\boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21} \boldsymbol{\Sigma}_{11}^{-1} \boldsymbol{\Sigma}_{12}) \det(\boldsymbol{\Sigma}_{11})]^{-1/2}. \quad (36)$$

Moreover, by using standard manipulations on matrices, we obtain

$$\begin{aligned} [(\mathbf{x} - \boldsymbol{\mu}) \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})]^{-d/2} &= [(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \\ &\quad + (\mathbf{x}_2 - \boldsymbol{\mu}_{2.1})^T \boldsymbol{\Sigma}_{22.1}^{-1} (\mathbf{x}_2 - \boldsymbol{\mu}_{2.1})]^{-d/2}, \end{aligned} \quad (37)$$

with  $\boldsymbol{\mu}_{2.1} = \boldsymbol{\mu}_2 - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}(\mathbf{x}_1 - \boldsymbol{\mu}_1)$  and  $\boldsymbol{\Sigma}_{2.1} = \boldsymbol{\Sigma}_{22} - \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}\boldsymbol{\Sigma}_{12}$ . Thus plugin Eq. 37 into Eq. 35 gives:

$$f_{\boldsymbol{\theta}}(\mathbf{x}_2|\mathbf{x}_1) \propto \frac{\det(\boldsymbol{\Sigma}_{22.1})^{-1/2} \left[ (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) + (\mathbf{x}_2 - \boldsymbol{\mu}_{2.1})^T \boldsymbol{\Sigma}_{22.1}^{-1} (\mathbf{x}_2 - \boldsymbol{\mu}_{2.1}) \right]^{-d/2}}{\left[ (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \right]^{-d_1/2}} \quad (38)$$

$$\propto \frac{\det(\boldsymbol{\Sigma}_{22.1})^{-1/2} \left[ (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \right]^{-d/2} \left[ 1 + \frac{(\mathbf{x}_2 - \boldsymbol{\mu}_{2.1})^T \boldsymbol{\Sigma}_{22.1}^{-1} (\mathbf{x}_2 - \boldsymbol{\mu}_{2.1})}{(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1)} \right]^{-d/2}}{\left[ (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \right]^{-d_1/2}} \quad (39)$$

$$\propto \frac{\det(\boldsymbol{\Sigma}_{22.1})^{-1/2} \left[ 1 + \frac{(\mathbf{x}_2 - \boldsymbol{\mu}_{2.1})^T \boldsymbol{\Sigma}_{22.1}^{-1} (\mathbf{x}_2 - \boldsymbol{\mu}_{2.1})}{(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1)} \right]^{-d/2}}{\left[ (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \right]^{\frac{d-d_1}{2}}}. \quad (40)$$

Looking at the numerator of Eq. 40, a multivariate  $t$ -distribution can be identified since

$$f_{\text{student}}(\mathbf{x}_2) \propto \left[ 1 + \frac{1}{\nu} (\mathbf{x}_2 - \boldsymbol{\mu}_{\text{student}})^T \boldsymbol{\Sigma}_{\text{student}}^{-1} (\mathbf{x}_2 - \boldsymbol{\mu}_{\text{student}}) \right]^{-(\nu+d_2)/2}, \quad (41)$$

where  $\boldsymbol{\Sigma}_{\text{student}}$  is the scale matrix of  $\mathbf{x}_2$ . By identification, the following results are obtained:

$$\begin{aligned} \nu &= d_1 \\ \boldsymbol{\mu}_{\text{student}} &= \boldsymbol{\mu}_{2.1} = \boldsymbol{\mu}_2 + \boldsymbol{\Sigma}_{21}\boldsymbol{\Sigma}_{11}^{-1}(\mathbf{x}_1 - \boldsymbol{\mu}_1) \\ \boldsymbol{\Sigma}_{\text{student}}^{-1} &= \frac{d_1}{(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1)} \times \boldsymbol{\Sigma}_{22.1}^{-1} \\ \boldsymbol{\Sigma}_{\text{student}} &= \frac{(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T (\boldsymbol{\Sigma}_{11})^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1)}{d_1} \times \boldsymbol{\Sigma}_{22.1} \end{aligned}$$

