Available online at www.sciencedirect.com

ScienceDirect

Fuzzy Sets and Systems 340 (2018) 1–37

FUZZY sets and systems

www.elsevier.com/locate/fss

Clustering of fuzzy data and simultaneous feature selection: A model selection approach

Arkajyoti Saha [a], Swagatam Das [b,*]

[a] *Stat-Math Unit, Indian Statistical Institute, 203 B.T.Road, Kolkata-700108, W.B., India*
[b] *Electronics and Communication Sciences Unit, Indian Statistical Institute, 203 B.T.Road, Kolkata-700108, W.B., India*

Received 1 August 2016; received in revised form 12 July 2017; accepted 28 November 2017
Available online 5 December 2017

Abstract

Fuzzy data occurs frequently in the fields of decision making, social sciences, and control theory. We consider the problem of clustering fuzzy data along with automatic component number detection and feature selection. A model selection criterion called minimum message length is used to address the problem of component number selection. The Bayesian framework can be adopted here, by applying an explicit prior distribution over the parameter values. We discuss both uninformative and informative priors. For the latter, a gradient descent algorithm for automatic optimization of the prior hyper-parameters is presented. The problem of simultaneous feature selection involves ordering the discriminative features according to their relative importance, and at the same time eliminating non-discriminative features. The feature selection problem is also formulated as a parameter estimation problem by extending the concept of feature saliency. Then the estimation can be computed simultaneously with the clustering steps. By combining the clustering, the cluster number detection and the feature selection into one estimation problem, we modified the fuzzy Expectation–Maximization (EM) algorithm to perform all of the estimation. Evaluation criteria are proposed and empirical study results are reported to showcase the efficacy of our proposals.

Keywords: Fuzzy data; Component number selection; Model selection; Minimum Message Length (MML); Bayesian; Jeffreys prior; DNW prior

1. Introduction

Often, the classes of objects encountered in the physical world suffer from non-random imprecision. Especially in the fields of artificial intelligence, pattern recognition, communication of information, decision making, social sciences, industrial engineering (automobile, control engineering, medicine, logistics, and vehicular communications), we frequently encounter cases where a precise value of a variable is either not available, cannot be measured, is economically infeasible, or is not required. It might also be the case that the magnitudes are ill defined or only linguistic values for the magnitudes are available. Fuzzy data helps us to mathematically model the vagueness with

* Corresponding author.
E-mail addresses: arkajyotisaha93@gmail.com (A. Saha), swagatam.das@isical.ac.in (S. Das).

https://doi.org/10.1016/j.fss.2017.11.015
0165-0114/

the least possible loss of information, while retaining the inherent simplicity of the data. The statistical formalization of the idea of imprecision in data can be performed in the following two ways.

1. **Physical interpretation of fuzzy data:** In this interpretation of fuzzy data, the independent existence of fuzzy datum is assumed, i.e. it is not associated with any underlying precise variable [1]. This assumes the data under consideration is intrinsically fuzzy and follows mathematical formalism of fuzzy random variables, which are defined as mappings from a probability space to a fuzzy subset [2] with certain measurability properties. Some recent literature follows this interpretation of fuzzy data for estimation and hypothesis testing purposes [3,4].
2. **Epistemic interpretation of fuzzy data:** In this approach, the fuzzy numbers are assumed to "imperfectly specify a value that is existing and precise, but not measurable with exactitude under the given observation conditions" [1]. Unlike the previous approach, here, a fuzzy datum is associated with an existing and precise random variable. It is interpreted as a "possibility distribution associated to a precise realization of a random variable that has only been partially observed" [5]. We adopt this approach towards the fuzzy data in this article.

Clustering is the method of partitioning the data based on some (dis)similarity measures, such that the data within a cluster is as similar as possible, and data from different clusters are as dissimilar as possible. Under the assumption that the data are produced (following some probability distribution) from one of a number (unknown) of alternative sources of generation, finite mixture models are rich enough to be extended to an involved statistical model by which issues like selection of an optimal number of clusters, feature saliency and the validity of a given model can be addressed in a formal and structured way [6].

Clustering of fuzzy data is a topic of interest to the modern research on imprecise data analytics [7–14]. Though the mixture model has been extensively used in clustering of crisp data it was first introduced in the fuzzy setup in [5,15]. The fuzzy EM algorithm performs the mixture model based clustering of fuzzy data. The major limitations with this algorithm are summarized below:

1. It does not take into consideration the number of components, which is one of the most important things to be determined in the clustering of a dataset and is unknown in general.
2. It does not consider selecting the most useful features, and hence suffers from poor performance in the presence of noise variables and higher computational load.
3. The standard fuzzy EM algorithm, which is used to fit the finite mixture model, suffers from problems [6] associated with the basic EM algorithm, like sensitivity to initialization and convergence to boundary of the parameter set [5].

In [15], the Monte-Carlo estimation was introduced to take care of the non-Gaussian scenario, which is developed as a generalization of [5]. The issue with the convergence to local maximum and dependence on initialization was taken care of with Bayesian approaches. In this article, we extend the Bayesian approaches in the perspective of feature selection. For the sake of computational simplicity, we have restricted our attention to only Gaussian mixture models. Similar to [15], the developed algorithm can also be extended in the non-Gaussian scenario with Monte-Carlo estimation.

In this paper, we introduce an automated model selection approach for the clustering of fuzzy data. We summarize the main contributions of the paper as follows:

1. In order to automatically determine the number of components, we incorporate a classical Bayesian model-selection technique, namely the Minimum Message Length (MML) criterion [16], along with different variations and choices of priors, in the conventional Fuzzy EM setup, which solves the problem of high sensitivity to initialization and convergence to the boundary of parameter set.
2. We formulate and incorporate the feature selection problem as an estimation problem by extending the concept of feature saliency [17] in the fuzzy data setup and develop a modified fuzzy EM algorithm to address the problem of feature saliency determination. To avoid the dependency on "good initialization", here we introduce the MML criterion as the model selection criterion with two different choices of priors (uninformative and informative). In the corresponding analysis, this guarantees that feature saliency of irrelevant or noisy features are driven towards zero, resulting in automatic feature selection.

For the sake of computational complexity we have kept ourselves restricted to the most commonly used finite Gaussian Mixture Model (GMM), however, the whole setup can be equivalently extended for finite mixture models with any distribution other than Gaussian.

2. Preliminaries

A word about the notation: bold faced letters, e.g., $\mathbf{x}, \mathbf{y}$ are used to represent vectors. Sets are represented by calligraphic upper-case alphabets, e.g., $\mathcal{X}, \mathcal{Y}$. Matrices are denoted by upper-case bold faced vectors, e.g., $\mathbf{X}, \mathbf{Y}$. Random variables are represented by upper-case alphabets, e.g., X, Y. The symbols $\mathbb{R}, \mathbb{N}$ and $\mathbb{R}^d$ denote the set of real numbers, the set of natural numbers, and the d-dimensional real vector space, respectively.

2.1. Mathematical formulation of fuzzy data

The concept of the epistemic interpretation of fuzzy data can be mathematically formulated in the following way. This approach assumes the existence of a precise random variable X (the complete data vector) taking values in a sample space $\mathcal{S}$ and describing the realizations of a random experiment. Here, we assume that instead of crisp values, only partial information about a realization $\mathbf{x}$ is available to us in the form of a fuzzy subset $\tilde{\mathbf{x}}$ of $\mathcal{S}$. The Borel measurable membership function associated with $\tilde{\mathbf{x}}$ is denoted by $\mu_{\tilde{\mathbf{x}}} : [0, 1]$. This fuzzy data is a representative of the observer's partial knowledge about the actual crisp realization of X. This particular fuzzy set $\tilde{\mathbf{x}}$ has a two-step generation procedure, resulting in two types of uncertainty [18] in the final presentation.

1. Step 1: The random experiment step, through which the crisp realization $\mathbf{x}$ is drawn from X. The aleatoric uncertainty related to this step is due to the random nature of the data generation and hence cannot be checked before the experiment is performed. It disappears once the experiment is over.
2. Step 2: Here, the observer encodes the gathered partial knowledge of $\mathbf{x}$ in terms of a possibility distribution $\mu_{\tilde{\mathbf{x}}}$. The uncertainty related to this step is epistemic and can be regulated by gathering additional information about the actual crisp realization.

Detailed discussion on this approach of handling fuzzy data along with real life examples of them are also available in the literature [5].

2.2. Clustering setup

We describe the clustering setup under consideration while developing the GMM based clustering procedure for fuzzy data.

1. Let $X = [X_1, X_2, \ldots, X_d]^T$ be a d-dimensional random variable, which follows a finite mixture distribution with g Gaussian components.
2. Let $\boldsymbol{\pi} = [\pi_1, \pi_2, \cdots, \pi_g]$ denote the mixing probabilities.
3. kth component is $N(\boldsymbol{m}_k, \boldsymbol{\Sigma}_k)$ and features are conditionally independent given the component label. In this particular case of GMM, this particular assumption is equivalent to adopting a diagonal covariance matrix i.e. $\boldsymbol{\Sigma}_k = \text{diag}(\sigma_{k1}^2, \sigma_{k2}^2, \cdots, \sigma_{kd}^2)$, which is a common choice for handling high-dimensional data, such as latent class models [19], hidden Markov models [20] etc.
4. The missing component label corresponding to the n data points are denoted by $\mathcal{Z} = \{Z_1, Z_2, \cdots, Z_n\}$; $Z_i = [Z_{i1}, Z_{i2}, \cdots, Z_{ig}]$, $Z_{ih} = 1$, $Z_{ik} = 0, \forall k \neq h$; which indicates $\mathbf{x}_i$ belongs to the hth component.
5. Let the underlying crisp dataset be given by a set of n independent and identically distributed (i.i.d.) samples, $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$. The available data are imprecise and represented using fuzzy numbers instead of a crisp value. Let the set of the observed incomplete data be denoted by $\tilde{\mathcal{X}} = \{\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \cdots, \tilde{\mathbf{x}}_n\}$, where the membership function corresponding to $\mathbf{x}_i$ is expressed by $\mu_{\tilde{\mathbf{x}}_i}$. In this article we assume,

$$\mu_{\tilde{\mathbf{x}}_i}(\mathbf{x}) = \prod_{l=1}^{d} \mu_{\tilde{x}_{il}}(x_l).$$

2.3. GMM based clustering and fuzzy EM algorithm

Here we consider $\mathcal{X}, \mathcal{Z}$ to be the complete data and $\tilde{\mathcal{X}}$ to be the observed or incomplete data. Our target is to estimate the unknown model parameters $(\mathcal{A})$.

2.3.1. Complete data log-likelihood

It is given by,

$$L(\mathcal{X}, \mathcal{Z} \mid \mathcal{A}) = \log[\mathbb{P}(\mathcal{X}, \mathcal{Z} \mid \mathcal{A})] = \sum_{k=1}^{g} \log \pi_k \sum_{i=1}^{n} Z_{ik} - \frac{nd}{2} \log 2\pi$$
$$-\frac{1}{2} \sum_{k=1}^{g} \sum_{i=1}^{n} Z_{ik} \sum_{l=1}^{d} \log \sigma_{kl} \sum_{k=1}^{g} \sum_{i=1}^{n} \sum_{l=1}^{d} \frac{Z_{ik}}{\sigma_{kl}^2} (x_{il} - m_{kl})^2.$$

2.3.2. The E-step

Let the current fit of $\mathcal{A}$ at the tth iteration be denoted by $\mathcal{A}^{(t)}$. The E-step consists of calculating the following quantity:

$$Q(\mathcal{A}, \mathcal{A}^{(t)}) = \mathbb{E}_{\mathcal{A}^{(t)}}(L(\mathcal{X}, \mathcal{Z} \mid \mathcal{A}) \mid \tilde{\mathcal{X}}) = \sum_{k=1}^{g} \log \pi_k \sum_{i=1}^{n} u_{ik}^{(t)} - \frac{nd}{2} \log 2\pi$$
$$-\sum_{k=1}^{g} \sum_{i=1}^{n} u_{ik}^{(t)} \sum_{l=1}^{d} \log \sigma_{kl} - \frac{1}{2} \sum_{k=1}^{g} \sum_{i=1}^{n} u_{ik}^{(t)} \sum_{l=1}^{d} \left(\frac{{\gamma_{ikl}^{(t)}}^2}{\sigma_{kl}^{(t)}} - 2\frac{m_{kl} \delta_{ikl}^{(t)}}{\sigma_{kl}^{(t)}} + \frac{m_{kl}^2}{\sigma_{kl}^{(t)}} \right)^2$$

where,

$$\alpha_{ikl}^{(t)} = \mathbb{P}_{\mathcal{A}^{(t)}}(\tilde{\mathbf{x}}_{il} \mid Z_{ik} = 1) = \int \mu_{\tilde{\mathbf{x}}_{il}}(x_{il}) p(x_{il} \mid \{m_{kl}^{(t)}, {\sigma_{kl}^{(t)}}^2\}) dx_{il},$$

$$\alpha_{ik}^{(t)} = \mathbb{P}_{\mathcal{A}^{(t)}}(\tilde{\mathbf{x}}_i \mid Z_{ik} = 1) = \prod_{l=1}^{d} \alpha_{ikl}^{(t)},$$

$$\beta_i^{(t)} = \mathbb{P}_{\mathcal{A}^{(t)}}(\tilde{\mathbf{x}}_i) = \sum_{k=1}^{g} \pi_k^{(t)} \alpha_{ik}^{(t)},$$

$$u_{ik}^{(t)} = \mathbb{E}_{\mathcal{A}^{(t)}}(Z_{ik} \mid \tilde{\mathcal{X}}) = \frac{\mathbb{P}_{\mathcal{A}^{(t)}}(\tilde{\mathbf{x}}_i \mid Z_{ik} = 1) \mathbb{P}_{\mathcal{A}^{(t)}}(Z_{ik} = 1)}{\mathbb{P}_{\mathcal{A}^{(t)}}(\tilde{\mathbf{x}}_i)} = \frac{\alpha_{ik}^{(t)} \pi_k^{(t)}}{\beta_i^{(t)}},$$

$$\gamma_{ikl}^{(t)} = \mathbb{E}_{\mathcal{A}^{(t)}}(x_{il}^2 \mid \tilde{\mathcal{X}}, Z_{ik} = 1) = \frac{\int x_{il}^2 \mu_{\tilde{\mathbf{x}}_{il}}(x_{il}) p(x_{il} \mid \{m_{kl}^{(t)}, {\sigma_{kl}^{(t)}}^2\}) dx_{il}}{\alpha_{ikl}^{(t)}},$$

$$\delta_{ikl}^{(t)} = \mathbb{E}_{\mathcal{A}^{(t)}}(x_{il}^2 \mid \tilde{\mathcal{X}}, Z_{ik} = 1) = \frac{\int x_{il} \mu_{\tilde{\mathbf{x}}_{il}}(x_{il}) p(x_{il} \mid \{m_{kl}^{(t)}, {\sigma_{kl}^{(t)}}^2\}) dx_{il}}{\alpha_{ikl}^{(t)}},$$

$$\mathbb{E}_{\mathcal{A}^{(t)}}(Z_{ik} x_{il}^2 \mid \tilde{\mathcal{X}}) = \mathbb{E}_{\mathcal{A}^{(t)}}(x_{il}^2 \mid \tilde{\mathcal{X}}, Z_{ik} = 1) \mathbb{P}_{\mathcal{A}^{(t)}}(Z_{ik} = 1 \mid \tilde{\mathcal{X}}) = \gamma_{ikl}^{(t)} u_{ik}^{(t)},$$
$$\mathbb{E}_{\mathcal{A}^{(t)}}(Z_{ik} x_{il} \mid \tilde{\mathcal{X}}) = \mathbb{E}_{\mathcal{A}^{(t)}}(x_{il} \mid \tilde{\mathcal{X}}, Z_{ik} = 1) \mathbb{P}_{\mathcal{A}^{(t)}}(Z_{ik} = 1 \mid \tilde{\mathcal{X}}) = \delta_{ikl}^{(t)} u_{ik}^{(t)}.$$

2.3.3. The M-step

In this step, the partial derivative of $Q(\mathcal{A}, \mathcal{A}^{(t)})$ with respect to the model parameters are set to zero, to obtain an upgradation rule for each of them in each iteration. The derived upgradation rules for the model parameters are as follows:

$$\pi_k^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} u_{ik}^{(t)}, \; m_{kl}^{(t+1)} = \frac{\sum_{i=1}^{n} u_{ik}^{(t)} \delta_{ikl}^{(t)}}{\sum_{i=1}^{n} u_{ik}^{(t)}},$$

$$\sigma_{kl}^{(t+1)} = \sqrt{\frac{\sum_{i=1}^{n} u_{ik}^{(t)} \left(\gamma_{ikl}^{(t)} + 2m_{kl}^{(t+1)} \delta_{ikl}^{(t+1)} + m_{kl}^{(t+1)^2}\right)}{\sum_{i=1}^{n} u_{ik}^{(t)}}}.$$

3. Automatic component number selection in fuzzy data with MML criterion

From an information theoretic point of view, the MML approach is a restatement of Ocham's razor; even when candidate models are not equal in goodness of fit accuracy to the data under consideration, the one generating the most compressed overall message containing the data, is more likely to be correct.

In conventional deterministic model selection, any component mixture model with one or more zero mixing probabilities is effectively identical with mixture models with a smaller number of components. Hence, instead of finding a class of candidate models and then using MML just as a model selection criteria [21], allowing the mixing probabilities to be zero, we can integrate the MML criterion from the very first stage to find the overall best model in the entire set of the available models [6].

3.1. Mathematical formulation of the MML criterion

The approximated message length for a mixture of distributions for dataset $\tilde{\mathcal{X}}$ is given by the following rule [6]:

$$M(\tilde{\mathcal{X}}, \mathcal{A}) \approx -\log(p(\mathcal{A})) - L(\tilde{\mathcal{X}} \mid \mathcal{A}) + \frac{1}{2}\log|\mathbf{I}(\mathcal{A})| + \frac{N_p}{2}(1 + \log(\kappa_{N_p})), \tag{3}$$

where $p(\mathcal{A})$ is the prior probability. $L(\tilde{\mathcal{X}} \mid \mathcal{A})$ is the likelihood of the data given the parameters; $\mathbf{I}(\mathcal{A})$ is the expected Fisher information matrix given by $-\mathbb{E}\left[D_{\mathcal{A}}^2 L(\tilde{\mathcal{X}} \mid \mathcal{A})\right]$. N_p is the number of parameters to be estimated and in our case, $N_p = (2d+1)g$. κ_{N_p} is the optimal quantization lattice constant for $\mathbb{R}^{N_p}$. We have $\kappa_1 = \frac{1}{12}$, as N_p grows, κ_{N_p} asymptotically tends to the value $\frac{1}{2\pi e}$. Thus it is enough to approximate it by $\frac{1}{12}$ [6]. The optimal number of clusters is obtained by finding the minimum of the message length with respect to the set of model parameters i.e. $\mathcal{A}$.

This MML criterion is closely related with other model selection criteria. The famous MDL criterion for model selection can be obtained from this condition by simply assuming a flat prior and then dropping it followed by the cancellation of order-1 terms [6]. We also observe that apart from the lattice constant, the derived MML criterion coincides with the conceptually different Laplace-empirical criterion [22].

