Optimization of Fuzzy C-Means with Alternating Direction Method of Multipliers

Benoit Albert, Violaine Antoine, Jonas Koko

Optimization of Fuzzy C-Means with Alternating Direction Method of Multipliers

Benoit Albert, Violaine Antoine, and Jonas Koko

Université Clermont Auvergne, CNRS UMR 6158, Mines de Saint-Etienne, Clermont-Auvergne-INP, LIMOS, 63000 Clermont Ferrand, France
{benoit.albert,violaine.antoine,jonas.koko}@uca.fr

Abstract. Among the clustering methods, K-Means and its variants are very popular. These methods solve at each iteration the first-order optimality conditions. However, in some cases, the function to be minimized is not convex, as for the Fuzzy C-Means version with Mahalanobis distance (FCM-GK). In this study, we apply the Alternating Directions Method of Multiplier (ADMM) to ensure a good convergence. ADMM is often applied to solve a separable convex minimization problem with linear constraints. ADMM is a decomposition/coordination method with a coordination step provided by Lagrange multipliers. By appropriately introducing auxiliary variables, this method allows the problem to be decomposed into easily solvable convex subproblems while keeping the same iterative structure. Numerical results have demonstrated the significant performance of the proposed method compared to the standard method, especially for high-dimensional data.

Keywords: Clustering · FCM · Mahalanobis distance · Optimization · ADMM

1 Introduction

Clustering is a data analysis process that consists to split n objects of the dataset into c subsets, with the idea that each group (subset) has similar objects and that the subsets are quite distinguishable from each other [15]. It allows the detection of hidden structures in data sets without prior knowledge. Several different approaches exist, the methods are distinguished by the nature of the partitions created. Among the models using centroids to represent clusters, there is a variant of K-Means called Fuzzy C-Mean (FCM) [2,3] which allows to take into account the uncertainty. This method creates a fuzzy partition that model the degree to which each object belongs to each cluster. It is still used in various fields such as bioinformatics [1] and image analysis [5,21]. The similarity between objects and centroids in the FCM algorithm is calculated with the Euclidean distance. The algorithm of Gustafson and Kessel FCM-GK [13] is an extension of FCM that adjusts an adaptive distance for each cluster. It allows us to take into account the shape of the clusters, to detect not only spherical structures but also ellipsoidal structures. Indeed, based on the Mahalanobis distance, the

algorithm adapts symmetric positive definite matrices interpreted as the inverse of the fuzzy covariance matrices of clusters. FCM and GK are two non-convex optimization problems under constraints for which the standard optimization method is the alternating optimization (AO) method, an iterative method of the Gauss-Seidel type.

The Alternating Direction Method of Multiplier (ADMM) is a simple but powerful decomposition-coordination method. It decomposes the problem into sub-problems, and the solutions obtained locally are coordinated by Lagrange multipliers to find a solution for the global problem. This method was introduced in the mid-1970s for the numerical approximation of non-smooth convex problems from mechanics [9,12]. This method has been used in many fields, first of all in nonlinear mechanics [8,11,12,16], also in image restoration [17], in neural networks [7], in large scale optimization [6,16], etc. A summary of ADMM applications in machine learning is available in [4]. The standard ADMM focuses on the minimization of separable (convex) functions with linear coupling constraints.

In this paper, we extend the application of ADMM to the non-convex cost function of FCM-GK. ADMM divides the FCM-GK problem into a sequence of simpler, uncoupled subproblems, through the appropriate introduction of unknown auxiliary variables. The solution formulation is close to the one obtained by alternating optimization for the original variables (centroids, distance-related membership matrices). The auxiliary variables sub-problem leads to the solution of small uncoupled linear systems. Numerical experiments on UCI machine learning data show that the proposed FCM-ADMM algorithm is robust, insensitive to random initialization, and generally creates better partitioning.

The paper is organized into four sections. Section 2 presents the GK model and the standard method optimization (AO). Then, in Section 3, we describe the application of the ADMM method in this context. In Section 4, the numerical experiments are presented. Finally, the conclusion and perspectives are given in Section 5.

2 FCM-GK model

2.1 Optimisation problem

Let the data set represented by $\boldsymbol{X} = (\boldsymbol{x}_1 \dots \boldsymbol{x}_n)$ contain n objects $\boldsymbol{x}_i \in \mathbb{R}^p$, p is the number of attributes. The objective is to group objects into c clusters $2 \leq c < n$. The variables used in the FCM-GK method are

- the matrix of membership degrees $(n \times c)$, $\boldsymbol{U} = (u_{ij})$ such that,

$$u_{ij} \in [0,1], \quad \sum_{j=1}^{c} u_{ij} = 1, \quad \sum_{i=1}^{n} u_{ij} > 0. \tag{1}$$

- the centroids of each group $\boldsymbol{\mathcal{V}} = \{\boldsymbol{v}_1, \dots, \boldsymbol{v}_c\}$, $\boldsymbol{v}_j \in \mathbb{R}^p$,

– the positive definite matrices, $\boldsymbol{\mathcal{S}} = \{\boldsymbol{S}_1, \ldots, \boldsymbol{S}_c\}$, inducing the norm of each group, $\boldsymbol{S}_j \in \mathbb{R}^{p \times p}$.