## References

- Anderson, T.W., 1957. Maximum likelihood estimates for a multivariate normal distribution when some observations are missing. *J. Am. Stat. Assoc.* 52, 200–203. doi:[10.1080/01621459.1957.10501379](https://doi.org/10.1080/01621459.1957.10501379).
- Bilodeau, M., Brenner, D., 1999. *Theory of multivariate statistics*. Springer, New York. doi:[10.1007/b97615](https://doi.org/10.1007/b97615).
- Bouveyron, C., Brunet-Saumard, C., 2014. Model-based clustering of high-dimensional data: A review. *Comput. Stat. Data Anal.* 71, 52–78. URL: <https://www.sciencedirect.com/science/article/pii/S0167947312004422>, doi:<https://doi.org/10.1016/j.csda.2012.12.008>.
- Bouveyron, C., Girard, S., Schmid, C., 2007. High-dimensional data clustering. *Comput. Stat. Data Anal.* 52, 502–519. URL: <https://www.sciencedirect.com/science/article/pii/S0167947307000692>, doi:<https://doi.org/10.1016/j.csda.2007.02.009>.
- Browne, R.P., McNicholas, P.D., 2015. A mixture of generalized hyperbolic distributions. *Can. J. Stat.* 43, 176–198. doi:<https://doi.org/10.1002/cjs.11246>.
- van Buuren, S., 2018. *Flexible imputation of missing data*. CRC press. doi:<https://doi.org/10.1201/9780429492259>.
- van Buuren, S., Groothuis-Oudshoorn, K., 2011. MICE: Multivariate imputation by chained equations in R. *J. Stat. Softw.* 45, 1–67. URL: <https://www.jstatsoft.org/v045/i03>, doi:[10.18637/jss.v045.i03](https://doi.org/10.18637/jss.v045.i03).
- Campbell, N.A., 1984. Mixture models and atypical values. *Math. Geol.* 16, 465–477. doi:[10.1007/BF01886327](https://doi.org/10.1007/BF01886327).
- Cismondi, F., Fialho, A.S., Vieira, S.M., Reti, S.R., Sousa, J.M., Finkelstein, S.N., 2013. Missing data in medical databases: Impute, delete or classify? *Artif. Intell. Med.* 58, 63–72. doi:<https://doi.org/10.1016/j.artmed.2013.01.003>.
- Conte, E., De Maio, A., Ricci, G., 2002. Covariance matrix estimation for adaptive CFAR detection in compound-Gaussian clutter. *IEEE Trans. Aerosp. Electron. Syst.* 38, 415–426. doi:[10.1109/TAES.2002.1008976](https://doi.org/10.1109/TAES.2002.1008976).