3.2. Approximation of the fisher information matrix

As $\mathbf{I}(\mathcal{A})$ in general cannot be obtained by analytical methods, in a crisp data scenario, we replace the expected Fisher information matrix by the block-diagonal complete data Fisher information matrix $\mathbf{I}_c(\mathcal{A})$, which provides an upper-bound to the expected Fisher information matrix [6]. Here using the fact that in GMM, the complete data corresponding to both crisp and fuzzy incomplete data are same, we adopt this technique in the fuzzy data scenario. Thus, we have,

$$\mathbf{I}_c(\mathcal{A}) = \text{block-diag}\{n\pi_1 \mathbf{I}^{(1)}(\mathcal{T}_1), n\pi_2 \mathbf{I}^{(1)}(\mathcal{T}_2), \cdots, n\pi_g \mathbf{I}^{(1)}(\mathcal{T}_g), \mathbf{I}_n(\boldsymbol{\pi})\},$$

where $\mathbf{I}^{(1)}(\mathcal{T}_k)$ is the Fisher information matrix for a single observation known to have been produced from the kth component. $\mathbf{I}_n(\boldsymbol{\pi})$ denotes the Fisher information matrix of the multinomial distribution with n observations. Under this approximation we have the following famous and thoroughly studied approximation of $\mathbf{I}(\mathcal{A})$ [6,17]:

$$|\mathbf{I}_c(\mathcal{A})| = |\mathbf{I}_n(\boldsymbol{\pi})| \prod_{k=1}^{g} |n\pi_k \mathbf{I}^{(1)}(\mathcal{T}_k)| = \prod_{k=1}^{g} n\pi_k^{-1} \prod_{l=1}^{d} \frac{2(n\pi_k)^2}{\sigma_{kl}^4}.$$

3.3. Choice of prior distributions

The choice of prior and its effect on MML criterion and the actual clustering in the perspective of crisp data is a thoroughly studied problem in the literature [23]. In the following subsections, we introduce some of the most

commonly used priors for the component number selection in fuzzy data. We derive the modified form of the MML criterion and derive an EM algorithm with automated model selection for each of the choices of prior.

In the case of crisp data, the Deterministic Annealing (DA) algorithm has been used successfully for avoiding the initialization problem in EM based clustering algorithms with the help of an uninformative initialization with high entropy. If the EM algorithm is accompanied with a similar kind of uninformative initialization (high entropy), it also exhibits a similar kind of self-annealing behavior. The success of the EM algorithm with random starting [22] can be explained with the help of a similar concept. We adopt this concept of uninformative high entropy initialization for fuzzy data by using:

1. uninformative priors;
2. informative priors with automated upgrade.

3.3.1. Uninformative priors

An uninformative prior is used to quantify vague or general information about the unknown model parameters. These objective priors can express "objective" information such as the range of model parameters but cannot express subjectively elicited information. Adopting an uninformative prior ensures that we are not assuming any kind of special preference for any of the components, which in turn guarantees a high entropy uninformative initialization.

3.3.2. Informative priors with automated upgrade

An informative prior is used to quantify specific information regarding the prior belief corresponding to the model parameters. If the prior belief is close to the true value, informative priors can improve the prediction results. On the other hand, if no definite information is available a priori, or the true value varies significantly form the prior belief, we lose the advantage of random starting with high entropy. Hence, we develop an additional step for optimization of the MML criterion with respect to the set of hyper-parameters, so that information gathered about the unknown model parameters in each iteration is used in the next iteration. We ensure a high entropy random start with the help of proper initialization of the hyper-parameters of the prior distribution.

3.4. Development of the MML criterion and an optimization algorithm with uninformative priors

In this section we concentrate on some uninformative priors and study the MML criterion and the corresponding E-step and M-step of the fuzzy EM algorithm.

3.4.1. Jeffreys prior

As we have no prior knowledge of the model parameters, we assume them to be prior independent and independent from the mixing probabilities. Hence the prior on the mixture parameters should be of the following form:

$$p(\mathcal{A}) = p_1(\pi_1, \pi_2, \cdots, \pi_g) \prod_{k=1}^{g} p_2(\mathcal{T}_k).$$

For each part of the aforementioned priors, we choose the standard non-informative Jeffreys prior. The Jeffreys prior is given by the following formula:

$$p_1(\pi_1, \pi_2, \cdots, \pi_g) \propto \sqrt{|\mathbf{I}(\boldsymbol{\pi})|} = 1/\sqrt{\prod_{k=1}^{g} \pi_k},$$

$$p_2(\mathcal{T}_k) \propto \sqrt{|\mathbf{I}^{(1)}(\mathcal{T}_k)|}, \forall k = 1, 2, \cdots, g.$$

Incorporation of the aforementioned priors in the MML criterion of Section 3.1 leads us to the following:

$$M(\tilde{\mathcal{X}}, \mathcal{A}) \approx d \sum_{k=1}^{g} \log(\frac{n\pi_k}{12}) + \frac{g}{2} \log(\frac{n}{12}) + \frac{3g}{2} - L(\tilde{\mathcal{X}} \mid \mathcal{A}).$$

This formulation of model selection is at par with the two-part code formulation of MML [21] and MDL criteria [24]. The problem with the aforementioned criterion is that we allow mixing probabilities to attain the value zero, but in that

case, this leads to a meaningless optimization task, hence we bypass this difficulty by coding only the components with non-zero mixing probabilities. For encoding the number of components with positive probability(g_p), an additional term is added to the aforementioned MML criterion, but we neglect the term due to the fact that this code length is constant and hence does not affect the optimization results. Invoking all these assumptions, we have the following optimization task in hand:

$$\tilde{\mathcal{A}} = \text{argmin}_A \left\{ d \sum_{k:\pi_k>0} \log(\frac{n\pi_k}{12}) + \frac{g_P}{2} \log(\frac{n}{12}) + \frac{3g_P}{2} - L(\tilde{\mathcal{X}} \mid \mathcal{A}) \right\}.$$

In the aforementioned optimization task, for fixed number of components with positive probability we can neglect the constant terms and the resulting optimization task can be approached from the Bayesian point of view as the MAP (Maximum A Posteriori Probability) estimation procedure resulting from the adoption of flat or no prior on the parameters specifying the model components and the following prior on the non-zero mixing probabilities:

$$p(\pi_1, \pi_2, \cdots, \pi_g) \propto \exp\left[-d \sum_{k:\pi_k>0} \log \pi_k \right].$$

The aforementioned-prior can be interpreted as a Dirichlet prior with negative exponent $-d$, hence improper. This differs from the original version of Jeffreys prior on mixing probabilities. We also observe that this prior is a conjugate prior to the multinomial likelihoods. Hence the upgradation rule for the mixing probabilities in the conventional fuzzy EM algorithm is modified as follows:

$$\pi_k^{(t+1)} = \left[\max\{0, (\sum_{i=1}^{n} u_{ik}^{(t)}) - d\} \right] / \left[\sum_{k=1}^{g} \max\{0, (\sum_{i=1}^{n} u_{ik}^{(t)}) - d\} \right].$$

Due to the use of a flat prior on $\mathcal{T}_k$, the upgradation rule for $\mathcal{T}_k$ is exactly same as that corresponding to the conventional fuzzy EM algorithm. It is worth noticing that the upgradation rule obtained here, performs automatic component annihilation. When one component becomes too weak, it is annihilated in the process. Thus, unlike the standard EM algorithm, this algorithm does not approach the boundary of the parameter space.

3.4.2. The uniform prior

Uniform prior has been used for crisp data in the literature [21]. We adopt this prior in the fuzzy data setup. We assume uniform prior for the mixing probabilities, given by,

$$p_1(\pi_1, \pi_2, \cdots, \pi_g) = (g-1)!,$$

Let some estimation of the population standard deviation and population mean of the underlying crisp observations be denoted by $\boldsymbol{\sigma}^{pop} = \left[\sigma_1^{pop}, \sigma_2^{pop}, \cdots, \sigma_g^{pop}\right]$ and $\boldsymbol{\mu}^{pop} = \left[\mu_1^{pop}, \mu_2^{pop}, \cdots, \mu_g^{pop}\right]$, respectively. Then for each dimension, i.e. $l = 1, 2, \cdots, d$; the uniform prior on $\sigma_{kl}, k = 1, 2, \cdots, g$; is given by a uniform distribution on 0 to σ_l^{pop}. Similarly for each dimension, i.e. $l = 1, 2, \cdots, d$; the uniform prior on $m_{kl}, k = 1, 2, \cdots, g$; is given by a uniform distribution on $(\mu_l^{pop} - \sigma_l^{pop}, \mu_l^{pop} + \sigma_l^{pop})$. Hence, the complete prior on component parameters is given by,

$$q_2(\mathcal{T}_k) = \frac{1}{2} \left[\prod_{l=1}^{d} \sigma_l^{pop} \right]^{-2}, k = 1, 2, \cdots, g.$$

We also observe that the order in which the parameters are stated, is irrelevant and hence we can save a message length of $-\log(g!)$. Incorporating this prior, the MML criterion in Section 3.1 is transformed into the following one:

$$M_U(\tilde{\mathcal{X}}, \mathcal{A}) \approx 2g \sum_{l=1}^{d} \log(\sqrt{2}\sigma_l^{pop}) - \log(g-1)! - L(\tilde{\mathcal{X}} \mid \mathcal{A}) + \frac{1}{2} g \log n$$
$$- \frac{1}{2} \sum_{k=1}^{g} \log(\pi_k) + \sum_{k=1}^{g} \sum_{l=1}^{d} \log(\frac{\sqrt{2}n\pi_k}{\sigma_{kl}^2}) + \frac{(2d+1)g_P}{2}(1 - \log 12) - \log(g!).$$

In the case of fuzzy data, we do not have a population of the d dimensional crisp data. Hence, in their absence, as an estimator of $\boldsymbol{\mu}^{pop}$ and $\boldsymbol{\sigma}^{pop}$, we use the mean of the expectation and the variance of the crisp data (given the fuzzy data and the membership function) respectively in the following way:

$$\mu_l^{pop} = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}(x_{il} \mid \tilde{\mathbf{x}}_{il}),$$

$$\sigma_l^{pop} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{Var}(x_{il} \mid \tilde{\mathbf{x}}_{il}) = \frac{1}{n}\sum_{i=1}^{n} (\mathbb{E}(x_{il}^2 \mid \tilde{\mathbf{x}}_{il}) - (\mathbb{E}(x_{il}^2 \mid \tilde{\mathbf{x}}_{il}))^2),$$

where

$$\mathbb{E}(x_{il} \mid \tilde{\mathbf{x}}_{il}) = \frac{\int x_l \mu_{\tilde{\mathbf{x}}_{il}}(x_l) q(x_l) dx_l}{\int \mu_{\tilde{\mathbf{x}}_{il}}(x_l) q(x_l) dx_l},$$

$$\mathbb{E}(x_{il}^2 \mid \tilde{\mathbf{x}}_{il}) = \frac{\int x_l^2 \mu_{\tilde{\mathbf{x}}_{il}}(x_l) q(x_l) dx_l}{\int \mu_{\tilde{\mathbf{x}}_{il}}(x_l) q(x_l) dx_l}.$$

Here, unlike the previous case, we follow the conventional approach for optimization, i.e. we first select a bunch of candidate models by using the conventional fuzzy EM algorithm (by varying the number of components in a suitable range) and then evaluate the corresponding message length for them. We choose the one with minimal value for the message length. The choice of this prior can be justified in the following way:

1. The prior distribution is scale and location invariant, hence the analysis is not affected by any kind of linear change in measurement unit.
2. The uniform prior on mixing probabilities signifies ignorance, whereas the prior on the component parameters suggests that we have imprecise knowledge of the range of the data and the underlying crisp realizations.

3.5. Development of the MML criterion and optimization algorithm based on informative priors with automated upgradation

In this section, we concentrate on some informative priors and study the MML criterion and the corresponding E-step, M-step of the fuzzy EM algorithm along with the additional hyper-parameter upgradation step.

3.5.1. Dirichlet–Normal–Wishart (DNW) prior

This hyper-parameter based prior is the product of a Dirichlet–Prior on the mixing probabilities and a joint Normal–Wishart prior on the component parameters (for notational simplicity, we assume that all the mixing probabilities are non-zero, in the presence of any mixing probability equal to zero, we simply drop the component as done in an earlier case). It is given by,

$$p(\mathcal{A}) = D(\boldsymbol{\pi} \mid \boldsymbol{\lambda}, \epsilon) \prod_{k=1}^{g}\prod_{l=1}^{d} q(m_{kl} \mid \{\mu_{kl}, \frac{\sigma_{kl}^2}{c}\}) W(\frac{1}{\sigma_{kl}^2} \mid \{\nu_{kl}^2, \tau\}),$$

where

$$D(\boldsymbol{\pi} \mid \boldsymbol{\lambda}, \epsilon) = \frac{\Gamma(\epsilon)}{\prod_{k=1}^{g}\Gamma(\epsilon\lambda_k)} \prod_{k=1}^{g} \pi_k^{\epsilon\lambda_k - 1},$$

$$\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \cdots, \lambda_g], \sum_{k=1}^{g}\lambda_k = 1, \lambda_k > 0, \forall k = 1, 2, \cdots, g; \epsilon > 0,$$

$$q(m_{kl} \mid \{\mu_{kl}, \frac{\sigma_{kl}^2}{c}\}) = \frac{\sqrt{c}}{\sigma_{kl}\sqrt{2\pi}} \exp(-\frac{c}{2}\frac{(m_{kl} - \mu_{kl})^2}{\sigma_{kl}^2}), c > 0,$$

$$W(\frac{1}{\sigma_{kl}^2} \mid \{\nu_{kl}^2, \tau\}) = \exp(\frac{1}{2\sigma_{kl}^2}\nu_{kl}^2)\frac{\nu_{kl}^{\tau}\sigma_{kl}^{2-\tau}}{\Gamma(\frac{\tau}{2})2^{\frac{\tau}{2}}}, \tau > 0, \nu_{kl} > 0.$$

Substituting these in the MML criterion of Section 3.1, we have the following:

$$M(\tilde{\mathcal{X}}, \mathcal{A}) \approx -\log \frac{\Gamma(\epsilon)}{\prod_{k=1}^{g} \Gamma(\epsilon\lambda_k)} + \sum_{k=1}^{g} (\frac{2d+1}{2} - \epsilon\lambda_k) \log \pi_k - gd \log \frac{\sqrt{c}}{\sqrt{2\pi}}$$
$$- \sum_{k=1}^{g} \sum_{l=1}^{d} \left[-\frac{c}{2} \frac{(m_{kl} - \mu_{kl})^2}{\sigma_{kl}^2} - \log \sigma_{kl} - \frac{1}{2\sigma_{kl}^2} \nu_{kl}^2 - \tau \log \sigma_{kl} + \tau \log \nu_{kl} \right]$$
$$- L(\tilde{\mathcal{X}} \mid \mathcal{A}) + (\frac{2dg+g}{2}) \log n + dg \log 2\Gamma(\frac{\tau}{2}) 2^{\frac{\tau}{2}} + \frac{N_P}{2}(1 + \log \kappa_{N_P}).$$

In the aforementioned optimization task, for fixed number of components with positive probability we can neglect the constant terms (in a particular iteration the hyper-parameters are assumed to be constant) and the resulting optimization task can be approached from the Bayesian point of view as the MAP estimation procedure resulting from adoption of the following priors:

1. Dirichlet type prior on mixing probabilities:

$$p(\pi_1, \pi_2, \cdots, \pi_g) \propto \exp \left[\sum_{k=1}^{g} (\frac{2d+1}{2} - \epsilon\lambda_k) \log \pi_k \right].$$

2. Joint Normal–Wishart prior on component parameters:

$$\prod_{k=1}^{g} \prod_{l=1}^{d} q(m_{kl} \mid \{\mu_{kl}, \frac{\sigma_{kl}^2}{c}\}) W(\frac{1}{\sigma_{kl}^2} \mid \{\nu_{kl}^2, \tau + 2\}).$$

We also observe that these priors are conjugate priors. Hence the upgradation rule for the mixing probabilities and the model parameters in the conventional fuzzy EM algorithm are modified as follows:

$$\pi_k^{(t+1)} = \frac{\max\{0, (\sum_{i=1}^{n} u_{ik}^{(t)}) - (\frac{2d+1}{2} - \epsilon^{(t)}\lambda^{(t)})\}}{\sum_{k=1}^{g} \max\{0, (\sum_{i=1}^{n} u_{ik}^{(t)}) - (\frac{2d+1}{2} - \epsilon^{(t)}\lambda^{(t)})\}},$$
$$m_{kl}^{(t+1)} = (\sum_{i=1}^{n} u_{ik}^{(t)} \delta_{ikl}^{(t)} + c^{(t)} \mu_{kl}^{(t)}) / (\sum_{i=1}^{n} u_{ik}^{(t)} + c^{(t)}),$$
$$\sigma_{kl}^{(t+1)} = \sqrt{\left[\sum_{i=1}^{n} u_{ik}^{(t)} (\gamma_{ikl}^{(t)} - 2m_{kl}^{(t+1)} \delta_{ikl}^{(t)} + (m_{kl}^{(t+1)})^2) + c^{(t)} (m_{kl}^{(t+1)} - \mu_{kl}^{(t+1)})^2 + (\nu_{kl}^{(t)})^2 \right] / \left[\sum_{i=1}^{n} u_{ik}^{(t)} + \tau^{(t)} + 1 \right]}.$$

Now, for maximum utilization of the data gathered about an unknown model parameter in each iteration, we also automatically upgrade the hyper-parameters by optimizing the MML criterion with respect to the hyper parameters,

$$\mu_{kl}^{(t+1)} = m_{kl}^{(t)};\ c^{(t+1)} = \frac{gd}{\sum_{k=1}^{d} \sum_{l=1}^{d} \frac{m_{kl}^{(t)} - \mu_{kl}^{(t)}{}^2}{\sigma_{kl}^{(t)}}};\ \nu_{kl}^{(t+1)} = \sqrt{\tau^{(t)}} \sigma_{kl}^{(t)}.$$

Now, we observe that for optimization with respect to $\lambda_k, \forall k = 1, 2, \cdots, g;\ \epsilon$ and τ, we do not get a closed form of the optimizer. We take a closer look at each of the optimization tasks to guarantee the existence of a unique optimizer, in each of the aforementioned cases and develop a gradient descent based algorithm [25] to obtain an approximation of the optimizer with the user-specified precision.

Optimization with respect to the λ_k's. For optimization of the message length with respect to $\lambda_k, \forall k = 1, 2, \cdots, g$; the optimization problem at hand is:

$$\mathbf{P}_1 :\ \text{minimize}\ F_1(\boldsymbol{\lambda}) = \log(\prod_{k=1}^{g} \Gamma(\epsilon\lambda_k)) - \sum_{k=1}^{g} \epsilon\lambda_k \log \pi_k, \sum_{k=1}^{g} \lambda_k = 1, \lambda_k > 0.$$

For the sake of computational convenience, we consider the following relaxed optimization task (if the optimizer turns out to be zero, we replace it by a very small quantity like 10^{-7}, for the sake of well-definedness; this approximation hardly matters for practical purposes as the function for arbitrary positive real numbers can only be approximated till finite precision):

$$\mathbf{EP}_1: \text{ minimize } F_1(\boldsymbol{\lambda}) = \log(\prod_{k=1}^{g}\Gamma(\epsilon\lambda_k)) - \sum_{k=1}^{g}\epsilon\lambda_k\log\pi_k, \sum_{k=1}^{g}\lambda_k = 1, \lambda_k \geqslant 0.$$

Now, we observe that the feasible set is convex and the function to be optimized is also strongly convex (Appendix A). Hence, there exists at most one solution. Now, $x \to \log x$ and $x \to \Gamma(x)$ are both continuous functions on the positive real line. Hence, $F_1(\boldsymbol{\lambda})$ is continuous. Here, the feasible set being a $g-1$-simplex is a compact set. Hence, $F_1(\boldsymbol{\lambda})$ has at least one minimizer in the feasible set, which in turn guarantees the existence of an unique minimizer.