The K-Means algorithm and its variants focuses on minimizing the intra-class inertia. In FCM-GK, the unknown variables $(\boldsymbol{U}, \boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{S}})$ are determined by optimizing the following problem

$$\min_{(\boldsymbol{U},\boldsymbol{\mathcal{V}},\boldsymbol{\mathcal{S}})} J(\boldsymbol{U}, \boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{S}}) = \sum_{i=1}^{n}\sum_{j=1}^{c} u_{ij}^m \boldsymbol{q}_{ij}^\top \boldsymbol{S}_j \boldsymbol{q}_{ij}, \tag{2}$$

with the constraints, $\forall i, j \in [1, n] \times [1, c]$,

$$u_{ij} \geq 0, \sum_{j=1}^{c} u_{ij} = 1, \sum_{i=1}^{n} u_{ij} > 0, \tag{3}$$

$$\det(\boldsymbol{S}_j) = \rho_j, \quad \forall j \in [1, c] \tag{4}$$

where

$$\boldsymbol{q}_{ij} = \boldsymbol{x}_i - \boldsymbol{v}_j. \tag{5}$$

The fuzzy parameter m allows us to control the fuzziness of the partition. It's usually fixed at 2 [19]. The constraint Eq. (4) avoid trivial solution for the minimization is the solution with all $\boldsymbol{S}_j$ matrices zero. From a geometric point of view, ρ_j is the constant volume of the cluster j.

2.2 Alternating Optimization method (AO)

The method used by Gustafson and Kessel to resolve this constrained problem is the alternating optimization method (AO) [13]. It is also used for the other versions of *k-means*, such as PFCM [20] and ECM [18]. Starting from $(\boldsymbol{U}^0, \boldsymbol{\mathcal{V}}^0, \boldsymbol{\mathcal{S}}^0)$, the method successively minimizes $\boldsymbol{U}$, $\boldsymbol{\mathcal{V}}$ and $\boldsymbol{\mathcal{S}}$ using first-order optimality conditions :

$$\boldsymbol{U}^{k+1} = \arg\min_{\boldsymbol{U} \in \mathcal{U}} J(\boldsymbol{U}, \boldsymbol{\mathcal{V}}^k, \boldsymbol{\mathcal{S}}^k), \tag{6}$$

$$\boldsymbol{\mathcal{V}}^{k+1} = \arg\min_{\boldsymbol{\mathcal{V}}} J(\boldsymbol{U}^{k+1}, \boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{S}}^k), \tag{7}$$

$$\boldsymbol{\mathcal{S}}^{k+1} = \arg\min_{\boldsymbol{\mathcal{S}} \in \mathcal{S}_1} J(\boldsymbol{U}^{k+1}, \boldsymbol{\mathcal{V}}^{k+1}, \boldsymbol{\mathcal{S}}). \tag{8}$$

With the two sets of constraints (3) and (4) :

$$\mathcal{U} = \left\{ u_{ij} \geq 0, \quad \sum_{j=1}^{c} u_{ij} = 1, \quad \sum_{i=1}^{n} u_{ij} > 0 \right\},$$

$$\mathcal{S}_1 = \left\{ \boldsymbol{S}, \quad p \times p \text{ symmetric positive matrix}, \det(\boldsymbol{S}) = 1 \right\}.$$

Algorithm 1 describes the FCM-GK algorithm. It stops when the partition is stabilized, i.e. when the absolute error between two successive $\boldsymbol{U}$ matrices (membership degrees) is smaller than a threshold fixed at 10^{-3}. Note that for t iterations, its complexity is $O(tnc^2p)$ [10].