- Delalleau, O., Courville, A., Bengio, Y., 2018. Efficient EM training of Gaussian mixtures with missing data. [arXiv:https://arxiv.org/abs/1209.0521](https://arxiv.org/abs/1209.0521).
- Dempster, A.P., Laird, N.M., Rubin, D.B., 1977. Maximum likelihood from incomplete data via the EM algorithm. *J. R. Stat. Soc.* 39.
- Eirola, E., Lendasse, A., Vandewalle, V., Biernacki, C., 2014. Mixture of Gaussians for distance estimation with missing data. *Neurocomputing* 131, 32–42. doi:<https://doi.org/10.1016/j.neucom.2013.07.050>.
- Farhangfar, A., Kurgan, L.A., Pedrycz, W., 2007. A novel framework for imputation of missing values in databases. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans* 37, 692–709. doi:[10.1109/TSMCA.2007.902631](https://doi.org/10.1109/TSMCA.2007.902631).
- Fraley, C., Raftery, A.E., 2002. Model-based clustering, discriminant analysis, and density estimation. *J. Am. Stat. Assoc.* 97, 611–631. URL: <http://www.jstor.org/stable/3085676>.
- Friedman, J., Hastie, T., Tibshirani, R., 2008. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* 9, 432–441. doi:[10.1093/biostatistics/kxm045](https://doi.org/10.1093/biostatistics/kxm045).
- Ghahramani, Z., Jordan, M., 1994a. Supervised learning from incomplete data via an EM approach, in: *Advances in Neural Information Processing Systems*, Morgan-Kaufmann. pp. 120–127. URL: <https://proceedings.neurips.cc/paper/1993/file/f2201f5191c4e92cc5af043eebfd0946-Paper.pdf>.
- Ghahramani, Z., Jordan, M.I., 1994b. Learning from Incomplete Data. Technical Report. Massachusetts Institute of Technology. URL: <http://mlg.eng.cam.ac.uk/zoubin/papers/review.pdf>.
- Higuera, C., Gardiner, K.J., Cios, K.J., 2015. Self-organizing feature maps identify proteins critical to learning in a mouse model of Down syndrome. *PLOS ONE* 10, 1–28. doi:[10.1371/journal.pone.0129126](https://doi.org/10.1371/journal.pone.0129126).
- Hippert-Ferrer, A., El Korso, M., Breloy, A., Ginolhac, G., 2022. Robust low-rank covariance matrix estimation with a general pattern of missing values. *Signal Processing* 195, 108460. doi:<https://doi.org/10.1016/j.sigpro.2022.108460>.
- Kelker, D., 1970. Distribution theory of spherical distributions and a location-scale parameter generalization. *Sankhya A* 32, 419–430.
- Lin, W.C., Tsai, C.F., 2020. Missing value imputation: a review and analysis of the literature (2006-2017). *Artif. Intell. Rev.* 53, 1487–1509. doi:[10.1007/s10462-019-09709-4](https://doi.org/10.1007/s10462-019-09709-4).
- Little, R.J., Rubin, D.B., 2002. *Statistical analysis with missing data*. 2nd ed., John Wiley & Sons, Inc. Hoboken, NJ, USA.
- Liu, F.T., Ting, K.M., Zhou, Z.H., 2012. Isolation-based anomaly detection. *ACM Trans. Knowl. Discov. Data* 6. doi:[10.1145/2133360.2133363](https://doi.org/10.1145/2133360.2133363).
- Mirza, B., Wang, W., Wang, J., Choi, H., Chung, N.C., Ping, P., 2019. Machine learning and integrative analysis of biomedical big data. *Genes* 10. doi:[10.3390/genes10020087](https://doi.org/10.3390/genes10020087).
- Moran, M., Inoue, Y., Barnes, E., 1997. Opportunities and limitations for image-based remote sensing in precision crop management. *Remote Sens. Environ.* 61, 319–346. doi:[https://doi.org/10.1016/S0034-4257\(97\)00045-X](https://doi.org/10.1016/S0034-4257(97)00045-X).
- Mouret, F., Albughdadi, M., Duthoit, S., Kouamé, D., Rieu, G., Tourneret, J.Y., 2021. Outlier detection at the parcel-level in wheat and rapeseed crops using multispectral and SAR time series. *Remote Sens.* 13, 956. URL: <http://dx.doi.org/10.3390/rs13050956>, doi:[10.3390/rs13050956](https://doi.org/10.3390/rs13050956).
- Mouret, F., Albughdadi, M., Duthoit, S., Kouamé, D., Rieu, G., Tourneret, J.Y., 2022. Reconstruction of Sentinel-2 derived time series using robust Gaussian mixture models. Application to the detection of anomalous crop development. *Comput. Electron. Agric.* 198, 106983. doi:<https://doi.org/10.1016/j.compag.2022.106983>.