To handle the equality constraints in P_1, we introduce the Lagrangian multiplier a and form the Lagrangian. Due to the strong convexity of $F_1(\boldsymbol{\lambda})$, solving $\mathbf{P}_1$ is equivalent to solving the following problem:

$$\mathbf{RP}_1: \text{ minimize } L_{F_1}(\boldsymbol{\lambda}, a) = F_1(\boldsymbol{\lambda}) + a(\sum_{k=1}^{g}\pi_k - 1); \lambda_k \geqslant 0.$$

The upgradation rule for $\lambda_k, \forall k = 1, 2, \cdots, g$; can be expressed in the following way:

$$\begin{aligned}
&\lambda_k^{(t+1)} = (z_k^{(t+1)})^2,\ (z_k^{(0)}, a^{(0)}) = (\varpi_k^{(0)}, b^{(0)}),\\
&z_k^{(t+1)} = \varpi_k^{(t)} - f_1\{2\Psi(\epsilon(\varpi_k^{(t)})^2)\epsilon\varpi_k^{(t)} - 2\epsilon\varpi_k^{(t)}\log\pi_k + b^{(t)}\},\\
&a^{(t+1)} = b^{(t)} - f_1(\sum_{k=1}^{g}(\varpi_k^{(t)})^2 - 1),\\
&(\varpi_k^{(t+1)}, b^{(t+1)}) = (z_k^{(t+1)}, a^{(t+1)}) + \frac{t}{t+3}((z_k^{(t+1)}, a^{(t+1)}) - (z_k^{(t)}, a^{(t)})).
\end{aligned}$$

Optimization with respect to the ϵ. For optimization of the message length with respect to ϵ, the optimization problem at hand is:

$$\mathbf{P}_2: \text{ minimize } F_2(\epsilon) = -\log\Gamma(\epsilon) + \log(\prod_{k=1}^{g}\Gamma(\epsilon\lambda_k)) - \sum_{k=1}^{g}\epsilon\lambda_k\log\pi_k, \epsilon > 0.$$

By the argument used in earlier case, we equivalently consider the following optimization task:

$$\mathbf{EP}_2: \text{ minimize } F_2(\epsilon) = -\log\Gamma(\epsilon) + \log(\prod_{k=1}^{g}\Gamma(\epsilon\lambda_k)) - \sum_{k=1}^{g}\epsilon\lambda_k\log\pi_k, \epsilon \geqslant 0.$$

Now, we observe that the feasible set is convex and the function to be optimized, is also strongly convex (Appendix B). Hence, there exists at most one solution. $F_2(\epsilon)$ is a continuous function on a compact feasible set (Appendix B). Hence, $F_2(\epsilon)$ has at least one minimizer in the feasible set, which in turn guarantees the existence of a unique minimizer.

The upgradation rule for ϵ can be expressed in the following way:

$$\begin{aligned}
&\epsilon^{(t+1)} = (\varepsilon^{(t+1)})^2,\ \varepsilon^{(0)} = \iota^{(0)},\\
&\varepsilon^{(t+1)} = \iota^{(t)} - 2f_2\left[-\Psi((\iota^{(t)})^2)\iota^{(t)} + \sum_{k=1}^{g}\Psi((\iota^{(t)})^2\lambda_k)\iota^{(t)}\lambda_k - \iota^{(t)}\lambda_k\log\pi_k\right],\\
&\iota^{(t+1)} = \varepsilon^{(t+1)} + \frac{t}{t+3}(\varepsilon^{(t+1)} - \varepsilon^{(t)}).
\end{aligned}$$

Optimization with respect to the τ. For optimization of the message length with respect to τ, the optimization problem at hand is:

$$\mathbf{P}_3: \text{ minimize } F_3(\tau) = \sum_{k=1}^{g}\sum_{l=1}^{d}\tau(\log\sigma_{kl} - \log\nu_{kl}) + gd\log(\Gamma(\frac{\tau}{2})2^{\frac{\tau}{2}}), \tau > 0.$$

By the argument used in the earlier case we equivalently consider the following optimization task:

$$\mathbf{EP}_3: \text{ minimize } F_3(\tau) = \sum_{k=1}^{g}\sum_{l=1}^{d} \tau(\log\sigma_{kl} - \log\nu_{kl}) + gd\log(\Gamma(\frac{\tau}{2})2^{\frac{\tau}{2}}), \tau \geqslant 0$$

By the argument used in the earlier case, (Appendix C), the upgradation rule for τ is given by the following rule:

$$\tau^{(t+1)} = (o^{(t+1)})^2,\ o^{(0)} = \upsilon^{(0)},$$
$$o^{(t+1)} = \upsilon^{(t)} - 2f_3\upsilon^{(t)}\left[\sum_{k=1}^{g}\sum_{l=1}^{d}(\log\sigma_{kl} - \log\nu_{kl}) + \frac{gd}{2}(\log 2 + \psi(\frac{(\upsilon^{(t)})^2}{2}))\right],$$
$$\upsilon^{(t+1)} = o^{(t+1)} + \frac{t}{t+3}(o^{(t+1)} - o^{(t)}).$$

Relation with Jeffreys prior-based optimization. We notice that the uninformative Jeffreys prior based optimization of the MML criterion can be derived as a special case of the automated upgraded DNW prior based optimization algorithm with the following fixed choice of the hyper-parameters:

$$\epsilon = \frac{g}{2};\ c = 0;\ \tau = 0;\ \lambda_k = \frac{1}{g};\ \nu_{kl} = 0.$$

In this case, we do not require the automatic upgradation of the model hyper-parameters.

4. Simultaneous feature selection and clustering using mixture models for fuzzy data

From practical aspects, to reduce the effect of noisy variables, selecting the most discriminating features and at the same time eliminating the non-discriminative ones is an important task prior to clustering. Among different concepts of feature relevancy, we adopt the one, which is suitable for the perspective of unsupervised learning of fuzzy data. A feature, which has a common distribution independent of the class labels, is a non-discriminative one and hence is considered irrelevant. It is difficult to determine for sure, the relevance of a feature in unsupervised learning. Hence, we quantify this uncertainty with the introduction of an additional d-dimensional vector-valued random variable denoted by

$$\Phi = [\Phi_1, \Phi_2, \cdots, \Phi_l],$$

where

$$\Phi_l = \begin{cases} 1, & \text{if } l\text{th feature is relevant} \\ 0, & \text{otherwise.} \end{cases}, \forall l = 1, 2, \cdots, d.$$

We evaluate each feature by feature saliency of the lth feature, which is defined as $\rho_l = P(\Phi_l = 1)$, i.e. the probability that the lth feature is relevant.

4.1. Clustering setup for clustering with feature saliency determination

We describe the clustering setup under consideration while developing the GMM (Gaussian Mixture Model) based clustering of fuzzy data with feature selection.

1. Let $\boldsymbol{\rho} = [\rho_1, \rho_2, \cdots, \rho_d]$, where $0 \leqslant \rho_l \leqslant 1, \forall l = 1, 2, \cdots, d$; denote the feature saliencies of the lth feature.
2. The common density corresponding to the lth feature, denoted by $c(x_l \mid \mathcal{L}_l)$, where $\mathcal{L}_l$ completely determines the density corresponding to the lth feature. We shall restrict ourselves to the case, where all the $c(. \mid .)$ are univariate Gaussian distributions, so here, $\mathcal{L}_l = \{m_{\triangle l}, \sigma^2_{\triangle l}\}$ completely determines the corresponding density.

4.2. Development of the fuzzy EM algorithm for GMM based clustering with feature saliency determination

In this GMM, we consider $\{\mathcal{X}, \mathcal{Z}, \boldsymbol{\Phi}\}$ to be the complete data and $\tilde{\mathcal{X}}$ to be the observed or incomplete data. Our target is to estimate the unknown model parameters ($\mathcal{B}$).

4.2.1. Complete data log-likelihood

The complete data log likelihood is given by,

$$
\begin{aligned}
&L(\mathcal{X}, \mathcal{Z}, \Phi \mid \mathcal{B}) = \log \mathbb{P}(\mathcal{X}, \mathcal{Z}, \Phi \mid \mathcal{B}) \\
&= \sum_{k=1}^{g} \log \pi_k \sum_{i=1}^{n} Z_{ik} + \sum_{i=1}^{n}\sum_{k=1}^{g} Z_{ik} \sum_{l=1}^{d} (\Phi_l \log \rho_l + (1-\Phi_l)\log(1-\rho_l)) \\
&- \frac{nd}{2}\log 2\pi - \sum_{i=1}^{n}\sum_{k=1}^{g} Z_{ik}\sum_{l=1}^{d}\Phi_l \log \sigma_{kl} - \frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{g} Z_{ik}\sum_{l=1}^{d}\Phi_l \frac{(x_{il}-m_{kl})^2}{\sigma_{kl}^2} \\
&- \sum_{i=1}^{n}\sum_{k=1}^{g} Z_{ik}\sum_{l=1}^{d}(1-\Phi_l)\log \sigma_{\triangle l} - \frac{1}{2}\sum_{i=1}^{n}\sum_{k=1}^{g} Z_{ik}\sum_{l=1}^{d}(1-\Phi_l)\frac{(x_{il}-m_{\triangle l})^2}{\sigma_{\triangle l}^2}.
\end{aligned}
$$

4.2.2. The E-step

Let the current fit of $\mathcal{B}$ at the tth iteration be denoted by $\mathcal{B}^{(t)}$. The E-step consists of calculating the following quantity:

$$
\begin{aligned}
&Q(\mathcal{B}, \mathcal{B}^{(t)}) = \mathbb{E}_{\mathcal{B}^{(t)}}(L(\mathcal{X}, \mathcal{Z} \mid \mathcal{B}) \mid \tilde{\mathcal{X}}) \\
&= \sum_{k=1}^{g} \log \pi_k \sum_{i=1}^{n} \zeta_{ik}^{(t)} - \frac{nd}{2}\log 2\pi + \sum_{l=1}^{d}\log(1-\rho_l)\sum_{i=1}^{n}\sum_{k=1}^{g} r_{ikl}^{(t)} \\
&+ \sum_{l=1}^{d}\log \rho_l \sum_{i=1}^{n}\sum_{k=1}^{g} s_{ikl}^{(t)} - \sum_{k=1}^{g}\sum_{l=1}^{d}\log \sigma_{kl}^2 \sum_{i=1}^{n} s_{ikl}^{(t)} - \sum_{k=1}^{g}\sum_{l=1}^{d}\log \sigma_{\triangle l}^2 \sum_{i=1}^{n} s_{ikl}^{(t)} \\
&- \frac{1}{2}\sum_{k=1}^{g}\sum_{l=1}^{d}\frac{1}{\sigma_{kl}^2}\sum_{i=1}^{n}\left[\xi_{ikl}^{(t)} s_{ikl}^{(t)} - 2\eta_{ikl}^{(t)} s_{ikl}^{(t)} m_{kl} + m_{kl}^2 s_{ikl}^{(t)}\right] \\
&- \frac{1}{2}\sum_{k=1}^{g}\sum_{l=1}^{d}\frac{1}{\sigma_{\triangle l}^2}\sum_{i=1}^{n}\left[\tau_{ikl}^{(t)} r_{ikl}^{(t)} - 2\omega_{ikl}^{(t)} r_{ikl}^{(t)} m_{\triangle l} + m_{\triangle l}^2 r_{ikl}^{(t)}\right],
\end{aligned}
$$

where,

$$
\begin{aligned}
f_{ikl}^{(t)} &= \mathbb{P}_{\mathcal{B}^{(t)}}(\tilde{\mathbf{x}}_{il} \mid Z_{ik} = 1) = \rho_l^{(t)} \int \mu_{\tilde{\mathbf{x}}_{il}}(x)\, p(x \mid m_{kl}^{(t)}, {\sigma_{kl}^{(t)}}^2) dx \\
&\qquad + (1-\rho_l^{(t)}) \int \mu_{\tilde{\mathbf{x}}_{il}}(y)\, p(y \mid m_{\triangle l}^{(t)}, {\sigma_{\triangle l}^{(t)}}^2) dy, \\
f_{ik}^{(t)} &= \mathbb{P}_{\mathcal{B}^{(t)}}(\tilde{\mathbf{x}}_i \mid Z_{ik} = 1) = \prod_{l=1}^{d} f_{ikl}^{(t)}, \\
\triangle_i^{(t)} &= \mathbb{P}_{\mathcal{B}^{(t)}}(\tilde{\mathbf{x}}_i) = \sum_{k=1}^{g} \pi_k^{(t)} f_{ik}^{(t)}, \\
\zeta_{ik}^{(t)} &= \mathbb{E}_{\mathcal{B}^{(t)}}(Z_{ik} \mid \mathcal{X}) = \frac{\mathbb{P}_{\mathcal{B}^{(t)}}(\tilde{\mathbf{x}}_i \mid Z_{ik} = 1)\mathbb{P}_{\mathcal{B}^{(t)}}(Z - ik = 1)}{\mathbb{P}_{\mathcal{B}^{(t)}}(\tilde{\mathbf{x}}_i)} = \frac{f_{ik}^{(t)}\pi_k^{(t)}}{\triangle_i^{(t)}}, \\
e_{ikl}^{(t)} &= \mathbb{P}_{\mathcal{B}^{(t)}}(\tilde{\mathbf{x}}_i \mid Z_{ik}\Phi_l = 1) = \int \mu_{\tilde{\mathbf{x}}_{il}}(x_l)\, p(x_l \mid \{m_{kl}^{(t)}, {\sigma_{kl}^{(t)}}^2\}) dx_l \\
s_{ikl}^{(t)} &= \mathbb{E}_{\mathcal{B}^{(t)}}(Z_{ik}\Phi_l \mid \mathcal{X}) = \frac{e_{ikl}^{(t)}\pi_k^{(t)}\rho_l^{(t)}}{\triangle_i^{(t)}} \prod_{l' \neq l} f_{ikl'}^{(t)}, \\
h_{ikl}^{(t)} &= \mathbb{P}_{\mathcal{B}^{(t)}}(\tilde{\mathbf{x}}_i \mid Z_{ik}(1-\Phi_l) = 1) = \int \mu_{\tilde{\mathbf{x}}_{il}}(x_l)\, p(x_l \mid \{m_{\triangle l}^{(t)}, (\sigma_{\triangle l}^{(t)})^2\}) dx_l,
\end{aligned}
$$

$$r_{ikl}^{(t)} = \mathbb{E}_{\mathcal{B}^{(t)}}(Z_{ik}(1-\Phi_l) \mid \mathcal{X}) = \zeta_{ik}^{(t)} - s_{ikl}^{(t)} = \frac{e_{ikl}^{(t)} \pi_k^{(t)} (1-\rho_l^{(t)})}{\triangle_i^{(t)}} \prod_{l' \neq l} f_{ikl'}^{(t)},$$

$$\xi_{ikl}^{(t)} = \mathbb{E}_{\mathcal{B}^{(t)}}(x_{il}^2 \mid \tilde{\mathcal{X}}, Z_{ik}\Phi_l = 1) = \frac{\int x_{il}^2 \mu_{\tilde{\mathbf{x}}_{il}}(x_l)\, p(x_l \mid \{m_{kl}^{(t)}, {\sigma_{kl}^{(t)}}^2\}) dx_l}{e_{ikl}^{(t)}},$$

$$\eta_{ikl}^{(t)} = \mathbb{E}_{\mathcal{B}^{(t)}}(x_{il} \mid \tilde{\mathcal{X}}, Z_{ik}\Phi_l = 1) = \frac{\int x \mu_{\tilde{\mathbf{x}}_{il}}(x)\, p(x \mid \{m_{kl}^{(t)}, {\sigma_{kl}^{(t)}}^2\}) dx}{e_{ikl}^{(t)}},$$

$$\tau_{ikl}^{(t)} = \mathbb{E}_{\mathcal{B}^{(t)}}(x_{il}^2 \mid \tilde{\mathcal{X}}, Z_{ik}(1-\Phi_l) = 1) = \frac{\int x^2 \mu_{\tilde{\mathbf{x}}_{il}}(x)\, p(x \mid \{m_{\triangle l}^{(t)}, {\sigma_{\triangle l}^{(t)}}^2\}) dx}{h_{ikl}^{(t)}},$$

$$\omega_{ikl}^{(t)} = \mathbb{E}_{\mathcal{B}^{(t)}}(x_{il} \mid \tilde{\mathcal{X}}, Z_{ik}(1-\Phi_l) = 1) = \frac{\int x \mu_{\tilde{\mathbf{x}}_{il}}(x)\, p(x \mid \{m_{\triangle l}^{(t)}, {\sigma_{\triangle l}^{(t)}}^2\}) dx}{h_{ikl}^{(t)}},$$

$$\mathbb{E}_{\mathcal{B}^{(t)}}(Z_{ik}\Phi_l x_{il}^2 \mid \tilde{\mathcal{X}}) = \xi_{ikl}^{(t)} s_{ikl}^{(t)},$$
$$\mathbb{E}_{\mathcal{B}^{(t)}}(Z_{ik}\Phi_l x_{il} \mid \tilde{\mathcal{X}}) = \eta_{ikl}^{(t)} s_{ikl}^{(t)},$$
$$\mathbb{E}_{\mathcal{B}^{(t)}}(Z_{ik}(1-\Phi_l) x_{il}^2 \mid \tilde{\mathcal{X}}) = \tau_{ikl}^{(t)} r_{ikl}^{(t)},$$
$$\mathbb{E}_{\mathcal{B}^{(t)}}(Z_{ik}(1-\Phi_l) x_{il} \mid \tilde{\mathcal{X}}) = \omega_{ikl}^{(t)} r_{ikl}^{(t)}.$$

4.2.3. The M-step

In this step, the partial derivative of $Q(\mathcal{B}, \mathcal{B}^{(t)})$ with respect to model parameters are set to zero, to obtain the upgradation rule for each of them in each iteration. The derived upgradation rules for the model parameters are as follows:

$$\pi_k^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} \zeta_{ik}^{(t)};\ m_{kl}^{(t+1)} = \sum_{i=1}^{n} \eta_{ikl}^{(t)} s_{ikl}^{(t)} / \sum_{i=1}^{n} s_{ikl}^{(t)},$$

$$\sigma_{kl}^{(t+1)} = \sqrt{\sum_{i=1}^{n} \left[\eta_{ikl}^{(t)} s_{ikl}^{(t)} - 2\eta_{ikl}^{(t)} s_{ikl}^{(t)} m_{kl}^{(t+1)} + (m_{kl}^{(t+1)})^2 s_{ikl}^{(t)} \right] / \sum_{i=1}^{n} s_{ikl}^{(t)}},$$

$$\rho_l^{(t+1)} = \sum_{i=1}^{n} \sum_{k=1}^{g} s_{ikl}^{(t)} / n;\ m_{\triangle l}^{(t+1)} = \sum_{i=1}^{n} \omega_{ikl}^{(t)} r_{ikl}^{(t)} / \sum_{i=1}^{n} r_{ikl}^{(t)},$$

$$\sigma_{\triangle l}^{(t+1)} = \sqrt{\sum_{i=1}^{n} \sum_{k=1}^{g} \left[\tau_{ikl}^{(t)} r_{ikl}^{(t)} - 2\omega_{ikl}^{(t)} r_{ikl}^{(t)} m_{\triangle l}^{(t+1)} + (m_{\triangle l}^{(t+1)})^2 r_{ikl}^{(t)} \right] / \sum_{i=1}^{n} r_{ikl}^{(t)}}.$$

5. Automatic feature and component number selection in fuzzy data with MML criterion

To avoid ending up with a degenerate solution, we need some prior knowledge about the component numbers, feature relevancy of each of the features under consideration as well as a good initialization. For addressing all these problems simultaneously, we adopt the MML approach. This ensures automatic model selection which performs automatic component number selection and optimum feature subset selection. In the following subsections, we concentrate on deriving the mathematical form of the MML criterion in this situation by approximation of the observed Fisher information matrix. Next we develop the MML criterion for different choices of the non-informative and informative priors with automatic upgradation and the corresponding upgradation rules for the fuzzy EM algorithm are for simultaneous feature selection and component number selection.

5.1. Approximation of the Fisher information matrix

We follow the approach in Section 4 by replacing the expected Fisher information matrix with the block-diagonal complete data Fisher information Matrix $\mathbf{I}_c(\mathcal{A})$ which provides an upper-bound to the expected Fisher information matrix [26]. Here we use the fact that in GMM with feature selection, the complete data corresponding to both of crisp and fuzzy incomplete data are same, we adopt this technique in the fuzzy data scenario.

$$\begin{aligned}\mathbf{I}_c(\mathcal{B}) = n \text{ block-diag}\big\{&\mathbf{I}(\boldsymbol{\pi}), \tfrac{1}{\rho_1(1-\rho_1)}, \tfrac{1}{\rho_2(1-\rho_2)}, \cdots, \tfrac{1}{\rho_d(1-\rho_d)}, \pi_1\rho_1\mathbf{I}^{(1)}(\mathcal{T}_{11}),\\ &\pi_1\rho_2\mathbf{I}^{(1)}(\mathcal{T}_{12}), \cdots, \pi_1\rho_d\mathbf{I}^{(1)}(\mathcal{T}_{1d}), \pi_2\rho_1\mathbf{I}^{(1)}(\mathcal{T}_{21}), \cdots, \pi_2\rho_d\mathbf{I}^{(1)}(\mathcal{T}_{2d}), \cdots,\\ &\pi_g\rho_1\mathbf{I}^{(1)}(\mathcal{T}_{g1}), \cdots, \pi_g\rho_d\mathbf{I}^{(1)}(\mathcal{T}_{gd}), (1-\rho_1)\mathbf{I}^{(1)}(\mathcal{L}_1), \cdots, (1-\rho_d)\mathbf{I}^{(1)}(\mathcal{L}_d)\big\},\end{aligned}$$

where $\mathbf{I}^{(1)}(\mathcal{T}_{kl})$ is the Fisher information matrix for a single observation corresponding to the lth dimension; known to have been produced from the component number $k, k = 1, 2, \cdots, g; l = 1, 2, \cdots, d$. $\mathbf{I}^{(1)}(\mathcal{L}_l)$ is the Fisher information matrix for a single observation corresponding to the lth dimension; known to have been produced from the common Gaussian distribution corresponding to the lth dimension. $\mathbf{I}(\boldsymbol{\pi})$ denotes the Fisher information matrix of the multinomial distribution with parameters $(\pi_1, \pi_2, \cdots, \pi_g)$. Under this, we have the following overall approximation of the determinant of expected Fisher information matrix which is a famous and thoroughly studied approximation in the literature [17].

$$\begin{aligned}|\mathbf{I}_c(\mathcal{A})| &= |\mathbf{I}_n(\boldsymbol{\pi})|\left[\prod_{k=1}^{g}\prod_{l=1}^{d}|n\pi_k\rho_l\mathbf{I}^{(1)}(\mathcal{T}_{kl})|\right]\left[\prod_{l=1}^{d}|n(1-\rho_l)\mathbf{I}^{(1)}(\mathcal{L}_l)|\right]\\ &= \frac{n^g}{\prod_{k=1}^{g}\pi_k}\prod_{k=1}^{g}\prod_{l=1}^{d}\frac{2(n\pi_k\rho_l)^2}{\sigma_{kl}^4}\prod_{l=1}^{d}\frac{2(n(1-\rho_l))^2}{\sigma_{kl}^4}.\end{aligned}$$

This approximation may also be used in the Laplace-empirical criterion for automatic component number selection and optimum feature subset selection in the fuzzy data setup.

5.2. Development of the MML criterion and optimization algorithm for fuzzy EM algorithm with feature and component number selection with uninformative priors

In this scenario, for the prior densities of the model parameters, we assume that the mixing probabilities, the component parameters, the feature saliencies corresponding to all the variables, and the parameters determining the common distribution corresponding to the features are independent, i.e. we want the prior to be of the following form:

$$p(\mathcal{B}) = p_1(\pi_1, \pi_2, \cdots, \pi_g)\left[\prod_{k=1}^{g}\prod_{l=1}^{d}p_2(\mathcal{T}_{kl})\right]\left[\prod_{l=1}^{d}p_3(\rho_l)p_4(\mathcal{L}_l)\right].$$

To express our lack of knowledge about the unknown parameters, for each part of the aforementioned priors, we choose the standard non-informative Jeffreys prior. The Jeffreys prior is given by the following formula:

$$\begin{aligned}&p_1(\pi_1, \pi_2, \cdots, \pi_g) \propto \sqrt{|\mathbf{I}(\boldsymbol{\pi})|} = 1/\sqrt{\prod_{k=1}^{g}\pi_k},\\ &p_2(\mathcal{T}_{kl}) \propto \sqrt{|\mathbf{I}^{(1)}(\mathcal{T}_{kl})|}, \forall k = 1, 2, \cdots, g; \forall l = 1, 2, \cdots, d,\\ &p_3(\rho_l) \propto \frac{1}{\rho_l(1-\rho_l)}, \forall l = 1, 2, \cdots, d,\\ &p_4(\mathcal{L}_l) \propto \sqrt{|\mathbf{I}^{(1)}(\mathcal{L}_l)|}, \forall l = 1, 2, \cdots, g.\end{aligned}$$

Incorporation of the aforementioned priors in the MML criterion, dropping the order-one terms, and components with zero mixing probabilities and features with saliency equal to 0 or 1 (the additional term being a constant, can be ignored) leads us to the following closed form approximate expression for $M(\tilde{\mathcal{X}}, \mathcal{B})$:

$$\sum_{k:\pi_k>0}\sum_{l:\rho_l>0}\log(n\rho_l\pi_k)+\sum_{l:\rho_l<1}\log(n(1-\rho_l))+\frac{1}{2}(g_P+d_P)\log n-L(\tilde{\mathcal{X}}\mid\mathcal{B}),$$

where, g_P and d_P are respectively the numbers of components with non-zero mixing probability and features with non-zero, non-one features salience. In the aforementioned optimization task, for a fixed number of components with positive probability and a number of features with non-one positive feature saliency, we can neglect the constant terms. Then, the resulting optimization task can be approached from the Bayesian point of view as the MAP estimation procedure resulting from adoption of flat or no prior on the parameters specifying the model components and adoption of the following priors:

1. Dirichlet type prior on the non-zero mixing probabilities:

$$p(\pi_1,\pi_2,\cdots,\pi_g)\propto\exp(-d_P\sum_{k:\pi_k>0}\log\pi_k).$$

2. Dirichlet type prior on the non-zero, non-one feature saliencies:

$$p(\rho_1,\rho_2,\cdots,\rho_g)\propto\exp\left[-g_P\sum_{l:1>\rho_l>0}\log\rho_l\right]\exp\left[\sum_{l:1>\rho_l>0}\log(1-\rho_l)\right].$$

We also observe that these priors are conjugate priors to the complete data log-likelihoods. The upgradation rule for the mixing probabilities in the fuzzy EM algorithm with automatic feature and component number selection is modified as follows:

$$\pi_k^{(t+1)}=\left[\max\{0,(\sum_{i=1}^{n}\zeta_{ik}^{(t)})-d_P\}\right]/\left[\sum_{k=1}^{g}\max\{0,(\sum_{i=1}^{n}\zeta_{ik}^{(t)})-d_P\}\right],$$

$$\rho_l^{(t+1)}=\frac{\left[\max\{0,(\sum_{i=1}^{n}\sum_{k=1}^{g}s_{ikl}^{(t)})-g_P\}\right]}{\left[\max\{0,(\sum_{i=1}^{n}\sum_{k=1}^{g}s_{ikl}^{(t)})-g_P\}\right]+\left[\max\{0,(\sum_{i=1}^{n}\sum_{k=1}^{g}r_{ikl}^{(t)})-1\}\right]}.$$

Due to the use of a flat prior on $\mathcal{T}_{kl}$ and $\mathcal{L}_l$, the upgradation rules for $\mathcal{T}_{kl}$ and $\mathcal{L}_l$ is exactly same as that corresponding to the fuzzy EM algorithm with simultaneous feature selection and clustering.

It is worth noticing that the upgradation rule obtained here performs automatic component annihilation as in the earlier case. The same is true for feature saliencies to go to zero or one. This saves us the trouble of dealing with almost singular covariance matrices. This automated component number selection and the component annihilation gives us freedom to alleviate the need of a good initialization point.

5.3. Development of the MML criterion and optimization algorithm for fuzzy EM algorithm with feature and component number selection based on informative priors with automated upgradation

In this section, we concentrate on some informative priors and study the MML criterion and the corresponding E-step, M-step of the fuzzy EM algorithm with simultaneous feature selection and clustering along with the additional hyper-parameter upgradation step.

5.3.1. Dirichlet–Normal–Wishart prior

This hyper-parameter based prior is a product of the Dirichlet–Prior on the mixing probabilities and a joint Normal–Wishart prior on the component parameters. We further assume the Dirichlet–Prior on each of the feature saliencies and joint Normal–Wishart prior on the parameters specifying each of the components. (For notational simplicity, we assume that all the mixing probabilities are non-zero and feature saliencies are non-zero and non-one. In the presence of any mixing probability equal to zero or feature saliency equal to zero (one), we simply drop the component or the

feature (common distribution corresponding to that feature) as done in earlier case.) Then we have,

$$p(\mathcal{B}) = D(\boldsymbol{\pi} \mid \boldsymbol{\lambda}, \epsilon) \prod_{k=1}^{g} \prod_{l=1}^{d} q(m_{kl} \mid \{\mu_{kl}, \frac{\sigma_{kl}^2}{c}\}) W(\frac{1}{\sigma_{kl}^2} \mid \{\nu_{kl}^2, \tau\}) \prod_{l=1}^{d} \left[D(\boldsymbol{\rho}_l \mid \boldsymbol{\varphi}_l, \vartheta_l) q(m_{\triangle l} \mid \{\mu_{\triangle l}, \frac{\sigma_{\triangle l}^2}{c_{\triangle l}}\}) W(\frac{1}{\sigma_{\triangle l}^2} \mid \{\nu_{\triangle l}^2, \tau_{\triangle l}\}) \right],$$

where $\boldsymbol{\rho}_l = [\rho_{l_1}, \rho_{l_1}]$, $\rho_{l_1} = \rho_l$, $\rho_{l_2} = 1 - \rho_l$. With this prior, the MML criterion is as follows:

$$\begin{aligned}
M(\tilde{\mathcal{X}}, \mathcal{B}) \approx & -\log(\frac{\Gamma(\epsilon)}{\prod_{k=1}^{g} \Gamma(\epsilon\lambda_k)}) + \sum_{k=1}^{g} (\frac{2d+1}{2} - \epsilon\lambda_k) \log(\pi_k) - gd \log(\frac{\sqrt{c}}{\sqrt{2\pi}}) \\
& - \sum_{k=1}^{g} \sum_{l=1}^{d} \left[-\frac{c}{2} \frac{(m_{kl} - \mu_{kl})^2}{\sigma_{kl}^2} - \log(\sigma_{kl}) - \frac{1}{2\sigma_{kl}^2} \nu_{kl}^2 - \tau \log(\sigma_{kl}) + \tau \log(\nu_{kl}) \right] \\
& - L(\tilde{\mathcal{X}} \mid \mathcal{A}) + (\frac{2dg + 2d + g}{2}) \log n + (dg + d) \log 2 + \frac{N_P}{2} (1 + \log(\kappa_{N_P})) \\
& - \sum_{l=1}^{d} \log(\frac{\Gamma(\vartheta_l)}{\prod_{s=1}^{2} \Gamma(\vartheta \varphi_{l_s})}) + \sum_{l=1}^{d} \sum_{s=1}^{2} (\frac{2g+1}{2} - \vartheta \varphi_{l_s}) \log(\rho_{l_s}) - \sum_{l=1}^{d} \log(\frac{\sqrt{c_{\triangle l}}}{\sqrt{2\pi}}) \\
& - \sum_{l=1}^{d} -\frac{c_{\triangle l}}{2} \frac{(m_{\triangle l} - \mu_{\triangle l})^2}{\sigma_{\triangle l}^2} - \log(\sigma_{\triangle l}) - \frac{1}{2\sigma_{\triangle l}^2} \nu_{\triangle l}^2 - \tau \log(\sigma_{\triangle l}) + \tau_{\triangle l} \log(\nu_{\triangle l}) \\
& + \sum_{l=1}^{d} \log(\Gamma(\frac{\tau_{\triangle l}}{2}) 2^{\frac{\tau_{\triangle l}}{2}}) + gd \log(\Gamma(\frac{\tau}{2}) 2^{\frac{\tau}{2}}).
\end{aligned}$$

In the aforementioned optimization task, for a fixed number of components with positive probability and features with non-one positive feature saliency, we can neglect the constant terms (in a particular iteration the hyper-parameters are assumed to be constant) and the resulting optimization task can be approached from the Bayesian point of view as the MAP estimation procedure resulting from adoption of the following priors:

1. Dirichlet type prior on the mixing probabilities:

$$p(\pi_1, \pi_2, \cdots, \pi_g) \propto \exp\left[-\sum_{k=1}^{g} (\frac{2d_P + 1}{2} - \epsilon\lambda_k) \log \pi_k \right].$$

2. Joint Normal–Wishart prior on component parameters:

$$\prod_{k=1}^{g} \prod_{l=1}^{d} q(m_{kl} \mid \{\mu_{kl}, \frac{\sigma_{kl}^2}{c}\}) W(\frac{1}{\sigma_{kl}^2} \mid \{\nu_{kl}^2, \tau + 2\}).$$

3. Dirichlet type prior on the feature saliencies:

$$p(\rho_1, \rho_2, \cdots, \rho_g) \propto \exp\left[-\sum_{l=1}^{d} (\frac{2g_P + 1}{2} - \varphi_{l_s} \vartheta_l) \log \rho_{l_s} \right].$$

4. Joint Normal–Wishart prior on parameters of each of the common distributions:

$$q(m_{kl} \mid \{\mu_{\triangle l}, \frac{\sigma_{\triangle l}^2}{c_{\triangle l}}\}) W(\frac{1}{\sigma_{\triangle l}^2} \mid \{\nu_{\triangle l}^2, \tau_{\triangle l} + 2\}).$$

We also observe that these priors are conjugate priors. Hence the upgradation rule for the mixing probabilities and the model parameters in the fuzzy EM algorithm with simultaneous feature selection and clustering can be modified as

follows:

$$\pi_k^{(t+1)} = \frac{\left[max\{0, (\sum_{i=1}^n \zeta_{ik}^{(t)}) - (\frac{2d+1}{2} - \epsilon^{(t)}\lambda_k^{(t)})\}\right]}{\sum_{k=1}^g \left[max\{0, (\sum_{i=1}^n \zeta_{ik}^{(t)}) - (\frac{2d+1}{2} - \epsilon^{(t)}\lambda_k^{(t)})\}\right]},$$

$$m_{kl}^{(t+1)} = \frac{\sum_{i=1}^n \eta_{ikl}^{(t)} s_{ikl}^{(t)} + c^{(t)}\mu_{kl}^{(t)}}{\sum_{i=1} n s_{ikl}^{(t)} + c^{(t)}},$$

$$\sigma_{kl}^{(t+1)} = \left[\begin{array}{c}\sum_{i=1}^n (\eta_{ikl}^{(t)} s_{ikl}^{(t)} - 2\eta_{ikl}^{(t)} s_{ikl}^{(t)} m_{kl}^{(t+1)} + (m_{kl}^{(t+1)})^2 s_{ikl}^{(t)}) \\ + c^{(t)}(m_{kl}^{(t+1)} - \mu_{kl}^{(t)})^2 + (\nu_{kl}^{(t)})^2\end{array}\right]^{\frac{1}{2}} / \left[\sum_{i=1} n s_{ikl}^{(t)} + \tau^{(t)} + 1\right]^{\frac{1}{2}},$$

$$\rho_l^{(t+1)} = \frac{\max\{0, (\sum_{i=1}^n \sum_{k=1}^g s_{ik}^{(t)}) - (\frac{2g+1}{2} - \vartheta_l^{(t)}\varphi_{l_1}^{(t)})\}}{\left[\begin{array}{l}\max\{0, (\sum_{i=1}^n \sum_{k=1}^g s_{ik}^{(t)}) - (\frac{2g+1}{2} - \vartheta_l^{(t)}\varphi_{l_1}^{(t)})\} \\ + \max\{0, (\sum_{i=1}^n \sum_{k=1}^g r_{ik}^{(t)}) - (\frac{2g+1}{2} - \vartheta_l^{(t)}\varphi_{l_2}^{(t)})\end{array}\right]},$$

$$m_{\triangle l}^{(t+1)} = \frac{\sum_{i=1}^n \sum_{k=1}^g \omega_{ikl}^{(t)} r_{ikl}^{(t)} + c_{\triangle l}^{(t)}\mu_{\triangle l}^{(t)}}{\sum_{i=1} n \sum_{k=1}^g r_{ikl}^{(t)} + c_{\triangle l}^{(t)}},$$

$$\sigma_{\triangle l}^{(t+1)} = \left[\begin{array}{c}\sum_{i=1}^n \sum_{k=1}^g (\tau_{ikl}^{(t)} r_{ikl}^{(t)} - 2\omega_{ikl}^{(t)} r_{ikl}^{(t)} m_{\triangle l}^{(t+1)} + (m_{\triangle l}^{(t+1)})^2 r_{ikl}^{(t)}) + \\ c_{\triangle l}^{(t)}(m_{\triangle l}^{(t+1)} - \mu_{\triangle l}^{(t)})^2 + (\nu_{\triangle l}^{(t)})^2\end{array}\right]^{\frac{1}{2}} / \left[\sum_{i=1}^n r_{ikl}^{(t)} + \tau_{\triangle l}^{(t)} + 1\right]^{\frac{1}{2}}.$$

Now, for maximum utilization of the data gathered about an unknown model parameter in each iteration, we also automatically upgrade the hyper-parameters by optimizing the MML criterion with respect to the hyper parameters, with the help of the following equations:

$$\mu_{kl}^{(t+1)} = m_{kl}^{(t)};\ c^{(t+1)} = \frac{gd}{\sum_{k=1}^d \sum_{l=1}^d \frac{m_{kl}^{(t)} - \mu_{kl}^{(t)}{}^2}{\sigma_{kl}^{(t)}}};\ \nu_{kl}^{(t+1)} = \sqrt{\tau^{(t)}}\sigma_{kl}^{(t)}.$$

$$\mu_{\triangle l}^{(t+1)} = m_{\triangle l}^{(t)};\ c_{\triangle l}^{(t+1)} = 1/\left[\frac{(m_{\triangle l}^{(t)} - \mu_{\triangle l}^{(t)})}{(\sigma_{\triangle l}^{(t)})^2}\right];\ \nu_{\triangle l}^{(t+1)} = (\tau_{\triangle l}^{(t)})^{\frac{1}{2}}\sigma_{\triangle l}^{(t)}.$$

Now, we observe that for optimization with respect to $\lambda_k, \epsilon, \tau, \tau_{\triangle l}, \vartheta_l, \varphi_{l_s}$ we do not get a closed form of optimizer. We take a closer look at each of the optimization tasks to guarantee the existence of a unique optimizer in each of the aforementioned cases and develop a gradient descent algorithm to obtain an approximation of the optimizer with user specified precision.