- Nash, W.J., Sellers, T.L., Talbot, S.R., Cawthorn, A.J., Ford, W.B., 1994. The population biology of abalone (haliotis species) in Tasmania: blacklip abalone (*H. rubra*) from the north coast and islands of bass strait. Sea Fisheries Division, Technical Report 48, p411.
- Ollila, E., Tyler, D.E., Koivunen, V., Poor, H.V., 2012. Complex elliptically symmetric distributions: Survey, new results and applications. *IEEE Trans. Signal Process.* 60, 5597–5625. doi:[10.1109/TSP.2012.2212433](https://doi.org/10.1109/TSP.2012.2212433).
- Pascal, F., Bombrun, L., Tournet, J.Y., Berthoumieu, Y., 2013. Parameter estimation for multivariate generalized Gaussian distributions. *IEEE Trans. Signal Process.* 61, 5960–5971. doi:[10.1109/TSP.2013.2282909](https://doi.org/10.1109/TSP.2013.2282909).
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* 12, 2825–2830. URL: <http://jmlr.org/papers/v12/pedregosa11a.html>.
- Peel, D., McLachlan, G.J., 2000. Robust mixture modelling using the t distribution. *Stat. Comput.* 25, 339–348. doi:<https://doi.org/10.1023/A:1008981510081>.
- Roizman, V., Jonckheere, M., Pascal, F., 2020. A flexible EM-like clustering algorithm for noisy data. To appear [arXiv:1907.01660](https://arxiv.org/abs/1907.01660).
- Roizman, V., Jonckheere, M., Pascal, F., 2021. Robust clustering and outlier rejection using the Mahalanobis distance distribution, in: *Proc. European Signal Processing Conference (EUSIPCO)*, Amsterdam, NL. pp. 2448–2452. doi:[10.23919/Eusipco47968.2020.9287356](https://doi.org/10.23919/Eusipco47968.2020.9287356).
- Ruan, L., Yuan, M., Zou, H., 2011. Regularized parameter estimation in high-dimensional gaussian mixture models. *Neural. Comput.* 23, 1605–1622. doi:[10.1162/NECO\\_a\\_00128](https://doi.org/10.1162/NECO_a_00128).
- Salberg, A.B., 2011. Land cover classification of cloud-contaminated multitemporal high-resolution images. *IEEE Transactions on Geoscience and Remote Sensing* 49, 377–387. doi:[10.1109/TGRS.2010.2052464](https://doi.org/10.1109/TGRS.2010.2052464).
- Shen, H., Li, X., Cheng, Q., Zeng, C., Yang, G., Li, H., Zhang, L., 2015. Missing information reconstruction of remote sensing data: A technical review. *IEEE Geosci.Remote Sens. Mag.* 3, 61–85. doi:[10.1109/MGRS.2015.2441912](https://doi.org/10.1109/MGRS.2015.2441912).
- Sportisse, A., Biernacki, C., Boyer, C., Josse, J., Lourdelle, M.M., Celeux, G., Laporte, F., 2021. Model-based clustering with missing not at random data. To appear [arXiv:2112.10425](https://arxiv.org/abs/2112.10425).
- Stekhoven, D.J., Buhlmann, P., 2011. MissForest. Nnon-parametric missing value imputation for mixed-type data. *Bioinformatics* 28, 112–118. doi:[10.1093/bioinformatics/btr597](https://doi.org/10.1093/bioinformatics/btr597).
- Tadjudin, S., Landgrebe, D., 2000. Robust parameter estimation for mixture model. *IEEE Trans. Geosci. Remote Sens.* 38, 439–445. doi:[10.1109/36.823939](https://doi.org/10.1109/36.823939).
- Troyanskaya, O., Cantor, M., Sherlock, G., Brown, P., Hastie, T., Tibshirani, R., Botstein, D., Altman, R.B., 2001. Missing value estimation methods for DNA microarrays. *Bioinformatics* 17, 520–525. doi:[10.1093/bioinformatics/17.6.520](https://doi.org/10.1093/bioinformatics/17.6.520).
- Wang, H.X., Zhang, Q.B., Luo, B., Wei, S., 2004. Robust mixture modelling using multivariate t-distribution with missing information. *Pattern Recognition Lett.* 25, 701–710. doi:<https://doi.org/10.1016/j.patrec.2004.01.010>.
- Wei, Y., Tang, Y., McNicholas, P.D., 2019. Mixtures of generalized hyperbolic distributions and mixtures of skew-t distributions for model-based clustering with incomplete data. *Comput. Stat. Data. Anal.* 130, 18–41. doi:<https://doi.org/10.1016/j.csda.2018.08.016>.