Optimization with respect to the λ_k's, ϵ and τ can be done as in the earlier case (see Section 3.5). As far as the optimization of the remaining hyper-parameters are concerned, it can be done in the following way:

Optimization with respect to the $\tau_{\triangle l}$. For optimization of the message length with respect to $\tau_{\triangle l}$ the optimization problem at hand is:

$$\mathbf{P}_{4l}:\ \text{minimize}\ F_{4l}(\tau_{\triangle l}) = \tau_{\triangle l}(\log\sigma_{\triangle l} - \log\nu_{\triangle l}) + \log(\Gamma(\frac{\tau_{\triangle l}}{2})2^{\frac{\tau_{\triangle l}}{2}}), \tau_{\triangle l} > 0.$$

By the argument used in earlier case, the upgradation rule for $\tau_{\triangle l}$ is given by the following rule:

$$\tau_{\triangle l}^{(t+1)} = (\phi_{\triangle l}^{(t+1)})^2;\ \phi_{\triangle l}^{(0)} = \upsilon_{\triangle l}^{(0)},$$

$$\phi_{\triangle l}^{(t+1)} = \upsilon_{\triangle l}^{(t)} - 2f_{4_l}\upsilon_{\triangle l}^{(t)}\left[(\log\sigma_{\triangle l} - \log\nu_{\triangle l}) + \frac{1}{2}(\log 2 + \psi(\frac{(\upsilon_{\triangle l}^{(t)})^2}{2}))\right],$$

$$\upsilon_{\triangle l}^{(t+1)} = \phi_{\triangle l}^{(t+1)} + \frac{t}{t+3}(\phi_{\triangle l}^{(t+1)} - \phi_{\triangle l}^{(t)}).$$

Optimization with respect to the ϑ_l. For optimization of the message length with respect to ϑ_l the optimization problem at hand is:

$$\mathbf{P}_{5l}: \text{ minimize } F_{5l}(\vartheta_l) = -\log\Gamma(\vartheta_l) + \log(\prod_{s=1}^{2}\Gamma(\vartheta_l\varphi_{l_s})) - \sum_{s=1}^{2}\vartheta_l\varphi_{l_s}\log\rho_{l_s}; \vartheta_l > 0.$$

The aforementioned optimization task can be performed similarly as in earlier cases. The upgradation rule for ϑ_l is given by the following rule:

$$\vartheta_l^{(t+1)} = (\varrho_l^{(t+1)})^2;\ \varrho_l^{(0)} = \chi^{(0)},$$
$$\varepsilon^{(t+1)} = \chi^{(t)} - 2f_{5_l}\left[-\Psi((\chi^{(t)})^2)\chi^{(t)} + \sum_{k=1}^{g}\Psi((\chi^{(t)})^2\lambda_k)\chi^{(t)}\lambda_k - \chi^{(t)}\lambda_k\log\pi_k\right],$$
$$\chi^{(t+1)} = \varrho^{(t+1)} + \frac{t}{t+3}(\varrho^{(t+1)} - \varrho^{(t)}).$$

Optimization with respect to the $\boldsymbol{\varphi}_l$'s. For optimization with respect to $\boldsymbol{\varphi}_l$ the optimization problem at hand is:

$$\mathbf{P}_{6l}: \text{ minimize } F_{6l}(\boldsymbol{\varphi}_l) = \log(\prod_{s=1}^{2}\Gamma(\vartheta_l\varphi_{l_s})) - \sum_{s=1}^{2}\vartheta_l\varphi_{l_s}\log\rho_{l_s}, \sum_{s=1}^{2}\rho_{l_s} = 1,$$
$$\rho_{l_s} > 0.$$

The aforementioned optimization task can be performed similarly as in earlier cases. The upgradation rule for $\varphi_{l_s}, s = 1, 2, \forall l = 1, 2, \cdots, d$ can be expressed as:

$$\varphi_{l_s}^{(t+1)} = (\varsigma_{l_s}^{(t+1)})^2;\ (\varsigma_{l_s}^{(0)}, a_l^{(0)}) = (\varpi_{l_s}^{(0)}, b_l^{(0)}),$$
$$\varsigma_{l_s}^{(t+1)} = \varpi_{l_s}^{(t)} - f_{6_l}\{2\Psi(\vartheta_l(\varpi_{l_s}^{(t)})^2)\vartheta_l\varpi_k^{(t)} - 2\vartheta_l\varpi_k^{(t)}\log\rho_{l_s} + b_l^{(t)}\},$$
$$a_l^{(t+1)} = b_l^{(t)} - f_{6_l}(\sum_{s=1}^{2}(\varpi_{l_s}^{(t)})^2 - 1),$$
$$(\varpi_{l_s}^{(t+1)}, b_l^{(t+1)}) = (\varsigma_{l_s}^{(t+1)}, a_l^{(t+1)}) + \frac{t}{t+3}((\varsigma_{l_s}^{(t+1)}, a_l^{(t+1)}) - (\varsigma_{l_s}^{(t)}, a_l^{(t)})).$$

Thus, we carry out the development of a mixture model based fuzzy clustering under some assumptions. Next, we briefly outline their importance and applicability in a real-life scenario.

1. **Assumption of diagonal covariance matrix**: This assumption is pretty common in the perspective of high-dimensional data, such as latent class models [19], hidden Markov models [20], naive Bayes classifier etc. Moreover, this structure plays an instrumental role in the introduction of feature saliency in the perspective of mixture model clustering [17].
2. **Assumption of separability of fuzzy membership**: This assumption plays an instrumental role in the development of the fuzzy EM algorithm and its applicability to the clustering problems [5,27].

6. Validity indices for mixture model clustering performances

For comparison of the clustering performance of a mixture model, we look at $t_{ik} = P_{\Delta_{opt}}(Z_{ik} = 1 \mid \tilde{\mathcal{X}})$, i.e. the probability (under estimated value of Δ_{opt}) of the ith data point to belong to the kth component. We classify the ith data point in $\hat{k}$th component, where $\hat{k}$th is defined as follows,

$$\hat{k} = \text{argmax}_k\, t_{ik}.$$

Now we have a label for each of the data points i.e. a "crisp partition" of the data, so we can compare the clustering performance by using the existing indices like the Rand index [28], Minkowski score [29], and Adjusted Rand Index (ARI) [30].

The problem with this method is, we need to know the actual label of the data to measure the clustering performance, which is impossible in most of the cases. Another problem with this crisp partitioning is that assigning the data point to the k corresponding with highest t_{ik}, we are losing significant information about the clustering performance. As an example, we consider the following scenario. Let T' and T'' be the assignment matrices of two different clusterings. Let the ith rows of these matrices be given by:

$$T'_{i*} = (0.25\ 0.25\ 0.5) \text{ and } T''_{i*} = (0.1\ 0.15\ 0.75).$$

In both the cases, we shall assign the ith data point to $k = 3$, but in the second case $t''_{i3} > t'_{i3}$ and we are not using this information. In the form of T, t_{ik} is analogous to u_{ik} = membership of ith data point in kth cluster/component and we also have:

$$\sum_{k=1}^{g} t_{ik} = 1, \ \forall i = 1, 2, \cdots, n; \ 0 \leqslant t_{ik} \leqslant 1, \ \forall i = 1, 2, \cdots, n; \forall k = 1, 2, \cdots, g.$$

Thus, we can intuitively adopt the cluster validity indices defined for the membership matrix (Partition Coefficient [31], Partition Entropy [32], Modified Partition Coefficient [33] etc.) here too. Now, we formally state how we can adopt these measures in the setup of our concern.

6.1. Probabilistic Partition Coefficient

For a given matrix T, we define the Probabilistic Partition Coefficient (V_{PPC}) of the matrix in the following way:

$$V_{PPC} = \left[\sum_{i=1}^{n} \sum_{k=1}^{g} t_{ik}^2 \right] / n. \tag{4}$$

Note that

$$V_{PPC} \leqslant \left[\sum_{i=1}^{n} \sum_{k=1}^{g} t_{ik} \right] / n = 1,$$

and

$$V_{PPC} \geqslant \left[\sum_{i=1}^{n} g(\sum_{k=1}^{g} t_{ik}/g)^2 \right] / n = \frac{1}{g},$$

where equality holds if and only if $t_{ik} = \frac{1}{g}, \forall i = 1, 2, \cdots, n; k = 1, 2, \cdots, g$. This is the most pathological case, where all the data points can belong to any of the g components, with equal probability. It is completely uncertain in which component any data point belongs; i.e. no cluster structure is found at all in the data, this may be due to the data or the algorithm used. The further the value of V_{PPC} from $1/g$, the lower the uncertainty in the partition. So, we can say that the higher the value of V_{PPC}, the further it is from complete uncertainty. It is not true that V_{PPC} ensures good partitioning near the value of 1, therefore this index cannot single-handedly assert the goodness of a partition. On the other hand, when discriminating between two partitions with equal value of a "crisp partition" based measure, the partition with the higher value of V_{PPC} is preferable to the other.

6.2. Probabilistic Modified Partition Coefficient

We see that range of value of V_{PPC} is $[1/g, 1]$, so to remove the dependency of V_{PPC} on the number of components, we make the following transformation and define the new index as the Probabilistic Modified Partition Coefficient (V_{PMPC}), in the following way:

$$V_{PMPC} = 1 - \frac{g}{g-1}(1 - V_{PPC}) = \frac{g}{g-1} V_{PPC} - \frac{1}{g-1}. \tag{5}$$

Table 1
Brief description of "crisp partition" based cluster validity indices.

Performance measure	Functional description	Measured property	Optimal partition
Minkowski score	$MS(A,O)=\sqrt{\frac{a_{01}+a_{10}}{a_{11}+a_{10}}}$	Matching between actual and obtained crisp partition	$\text{argmin}_O\, MS$
Adjusted rand index	$ARI(A,O)=ARI(n_{ij},ni.,n_{.j},n)$	Matching between actual and obtained crisp partition	$\text{argmax}_O\, ARI$

Now, V_{PMPC} is an increasing function of V_{PPC}, so has the same interpretation advantages and short comings of V_{PPC}, but here in this index the dependency on g is removed.

The biggest difference between V_{PC}, V_{MPC} and V_{PPC}, V_{PMPC} proposed here is that, those measures were used as a tool to determine the cluster numbers but here we have incorporated automatic model selection in our algorithm, we do not need to use these measures for selection of cluster number, but we can use them to discriminate between the goodness of two or more T matrices, having the same measure, in terms of a "Crisp Partition" of the data.

We also provide a summary of the existing hard partition based validity indices used in this article for measurement of cluster accuracy in Table 1. For obtaining a so-called "crisp partition" of the data for each of the data points, we look at their respective probability of belonging to each of the clusters and assign it to the cluster for which the probability is the highest. Let $A=\{t_1,t_2,\cdots,t_R\}$ and $O=\{s_1,s_2,\cdots,s_c\}$ be two valid partitions of the given data. Let A be the actual partition and O be the obtained partition. We wish to measure $n_{ij}=$ no. of objects present in both cluster and t_i and s_j, $n_{i.}=$ no. of objects present in cluster t_i, $n_{.j}=$ no. of objects present in cluster s_j; $a_{01}=$ no. of pairs that are in the same cluster only in O, $a_{10}=$ no. of pairs that are in the same cluster only in A, $a_{11}=$ no. of pairs that are in the same cluster in both A and O. Then the ARI can be expressed in the following way:

$$ARI(n_{ij},n_{i.},n_{.j},n)=\frac{\sum_{i,j}\binom{n_{ij}}{2}-\left[\sum_i\binom{n_{i.}}{2}\sum_j\binom{n_{.j}}{2}\right]/\binom{n}{2}}{\frac{1}{2}\left[\sum_i\binom{n_{i.}}{2}+\sum_j\binom{n_{.j}}{2}\right]-\left[\sum_i\binom{n_{i.}}{2}\sum_j\binom{n_{.j}}{2}\right]} \tag{6}$$

7. Experimental results

In this section, we discuss the implementation results of the developed novel algorithms for the purpose of component number selection, feature saliency detection and simultaneous component number and feature selection for fuzzy data. In the following experimental setting the imprecise knowledge about the actual value of a variable is represented as a fuzzy datum. In a synthetic setup, we first simulate crisp data. Then, we transform each of the data point into a fuzzy data point. Here we restrict ourselves to the two most commonly used fuzzy numbers, i.e. the Trapezoidal fuzzy number and the triangular fuzzy number, but our algorithm will remain exactly same for any other fuzzy membership function.

The membership function of a univariate trapezoidal fuzzy number is defined by four scalars a,b,c and d, such that: $(a\leqslant b\leqslant c\leqslant d)$

$$\mu_{\tilde{X}}(x)=\begin{cases}(x-a)/(b-a), & \text{if } a\leqslant x\leqslant b,\\ 1, & \text{if } b\leqslant x\leqslant c,\\ (d-x)/(d-c), & \text{if } c\leqslant x\leqslant d,\\ 0. & \text{Otherwise.}\end{cases}$$

Now, each coordinate x_{ij} of a crisp data point $\mathbf{x}_i$ was transformed into a trapezoidal fuzzy number by the following rule. Four i.i.d observations were drawn from a uniform distribution; then we fix r and s and find out the corresponding a_{ij},b_{ij},c_{ij} and d_{ij}. Let the i.i.d realizations corresponding to x_{ij} be denoted by u_{1ij},u_{2ij},u_{3ij} and u_{4ij}. Then we have:

$$a'_{ij}=u_{1ij}(x_{ij}-s),$$
$$b'_{ij}=u_{2ij}(x_{ij}-r),$$

$$c'_{ij} = u_{3ij}(x_{ij} + r),$$
$$d'_{ij} = u_{4ij}(x_{ij} + s).$$

Now, we order $a'_{ij}, b'_{ij}, c'_{ij}$ and d'_{ij} increasingly to get a_{ij}, b_{ij}, c_{ij} and d_{ij} such that $a_{ij} \leqslant b_{ij} \leqslant c_{ij} \leqslant d_{ij}$.

The membership function of a univariate triangular fuzzy number is defined by the three scalars a, b and c, such that $(a \leqslant b \leqslant c)$:

$$\mu_{\tilde{X}}(x) = \begin{cases} (x-a)/(b-a), & \text{if } a \leqslant x \leqslant b, \\ (c-x)/(c-b), & \text{if } b \leqslant x \leqslant c, \\ 0. & \text{Otherwise.} \end{cases}$$

Each coordinate x_{ij} of a crisp data point $\mathbf{x}_i$ was transformed into a triangular fuzzy number by the following rule. Three i.i.d observations were drawn from a uniform distribution; then we fix s and find out the corresponding a_{ij}, b_{ij} and c_{ij}. Let the i.i.d realizations corresponding to x_{ij} be denoted by u_{1ij}, u_{2ij} and u_{3ij}. Then we have:

$$a'_{ij} = u_{1ij}(x_{ij} - s),$$
$$b'_{ij} = u_{2ij}(x_{ij}),$$
$$c'_{ij} = u_{3ij}(x_{ij} + s).$$

We put a'_{ij}, b'_{ij} and c'_{ij} in an increasing order to get a_{ij}, b_{ij} and c_{ij} such that $a_{ij} \leqslant b_{ij} \leqslant c_{ij}$. Here, we note that a triangular fuzzy number (a, b, d) can be interpreted as a special case of the trapezoidal fuzzy number (a, b, c, d) with $b = c$. Graphically, if the right and left centers of a trapezoidal fuzzy number coincide, then that fuzzy number is a triangular fuzzy number with right and left spread equal to the right and left spread of the trapezoidal fuzzy number and center equal to the common left and right center of the trapezoidal fuzzy number.

Here, we concentrate on the following three novel theoretical developments, presented in the article.

1. Automatic component number selection in the conventional fuzzy EM algorithm.
2. Importance of the introduction of feature selection in presence of the noise variable.
3. Simultaneous component number selection and feature selection.

Our first focus is on model selection. We start with the easiest 2-dimensional setup. Here, we are interested to check the efficacy of the conventional fuzzy EM algorithm with automatic component number selection. The setup is as follows. We generated 100 data points each from two equi-probable 2-dimensional Gaussian distributions; $N(m_i, \mathbf{I}_2),\ i = \{1, 2\}$, where $m_1 = (2, 2)$ and $m_2 = (-2, -2)$. The original crisp data (Crisp Data 1) generated from the GMM is represented in Fig. 1a. We fix $r = 0.5, s = 2.0$ and the obtained trapezoidal fuzzy numbers (Data 1) are plotted in Fig. 2a. The rectangular boxes denote the alpha-cut of the membership function, with $\alpha = 0.75$. The different line style of each rectangle represents the cluster it belongs to. We ran the proposed algorithm 30 times, each time we initialized with component number $g = 6$. The parameters of the 6 components were initialized at random, so that they collectively cover the whole data. The stopping threshold was set to be 10^{-7}. In all the 30 runs of the algorithm, both the components were always correctly detected (Fig. 2b). The change in the corresponding density estimate in a typical run of the conventional fuzzy EM algorithm with automatic component number selection is provided in Fig. 4a–4f.

Next, we focus on the need of feature saliency in the perspective of the presence of noise variables. We start experiments in the easiest 2-dimensional setup (the 1-dimensional setup has no practical significance if we are interested in checking if our algorithm can detect the actual feature and the proper clustering structure in the presence of noise variable(s)). We construct this setup as follows. We generated 100 data points each from two equi-probable 1-dimensional Gaussian distributions; $N(m_i, \frac{1}{4}), i = \{1, 2\}$, where $m_1 = -2$ and $m_2 = 2$. We then simulate data points from a common distribution, $N(-2, \frac{1}{4})$ and append them from the noisy variable with the earlier 1-dimensional data. The original crisp data (Crisp Data 2), generated from the GMM is shown in Fig. 1b. We fix $r = 0.5, s = 2.0$ and the obtained trapezoidal fuzzy numbers (Data 2) are plotted in Fig. 3a. The rectangular boxes denote the alpha-cut of the membership function, with $\alpha = 0.75$. The different line style of each rectangle represents the cluster it belongs to.

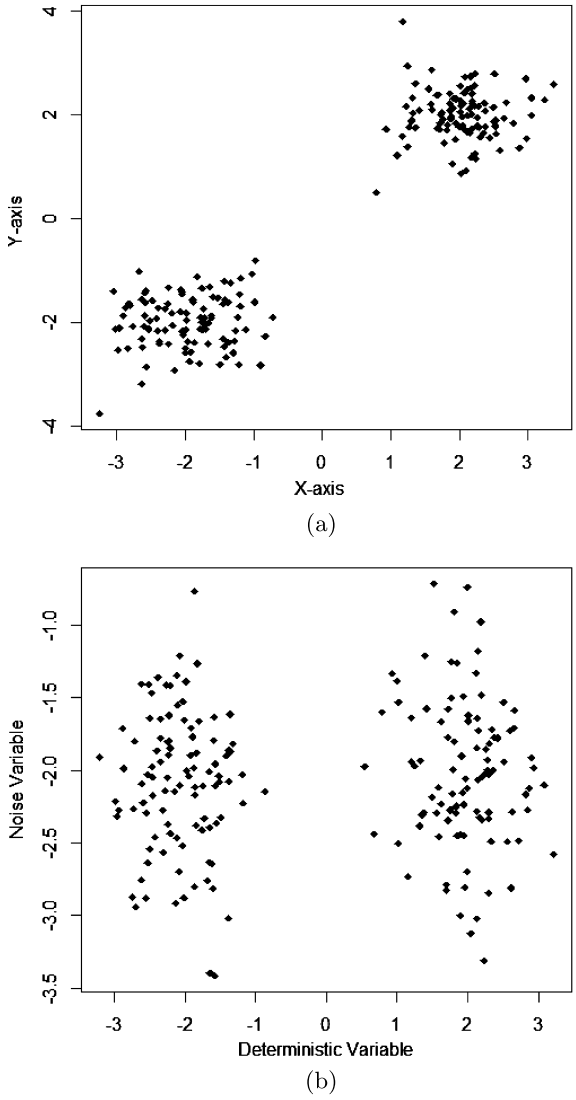
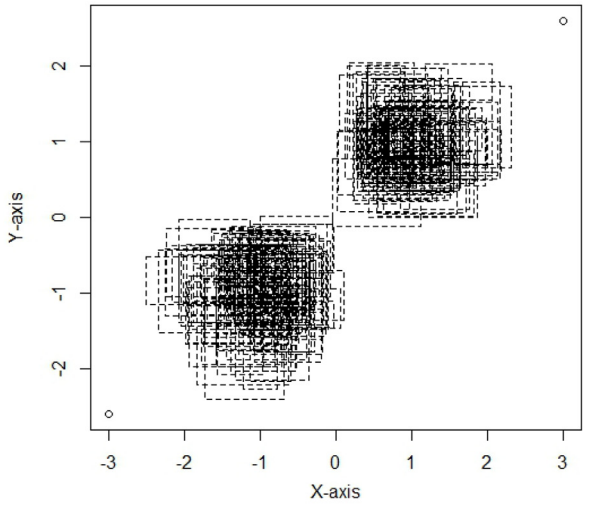


Fig. 1. (a) Crisp Data 1, (b) Crisp Data 2.

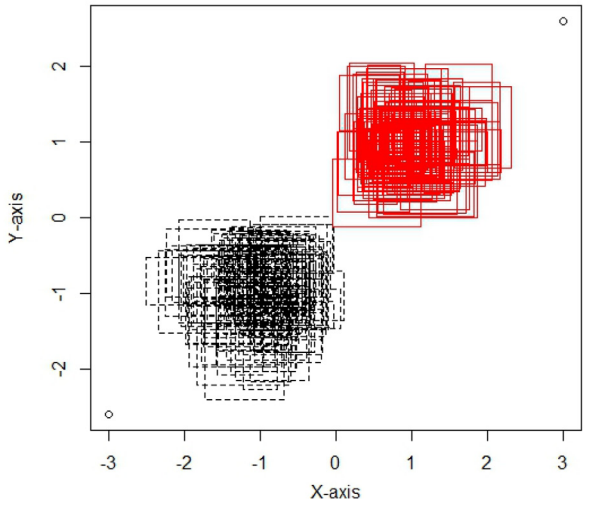
In all the runs of the algorithm, both the components were correctly detected (Fig. 3b). The box plot of feature saliencies of both the features (variables) clearly shows, that fuzzy EM with feature saliency detection was able to detect the noise variable, in all the cases (Fig. 11a).

First, we focus on simultaneous feature saliency detection and component number selection. We perform simultaneous feature and component number selection and clustering on the two sets of data (Data 1, Data 2) developed in the earlier two cases. We ran the proposed algorithm 30 times, each time we initialized with component number $g = 6$. The parameters of the 6 components were initialized at random, so that they collectively cover the whole data. The parameters of the common distribution were initialized so that it covers the whole available data. Both the feature saliency values were set to 0.50 (we avoid any kind of partiality in giving any feature any kind of preference in detection of the cluster structure). The stopping threshold was set to be 10^{-7}.

The algorithm correctly detected the deterministic features and the underlying components correctly in both the cases. The box plot of feature saliencies (Fig. 11b (Data 2) and Fig. 11c (Data 1)) clearly shows that our algorithm was able to differentiate between the deterministic variables and noise variables in all the cases. The change in the corresponding density estimates in a typical run of the algorithm is provided in Figs. 5a–5f (Data 1) and Figs. 8a to 8f (Data 2).



(a)



(b)

Fig. 2. (a) Data 1, (b) Partitioned Data 1.

Next, we focus on the effect of feature saliency detection, hence we do not perform component selection in this case. We ran the proposed algorithm 30 times, each time the initialization of the component parameters corresponding to the conventional fuzzy EM and the conventional fuzzy EM with feature saliency was identical. For the fuzzy EM with feature saliency detection, the parameters of the common distribution were initialized to cover the whole available data. Both the feature saliency values were set to 0.5 (we avoid any kind of partiality in giving any feature any kind of preference in detection of the cluster structure). The stopping threshold was set to be 10^{-7}. The comparative changes in the corresponding density estimate in a typical run of the conventional fuzzy EM algorithm and the fuzzy EM with feature saliency detection is provided in Figs. 6a to 6f (without feature saliency determination) and Figs. 7a to 7f (with feature saliency determination). Fig. 4a–Fig. 4f demonstrate the density estimate in a typical run of the fuzzy algorithm with feature saliency but without component number selection. It clearly demonstrates that our proposed method performs well even without component number selection.

Next, we try to further increase the complexity of the problem (Data 2). We increase the complexity of the problem in two directions,

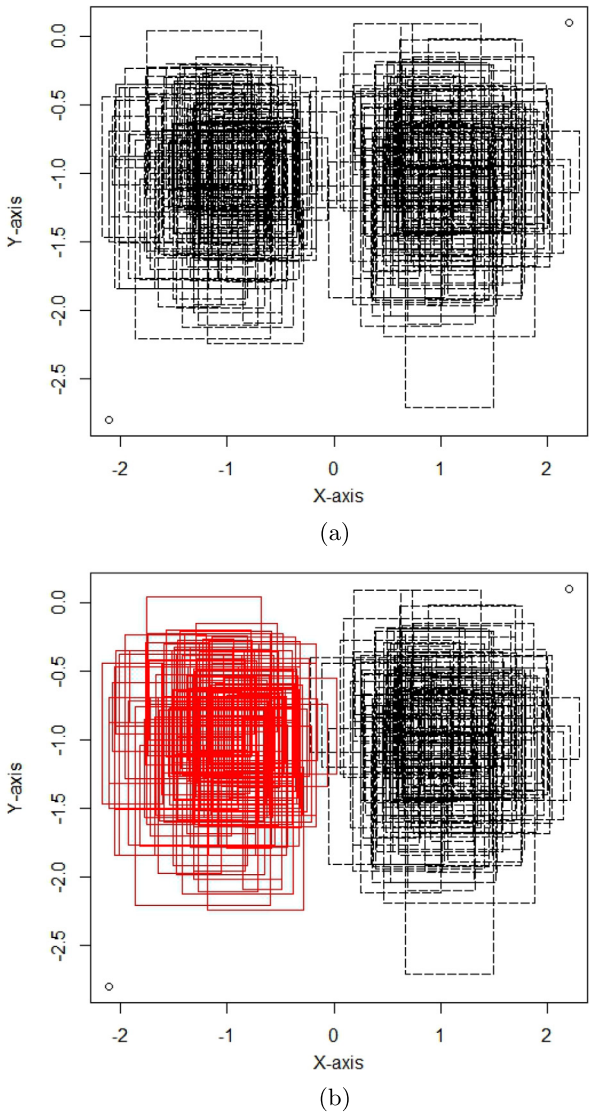


Fig. 3. (a) Data 2, (b) Partitioned Data 2.

1. We increase the number of noise variables.
2. We increase the number of initial components.

We present the clustering result on varied component numbers and noise variables in Tables 2, 3 and 4.

Here we observe that a gradual decrease in average Adjusted Rand Index, Probabilistic Partition Coefficient and Probabilistic Modified Partition Coefficient is observed as we increase the number of initial components and the number of noise variables. Though the average Adjusted Rand Index score is always more than 0.89, which gives a clear indication that the newly developed Fuzzy EM algorithm with automated feature and component selection is successful in finding the inherent cluster structure of the data, even in presence of noise variables.

Next, we test the performance of the presented algorithm on triangular fuzzy numbers. We generated 100 data points each from three equi-probable 1-dimensional Gaussian distributions; $N(m_i, \frac{1}{4}), i = \{1, 2, 3\}$, where $m_1 = -5$, $m_2 = 0$ and $m_3 = 5$. We then simulate 300 data points from a common distribution, $N(0, \frac{1}{4})$ and append them as the noisy variable with the 1-dimensional data generated earlier. The original crisp data (Crisp Data 3) generated from the GMM is represented in Fig. 9a.

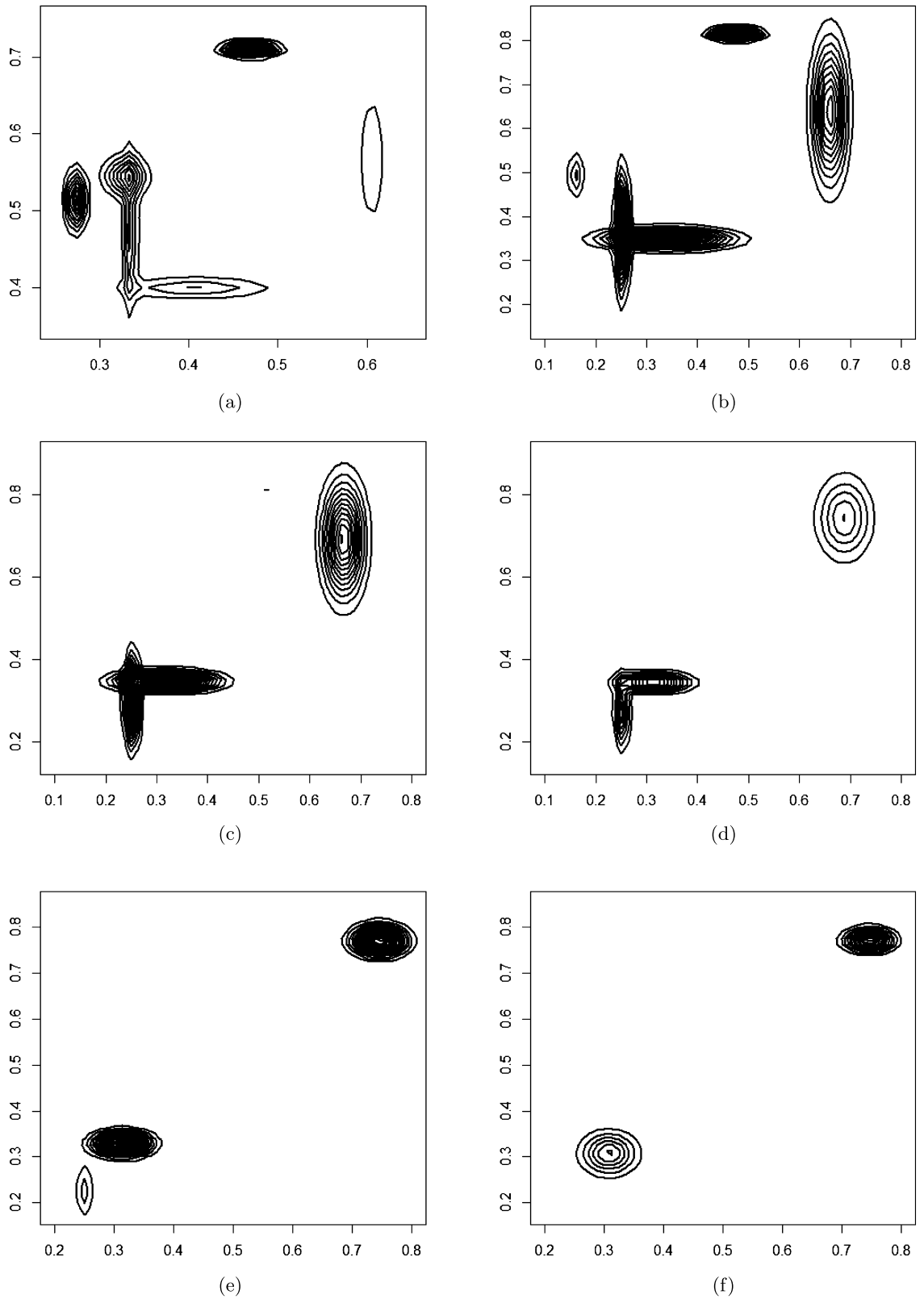


Fig. 4. (a) Initialization, (b) Iteration 1, (c) Iteration 4, (d) Iteration 10, (e) Iteration 50, (f) Iteration 80.

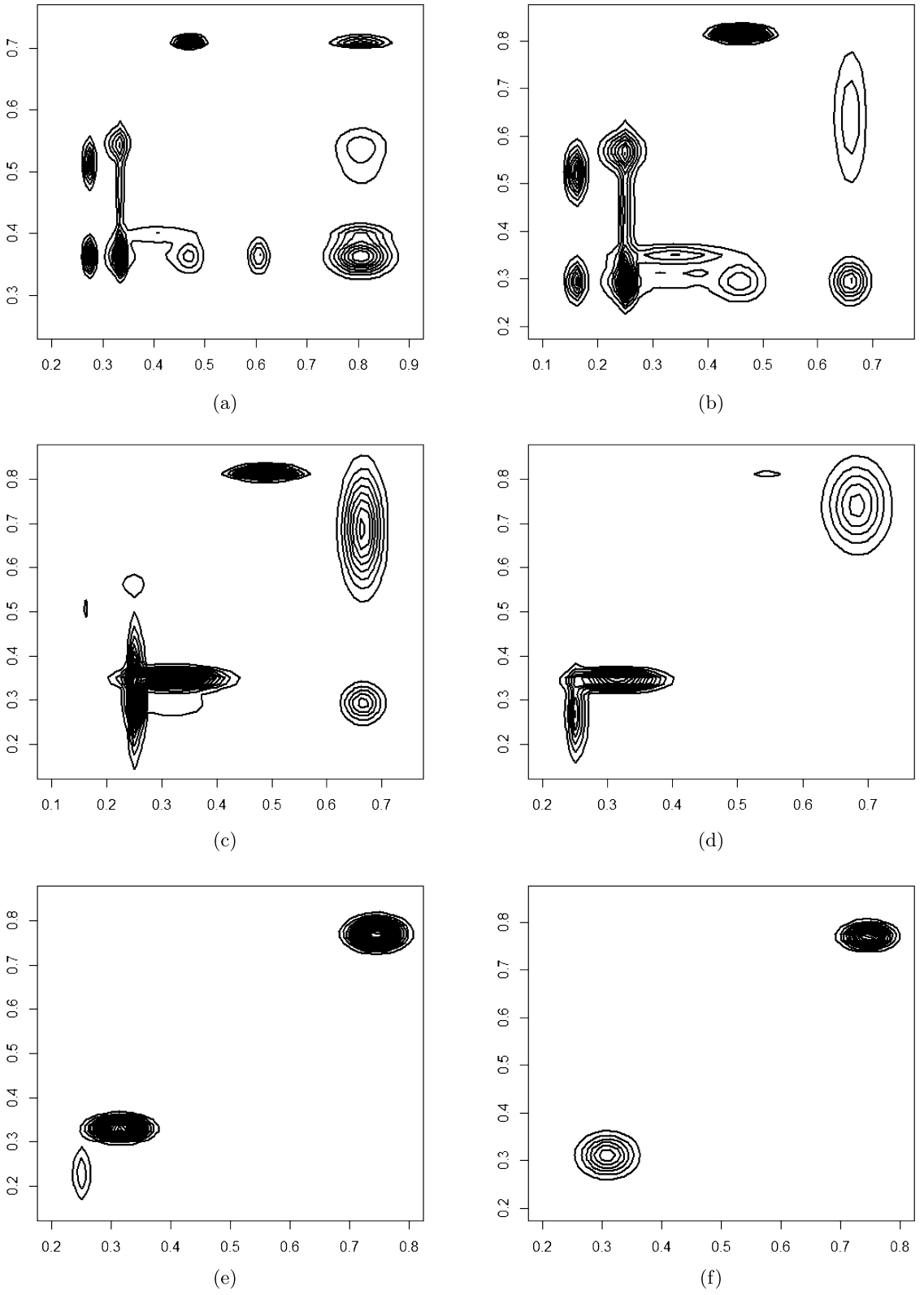


Fig. 5. (a) Initialization, (b) Iteration 1, (c) Iteration 4, (d) Iteration 10, (e) Iteration 50, (f) Iteration 80.

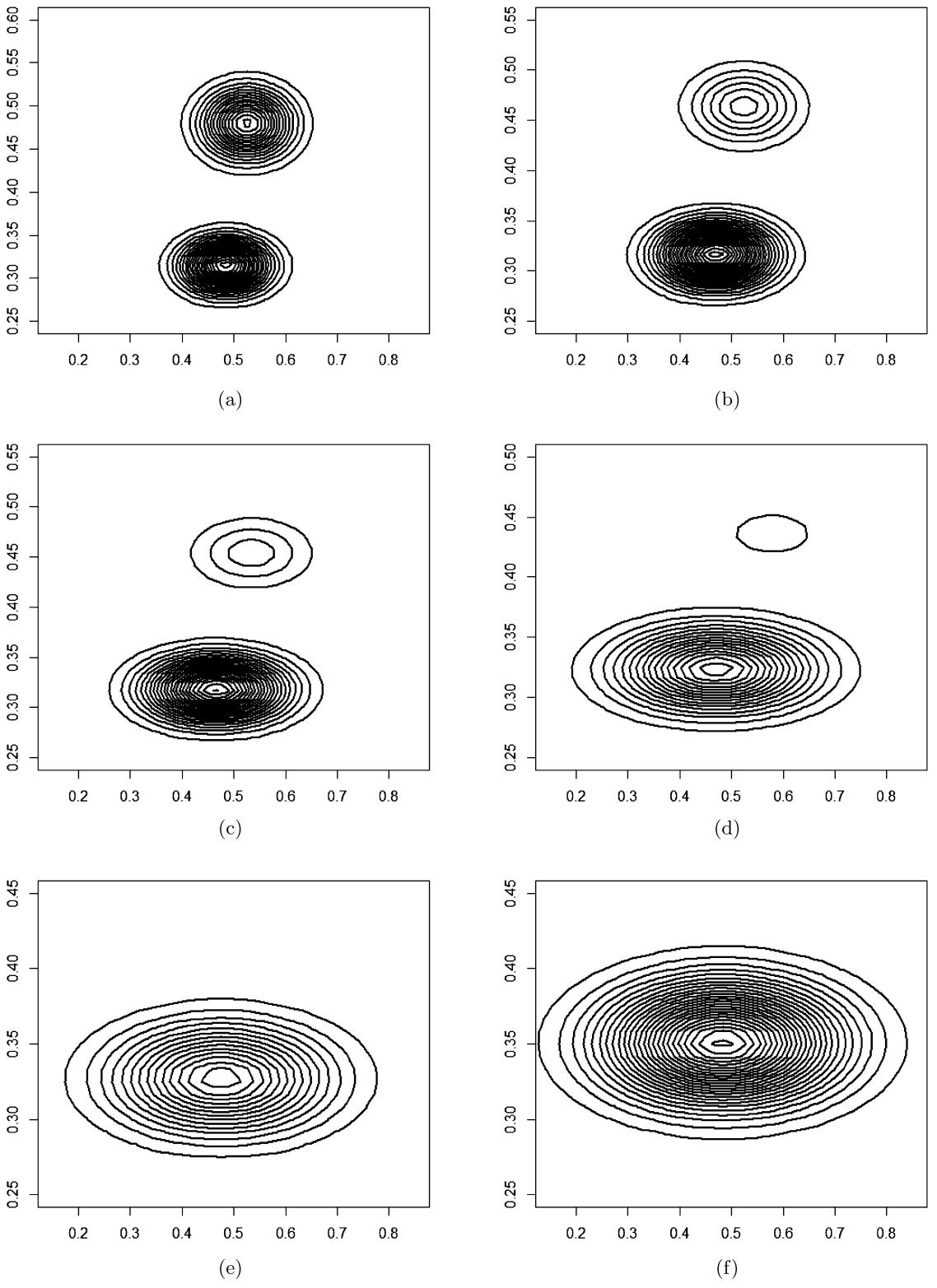


Fig. 6. (a) Initialization, (b) Iteration 1, (c) Iteration 2, (d) Iteration 5, (e) Iteration 7, (f) Iteration 20.

Table 2
ARI values.

No. of initial Components	$d = 2$	$d = 5$	$d = 10$
$g = 6$	0.9755	0.9566	0.9233
$g = 8$	0.9665	0.9433	0.9185
$g = 10$	0.9334	0.9008	0.8905

Table 3
V_{PPC} values.

No. of initial Components	$d = 2$	$d = 5$	$d = 10$
$g = 6$	0.9975	0.9861	0.9618
$g = 8$	0.9753	0.9677	0.9511
$g = 10$	0.9696	0.9422	0.9181

Table 4
V_{PMPC} values.

No. of initial Components	$d = 2$	$d = 5$	$d = 10$
$g = 6$	0.9972	0.9806	0.9332
$g = 8$	0.9514	0.9487	0.9242
$g = 10$	0.9381	0.9137	0.9006

We fix $s = 2.0$ and the obtained trapezoidal fuzzy numbers (Data 3) are plotted in Fig. 9b and Fig. 9c respectively. The rectangular boxes denote the alpha-cut of the membership function, with $\alpha = 0.75$. The different line style of each rectangle represents the cluster it belongs to.

We focus on simultaneous feature saliency detection and component number selection in the perspective of triangular fuzzy numbers. We perform simultaneous feature and component number selection and clustering on the triangular fuzzy data under consideration. We ran the proposed algorithm 30 times, each time we initialized with component number $g = 9$. The parameters of the 9 components were initialized at random, so that they collectively cover the whole data. The parameters of the common distribution were initialized so that it covers the whole available data. Both the feature saliency values were set to 0.50 (we avoid any kind of partiality in giving any feature any kind of preference in detection of the cluster structure). The stopping threshold was set to be 10^{-7}. The algorithm correctly detected the deterministic features and the underlying components correctly in both cases. The box plot of the feature saliencies of the both the features (variables) clearly shows that our algorithm was able to differentiate between the deterministic variables and noise variables in all the cases (Fig. 11d). The change in the corresponding density estimates in a typical run of the algorithm is provided in Figs. 9d to 9i.

Next, we focus on simultaneous feature saliency detection and component number selection from the perspective of real word fuzzy data. We perform simultaneous feature and component number selection and clustering on the Blood Pressure data [27].

We ran the Fuzzy EM algorithm with automated component number and feature selection on the blood data used in [27]. This dataset presents statistics on daily measurements of the systolic and diastolic pressures on patients. Here, each measurement is precise; however, for each patient, center and spread values only were stored. Thus, the fuzziness of a datum stems from the variability of the measurements performed on each patient and the choice to summarize these measurements using their center and their spread only. Although this interpretation differs from the point of view followed in this paper, we used these data to compare our results with those presented in [27]. We interpreted these data as triangular fuzzy numbers, which are a special case for trapezoidal fuzzy numbers. In addition, we assumed that each center was equidistant to the minimal and maximal values. First, the data were cantered and scaled with respect to the mean and standard deviation of the center values. The centers of the fuzzy data form a crisp dataset, which is presented in Fig. 10a. The obtained fuzzy data and the partitioned data are presented in Figs. 10b–10c.

We perform simultaneous feature and component number selection and clustering on the triangular fuzzy data under consideration. We ran the proposed algorithm 30 times, each time we initialized with component number $g = 4$. The parameters of the 4 components were initialized at random, so that they collectively cover the whole data. The

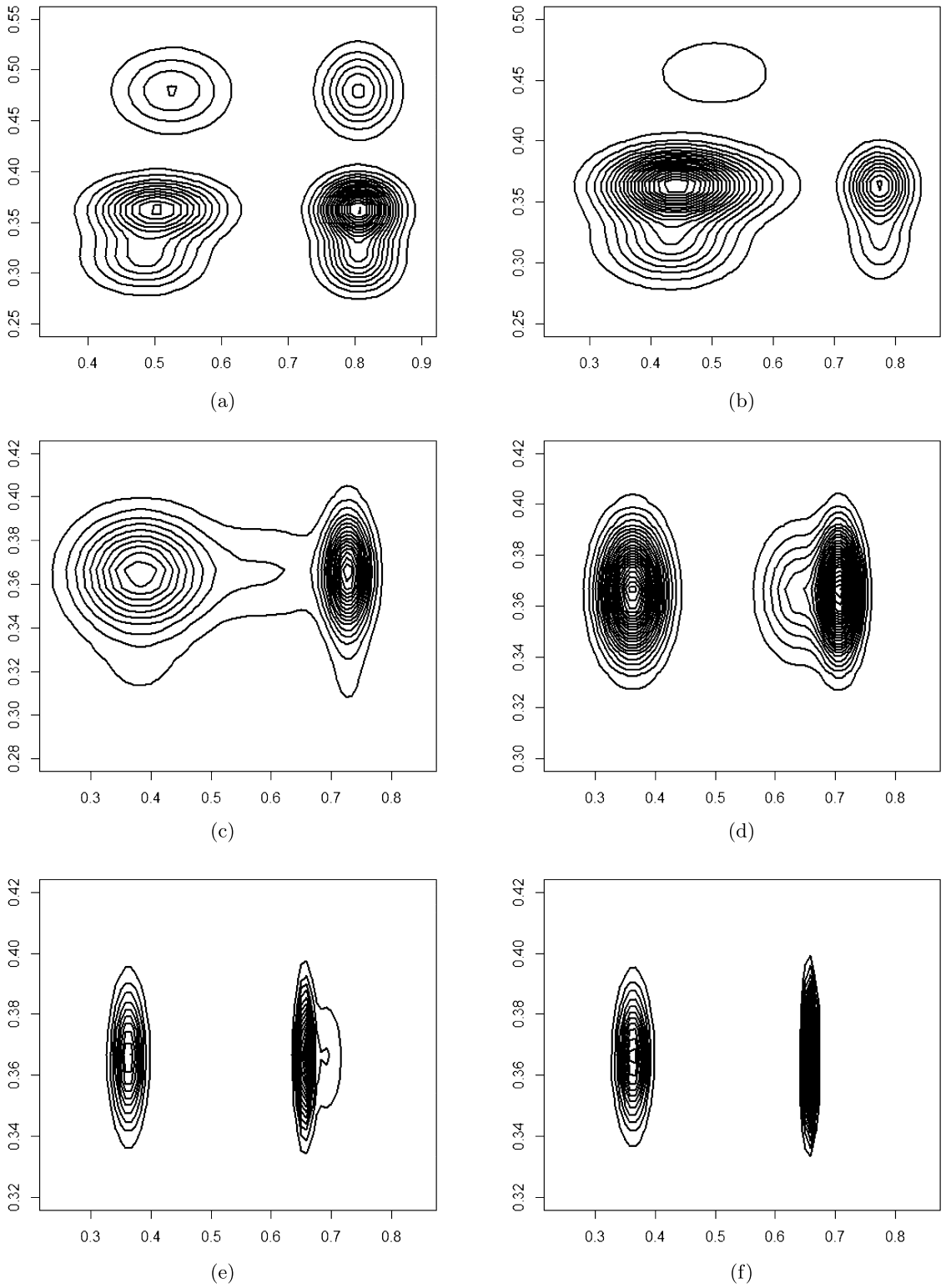


Fig. 7. (a) Initialization, (b) Iteration 2, (c) Iteration 10, (d) Iteration 20, (e) Iteration 100, (f) Iteration 500.

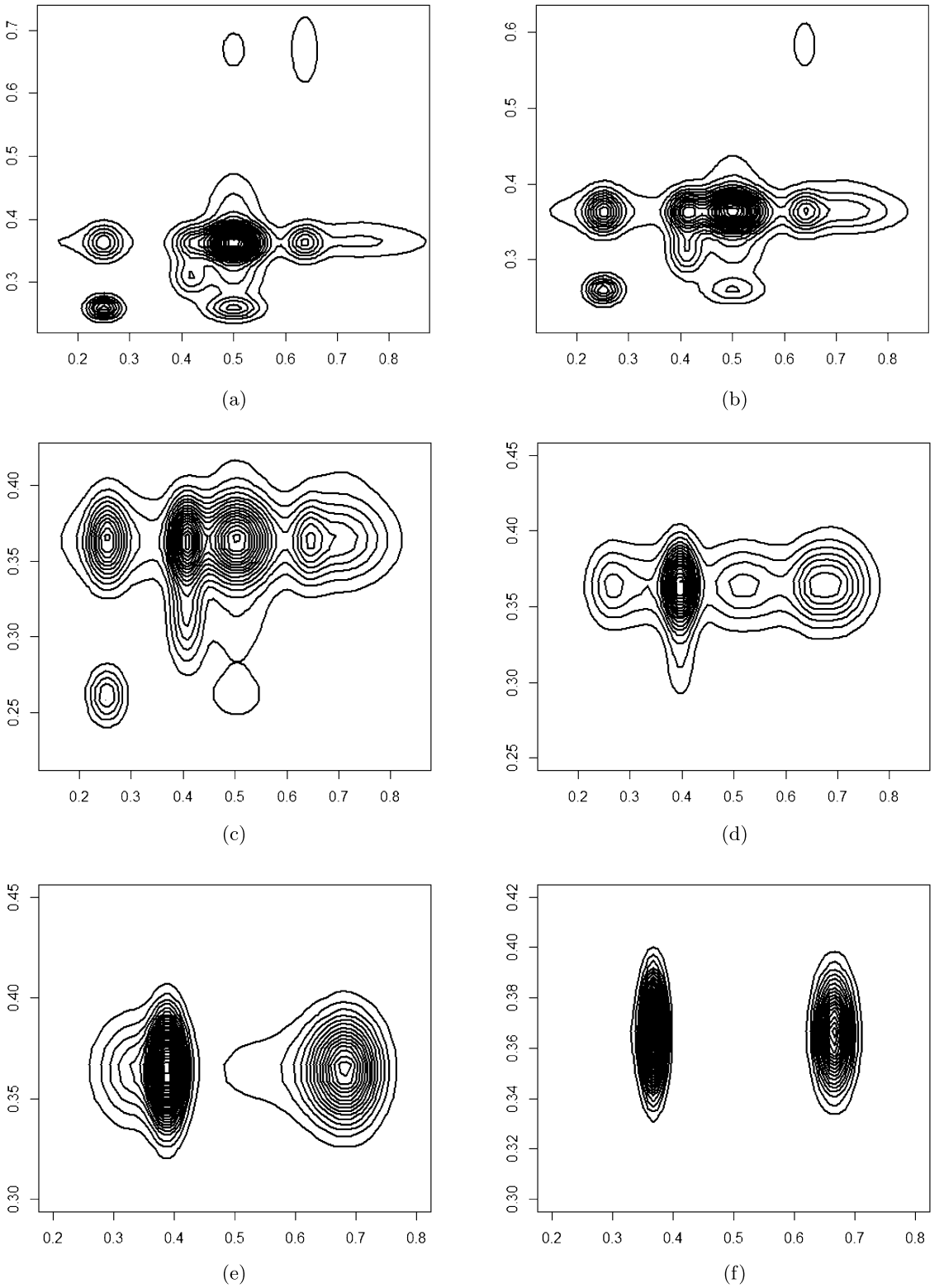


Fig. 8. (a) Initialization, (b) Iteration 1, (c) Iteration 2, (d) Iteration 5, (e) Iteration 8, (f) Iteration 120.

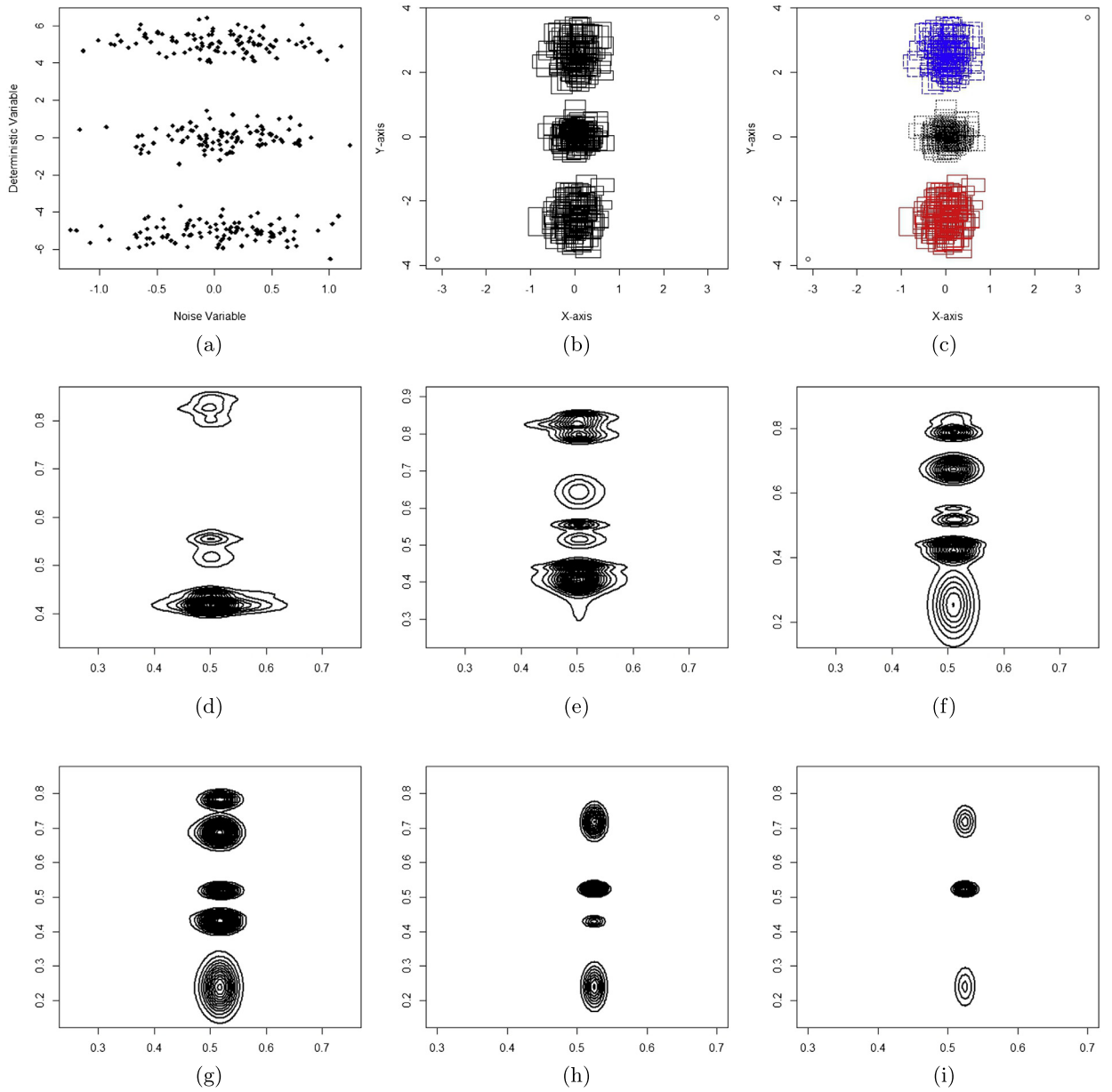


Fig. 9. (a) Crisp Data 3, (b) Data 3, (c) Partitioned Data 3, (d) Initialization, (e) Iteration 1, (f) Iteration 5, (g) Iteration 10, (h) Iteration 70, (i) Iteration 170.

parameters of the common distribution were initialized so that it covers the whole available data. Both the feature saliency values were set to 0.5 (we avoid any kind of partiality in giving any feature any kind of preference in detection of the cluster structure). The stopping threshold was set to be 10^{-7}.

The algorithm correctly detected both the features as the deterministic features as indicated by existing literature on the data [27]. The change in the corresponding density estimates in a typical run of the algorithm is provided in Figs. 10d to 10i.

Here, by observing the general trend, we see that there are two predominant classes in the data. The group of persons characterized by low values of Systolic Pressure in terms of both center and spread and high values of Diastolic Pressure in terms of both centers and spreads. The second cluster can be described just in the opposite way. In the

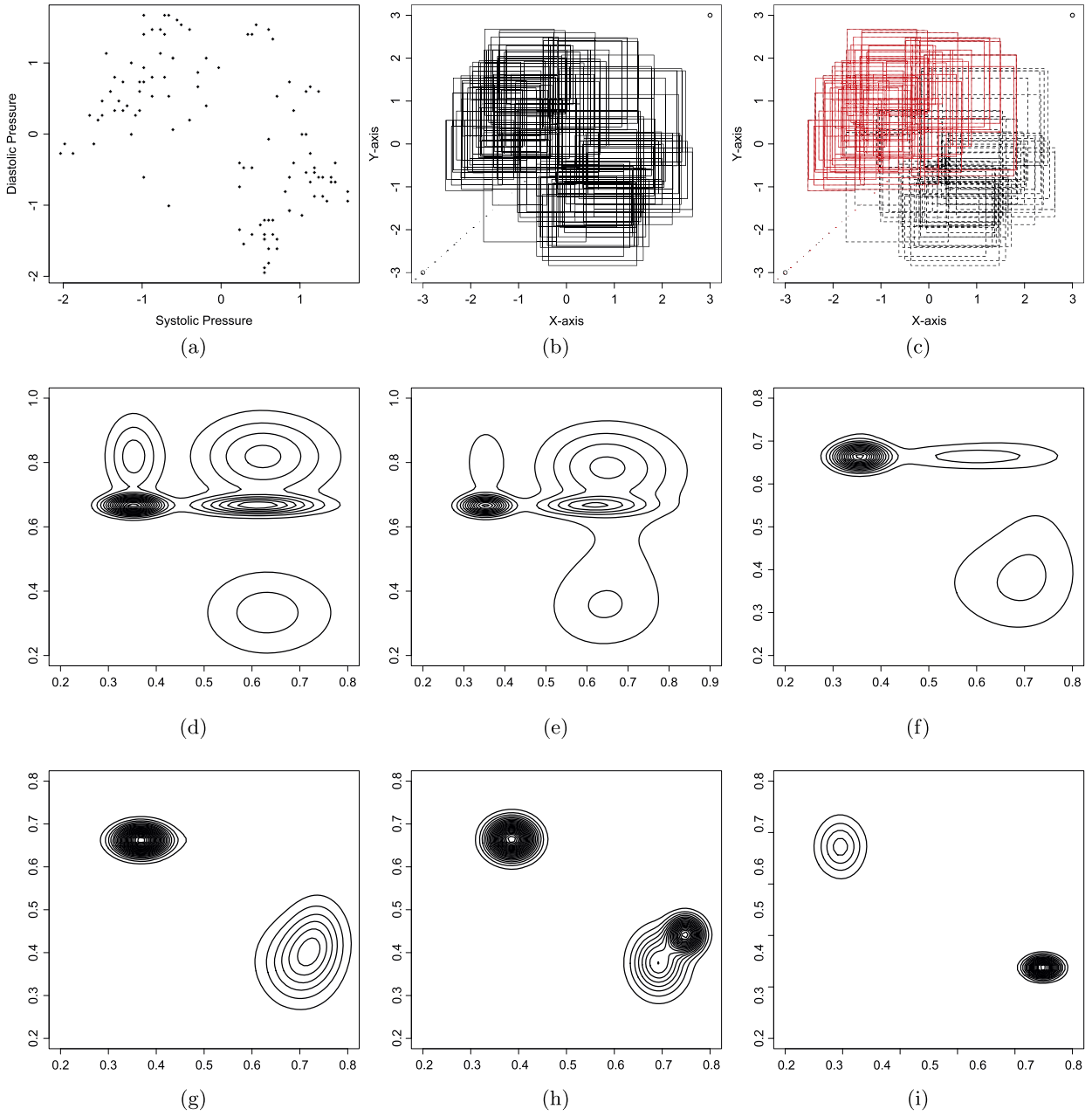


Fig. 10. (a) Crisp Blood-Pressure Data, (b) Fuzzy Blood-Pressure Data, (c) Partitioned Blood-Pressure Data, (d) Initialization, (e) Iteration 1, (f) Iteration 6, (g) Iteration 20, (h) Iteration 50, (i) Iteration 100.

clustering process 54 data points were assigned to the first cluster and 54 points were assigned to the second cluster. Hence, the obtained clusters are strongly consistent with the raw data.

7.1. High dimensional data

We have presented the clustering result on varied component number and noise variable in Tables 2, 3 and 4, where we deal with a 10-dimensional dataset. We further notice that even if we further increase the number of noise attributes ($d = 50$), the average ARI remains $\geqslant 0.8$ with initial component number 6. We also observe that in most of the cases,

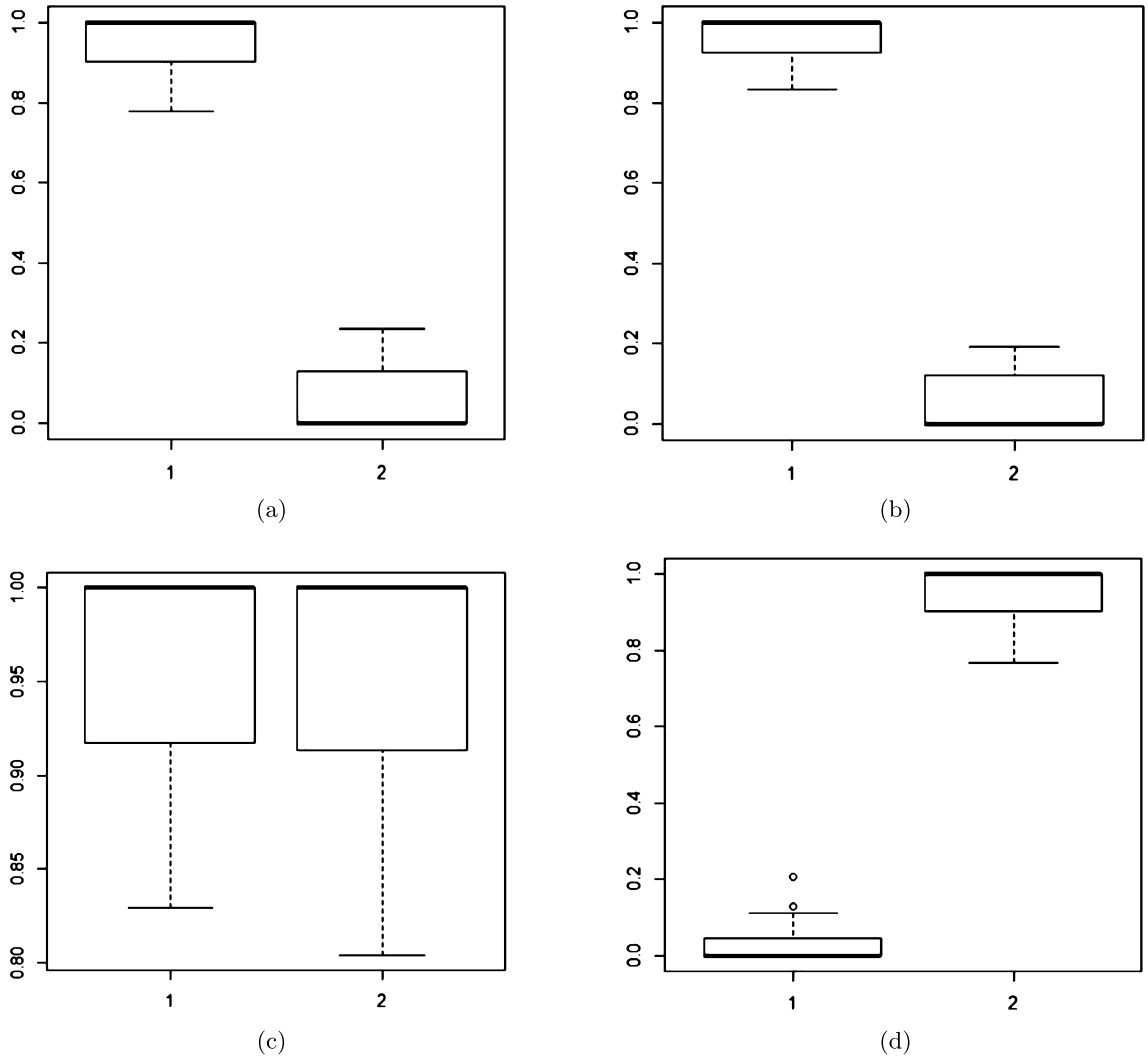


Fig. 11. Box Plot of obtained feature saliencies corresponding to (a) Data 2 with 2 initial components, (b) Data 2 with 6 initial components, (c) Data 1, (d) Data 3.

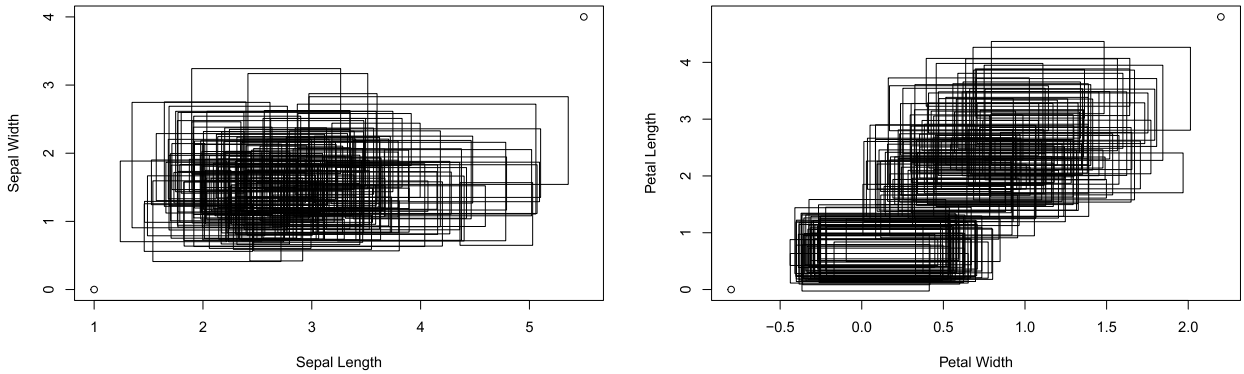
the proposed algorithm could determine the right feature saliency (i.e. 1 for the deterministic feature and 0 for the noise features).

7.2. Iris data

We also carried out a comparison of the clustering performance of the fuzzy EM [15] and the proposed algorithm on the fuzzified version of Iris Data. We chose this data for two specific reasons:

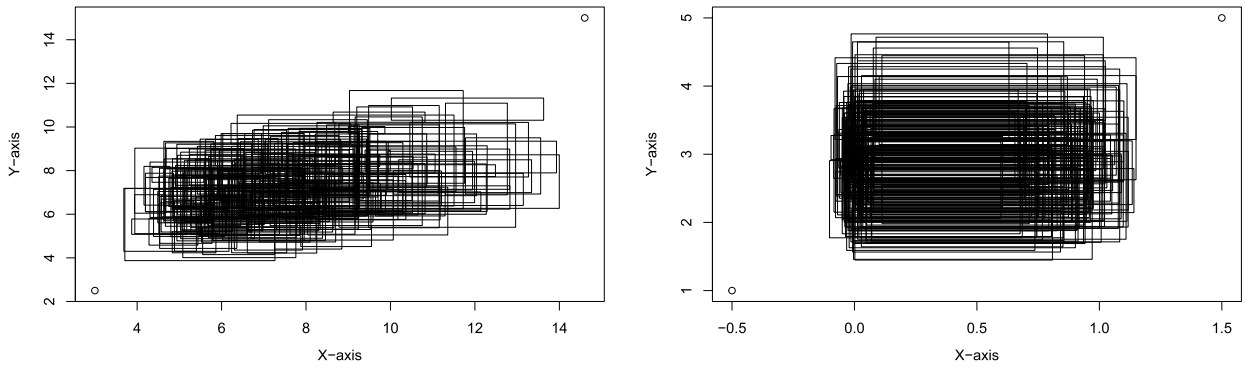
- The presence of noise features.
- The presence of overlapping clusters.

We fuzzified the Iris data into trapezoidal fuzzy data using the previously used values of $r = 0.5$ and $s = 2.0$. We plot the generated fuzzy data and quite expectedly observe that Petal length and Petal width are two discriminating variables. On the other hand the presence of Sepal length and Sepal Width as non-discriminating features will worsen the clustering performance (Only 2 2-dimensional plots [Figs. 12a to 12b] are presented here, where we demonstrate



(a) (b)

Fig. 12. Fuzzy iris data corresponding to (a) Sepal Length and Sepal Width, (b) Petal length and Petal width.



(a) (b)

Fig. 13. Fuzzy seed data corresponding to (a) dimensions 1 and 2, (b) dimensions 3 and 4.

Table 5
ARI values.

Dataset	ARI
Iris	0.951
Seed	0.843

the prominent cluster structure with respect to the Petal length and Petal width and the apparent absence of that with respect to the Sepal Length and Sepal Width). Our proposed algorithm could identify the noise features and their feature saliencies were driven towards zero, whereas the feature saliencies of the deterministic features were driven towards 1.

7.3. Seed data

We also test our data on seeds data where there is no known existence of noise features. We fuzzify and generate trapezoidal fuzzy numbers from the crisp data with $r = 0.5$ and $s = 2.0$. We plot the generated fuzzy numbers corresponding to the first four dimensions of the data (Figs. 13a to 13b).

We also present the corresponding ARI in Table 5.

8. Conclusion

In this article, we have introduced the concept of model selection in the perspective of the fuzzy EM algorithm, which performs simultaneous cluster number determination and clustering of the fuzzy data. We have proposed several methods of model selection in this aspect. Next, we have introduced the concept of feature saliency for fuzzy data and discussed its importance. We have proposed a novel clustering algorithm for the clustering of fuzzy data with automated component number and feature selection. This mixture model based clustering algorithm performs automatic component number selection and feature saliency determination, which eventually leads to feature selection. With the help of various experimental results on synthetic and real world data, we have demonstrated that the novel developments of the fuzzy EM algorithm proposed in the paper is able to recover the inherent cluster structure of the data even in presence of noise variables, when the number of actual clusters is not known beforehand. The experimental results have also indicated the robustness of the proposed framework. The novelty of our paper lies in the introduction of the concept of automated component number selection, feature saliency and feature selection in the perspective of fuzzy data and incorporating those concepts into a single and conventional mixture model based fuzzy EM algorithm.

An obvious future extension may include comparison of the practical performance of several model selection criteria in the perspective of fuzzy data hence leading to a better mixture model based clustering algorithm for fuzzy data. Another path of the future extension of the article is extending the newly developed concepts of feature saliency to the classification of fuzzy data, when class labels for some of the examples are available. Furthermore, additional constraints can be augmented into the optimization schemes, thus leading to the constrained clustering of fuzzy data, where some of the data elements may have partial information on belonging to a cluster together or never belonging to the same cluster (in a semi-supervised learning framework).

Appendix A

Optimization with respect to the λ_k's

$$\frac{\partial}{\partial \lambda_k} F_1(\boldsymbol{\lambda}) = \frac{\partial}{\partial \lambda_k} F_1 \log(\Gamma(\epsilon\lambda_k)) - \epsilon \log \pi_k,$$

$$\frac{\partial^2}{\partial \lambda_k^2} F_1(\boldsymbol{\lambda}) = \frac{\partial^2}{\partial \lambda_k^2} F_1 \log(\Gamma(\epsilon\lambda_k)) = \sum_{n=0}^{\infty} \frac{\epsilon^2}{(\epsilon\lambda_k + n)^2} > 0.$$

Hence, the objective function $F_1(\boldsymbol{\lambda})$ is convex with respect to λ_k.

Appendix B

Optimization with respect to the ϵ

$$\frac{\partial^2}{\partial \epsilon^2} F_2(\epsilon) = \sum_{n=0}^{\infty} \sum_{k=1}^{g} \frac{\lambda_k^2}{(\epsilon\lambda_k + n)^2} - \sum_{n=0}^{\infty} \frac{1}{(\epsilon + n)^2} > 0.$$

Hence, the objective function $F_2(\epsilon)$ is strictly convex with respect to ϵ.

$$\begin{aligned}
\frac{\partial}{\partial \epsilon} F_2(\epsilon) &= \frac{\partial}{\partial \epsilon} \log \Gamma(\epsilon) + \sum_{k=1}^{g} \frac{\partial}{\partial \epsilon} \log(\Gamma(\epsilon\lambda_k)) - \sum_{k=1}^{g} \lambda_k \log \pi_k \\
&= \sum_{n=0}^{\infty} \left(\frac{1}{\epsilon + n} - \frac{1}{n+1}\right) - \sum_{k=1}^{g} \sum_{n=0}^{\infty} \left(\frac{1}{\epsilon\lambda_k + n} - \frac{1}{n+1}\right) - \sum_{k=1}^{g} \lambda_k \log \pi_k \\
&= \sum_{n=0}^{\infty} \frac{1}{\epsilon + n} - \sum_{k=1}^{g} \sum_{n=0}^{\infty} \frac{1}{\epsilon\lambda_k + n} - \sum_{k=1}^{g} \lambda_k \log \pi_k
\end{aligned}$$

Hence,

$$\lim_{\epsilon\to\infty}\frac{\partial}{\partial\epsilon}F_2(\epsilon) = -\sum_{k=1}^{g}\lambda_k\log\pi_k > 0$$

After some significantly high value of $\epsilon = \epsilon_{max}$, $\frac{\partial}{\partial\epsilon}F_2(\epsilon) > 0$, hence, for minimization we concentrate on $\epsilon \leqslant \epsilon_{max}$. So, the feasible set can be considered as a compact set.

Appendix C

Optimization with respect to the τ

$$\frac{\partial}{\partial\tau}F_3(\tau) = \sum_{k=1}^{g}\sum_{l=1}^{d}(\log\sigma_{kl} - \log\nu_{kl}) + gd\left[\sum_{n=0}^{\infty}\left[\frac{0.5}{\frac{\tau}{2}+n} - \frac{1}{n+1}\right] + \frac{\log 2}{2}\right].$$

After some significantly high value of $\tau = \tau_{max}$, $\frac{\partial}{\partial\tau}F_3(\tau) > 0$, hence, for minimization, we concentrate on $\tau \leqslant \tau_{max}$. So, the feasible set can be considered as a compact set.

References

[1] J. Gebhardt, M.A. Gil, R. Kruse, Fuzzy set-theoretic methods in statistics, in: Fuzzy Sets in Decision Analysis, Operations Research and Statistics, Springer, 1998, pp. 311–347.
[2] M.A. Gil, M. López-Díaz, D.A. Ralescu, Overview on the development of fuzzy random variables, Fuzzy Sets Syst. 157 (19) (2006) 2546–2557.
[3] A. Colubi, Statistical inference about the means of fuzzy random variables: applications to the analysis of fuzzy- and real-valued data, Fuzzy Sets Syst. 160 (3) (2009) 344–356.
[4] G. González-Rodríguez, M. Montenegro, A. Colubi, M.Á. Gil, Bootstrap techniques and fuzzy random variables: synergy in hypothesis testing with fuzzy data, Fuzzy Sets Syst. 157 (19) (2006) 2608–2613.
[5] T. Denœux, Maximum likelihood estimation from fuzzy data using the EM algorithm, Fuzzy Sets Syst. 183 (1) (2011) 72–91.
[6] M.A. Figueiredo, A.K. Jain, Unsupervised learning of finite mixture models, IEEE Trans. Pattern Anal. Mach. Intell. 24 (3) (2002) 381–396.
[7] P. D'Urso, P. Giordani, A weighted fuzzy c-means clustering model for fuzzy data, Comput. Stat. Data Anal. 50 (6) (2006) 1496–1523.
[8] G. González-Rodríguez, A. Colubi, P. D'Urso, M. Montenegro, Multi-sample test-based clustering for fuzzy random variables, Int. J. Approx. Reason. 50 (5) (2009) 721–731.
[9] P. D'Urso, L. De Giovanni, Midpoint radius self-organizing maps for interval-valued data with telecommunications application, Appl. Soft Comput. 11 (5) (2011) 3877–3886.
[10] R. Coppi, P. D'Urso, P. Giordani, Fuzzy and possibilistic clustering for fuzzy data, Comput. Stat. Data Anal. 56 (4) (2012) 915–927.
[11] P. D'Urso, L. De Giovanni, Robust clustering of imprecise data, Chemom. Intell. Lab. Syst. 136 (2014) 58–80.
[12] P. D'Urso, L. De Giovanni, R. Massari, Self-organizing maps for imprecise data, Fuzzy Sets Syst. 237 (2014) 63–89.
[13] J.M. Leski, et al., Fuzzy c-ordered medoids clustering for interval-valued data, Pattern Recognit. 58 (2016) 49–67.
[14] P. D'Urso, R. Massari, L. De Giovanni, C. Cappelli, Exponential distance-based fuzzy clustering for interval-valued data, Fuzzy Optim. Decis. Mak. 16 (1) (2017) 51–70.
[15] B. Quost, T. Denoeux, Clustering and classification of fuzzy data using the fuzzy em algorithm, Fuzzy Sets Syst. 286 (2016) 134–156.
[16] C.S. Wallace, D.L. Dowe, Minimum message length and Kolmogorov complexity, Comput. J. 42 (4) (1999) 270–283.
[17] M.H. Law, M.A. Figueiredo, A.K. Jain, Simultaneous feature selection and clustering using mixture models, IEEE Trans. Pattern Anal. Mach. Intell. 26 (9) (2004) 1154–1166.
[18] H.G. Matthies, Quantifying uncertainty: modern computational representation of probability and applications, in: Extreme Man-Made and Natural Hazards in Dynamics of Structures, Springer, 2007, pp. 105–135.
[19] K.L. Nylund, T. Asparouhov, B.O. Muthén, Deciding on the number of classes in latent class analysis and growth mixture modeling: a Monte Carlo simulation study, Struct. Equ. Model. 14 (4) (2007) 535–569.
[20] F. Sha, L.K. Saul, Large margin hidden Markov models for automatic speech recognition, in: Advances in Neural Information Processing Systems, 2006, pp. 1249–1256.
[21] J.J. Oliver, R.A. Baxter, C.S. Wallace, Unsupervised learning using mml, in: ICML, 1996, pp. 364–372.
[22] G. McLachlan, D. Peel, Finite Mixture Models, John Wiley & Sons, 2004.
[23] L. Shi, S. Tu, L. Xu, Learning Gaussian mixture with automatic model selection: a comparative study on three Bayesian related approaches, Front. Electr. Electron. Eng. China 6 (2) (2011) 215–244.
[24] J. Rissanen, Stochastic Complexity in Statistical Inquiry, World Scientific, 1989.
[25] Y. Nesterov, A method of solving a convex programming problem with convergence rate $o(1/k^2)$, in: Soviet Mathematics Doklady, vol. 27, 1983, pp. 372–376.
[26] D.M. Titterington, Statistical Analysis of Finite Mixture Distributions, Ph.D. thesis, Institute of Philosophy, 2005.

[27] B. Quost, T. Denœux, Clustering fuzzy data using the fuzzy EM algorithm, in: Scalable Uncertainty Management, Springer, 2010, pp. 333–346.
[28] W.M. Rand, Objective criteria for the evaluation of clustering methods, J. Am. Stat. Assoc. 66 (336) (1971) 846–850.
[29] N. Jardine, R. Sibson, Mathematical Taxonomy, John Wiley, London, 1971.
[30] L. Hubert, P. Arabie, Comparing partitions, J. Classif. 2 (1) (1985) 193–218.
[31] J.C. Bezdek, Cluster validity with fuzzy sets, J. Cybern. 3 (3) (1973) 58–73, https://doi.org/10.1080/01969727308546047.
[32] J.C. Bezdek, J.C. Dunn, Optimal fuzzy partitions: a heuristic for estimating the parameters in a mixture of normal distributions, IEEE Trans. Comput. 100 (8) (1975) 835–838.
[33] R.N. Dave, Validating fuzzy partitions obtained through c-shells clustering, Pattern Recognit. Lett. 17 (6) (1996) 613–623.