Algorithm 1 FCM-GK

Intput : c
$err = 0, k = 0,$
$\boldsymbol{U}^0$ random initialization or through FCM.
while $err > 10^{-3}$ **do**
  $k \leftarrow k + 1$
  compute $\boldsymbol{\mathcal{V}}^k : \boldsymbol{v}_j^k = \frac{\sum_{i=1}^n u_{ij}^{k-1} \boldsymbol{x}_i}{\sum_{i=1}^n u_{ij}^{k-1}}, \boldsymbol{q}_{ij}^k = \boldsymbol{x}_i - \boldsymbol{v}_j^k.$
  compute $\boldsymbol{\mathcal{S}}^k : \boldsymbol{\Sigma}_j^k = \sum_{i=1}^n u_{ij}^{k-1} \boldsymbol{q}_{ij}^k (\boldsymbol{q}_{ij}^k)^\top, \boldsymbol{S}_j^k = \det(\boldsymbol{\Sigma}_j)^{\frac{1}{p}} (\boldsymbol{\Sigma}_j^k)^{-1}.$
  compute $\boldsymbol{U}^k : u_{ij}^k = \left[\sum_{\ell=1}^c \frac{(\boldsymbol{q}_{ij}^k)^\top \boldsymbol{S}_j^k \boldsymbol{q}_{ij}^k}{(\boldsymbol{q}_{i\ell}^k)^\top \boldsymbol{S}_\ell^k \boldsymbol{q}_{i\ell}^k} \right]^{-1}.$
  $err \leftarrow \| \boldsymbol{U}^k - \boldsymbol{U}^{k-1} \|$
end while
Output : $\boldsymbol{U}^k$,$\boldsymbol{\mathcal{V}}^k$,$\boldsymbol{\mathcal{S}}^k$

3 Alternating Direction Method of Multipliers (ADMM)

The main idea of Alternating Direction Methods of Multipliers, introduced in the mid-1970s, is to use a decomposition/coordination process where the coordination is realized by Lagrange multipliers [8,9,12].

3.1 Augmented Lagrangien's formulation

ADMM does not only minimize the objective function but also the associated augmented Lagrangian. Before formulating the latter, it is necessary to introduce auxiliary variables into the original problem to obtain a constrained block optimization problem. First, we write the characteristic functions of the original constraints to introduce them in the function to be minimized.

$$I_{\mathcal{U}}(\boldsymbol{U}) = \begin{cases} 0 & \text{if } \boldsymbol{U} \in \mathcal{U} \\ +\infty & \text{else} \end{cases}, \quad I_{\mathcal{S}_1}(\boldsymbol{S}_j) = \begin{cases} 0 & \text{if } \boldsymbol{S}_j \in \mathcal{S}_1 \\ +\infty & \text{else.} \end{cases}$$

and $I_{\mathcal{S}_1}(\boldsymbol{\mathcal{S}}) = \sum_j I_{\mathcal{S}_1}(\boldsymbol{S}_j)$. In addition to the auxiliary variables $\boldsymbol{\mathcal{Q}}$ (5), we introduce the variables $\boldsymbol{\mathcal{P}}$

$$\boldsymbol{p}_{ij} = u_{ij} \boldsymbol{q}_{ij} = u_{ij} (\boldsymbol{x}_i - \boldsymbol{v}_j).$$

Thus, we reformulate the cost function as (2) which becomes :

$$J(\boldsymbol{U}, \boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{S}}, \boldsymbol{\mathcal{Q}}, \boldsymbol{\mathcal{P}}) = \sum_{i=1}^{n} \sum_{j=1}^{c} \boldsymbol{p}_{ij}^{\top} \boldsymbol{S}_j \boldsymbol{p}_{ij}. \tag{9}$$

To simplify the writings we note :
$\mathbb{U} = (\boldsymbol{U}, \boldsymbol{\mathcal{V}}, \boldsymbol{\mathcal{S}})$ the set of variables of the problem and $\mathbb{Q} = (\boldsymbol{\mathcal{Q}}, \boldsymbol{\mathcal{P}})$ the set of auxiliary variables. The constrained minimization problem becomes (2)-(8)

$$\min J(\mathbb{U}, \mathbb{Q}) + I_{\mathcal{U}}(\boldsymbol{U}) + I_{\mathcal{S}_1}(\boldsymbol{\mathcal{S}}) \tag{10}$$

under constraints

$$\boldsymbol{q}_{ij} = \boldsymbol{x}_i - \boldsymbol{v}_j, \tag{11}$$

$$\boldsymbol{p}_{ij} = u_{ij} \boldsymbol{q}_{ij}. \tag{12}$$

The coupling constraints are defined in such a way as to guarantee the equivalence (in terms of solution) with the original problem (2)-(8), while allowing independent optimization of variables. With (10)-(12), the function of the augmented Lagrangian is :

$$\begin{aligned} \mathscr{L}_r(\mathbb{U}, \mathbb{Q}, \mathbb{Y}) = J(\mathbb{U}, \mathbb{Q}) + I_{\mathcal{U}}(\boldsymbol{U}) + I_{\mathcal{S}_1}(\boldsymbol{\mathcal{S}}) \\ + \sum_{i,j} \left[\boldsymbol{y}_{ij}^{\top} (\boldsymbol{q}_{ij} - \boldsymbol{x}_i + \boldsymbol{v}_j) + \boldsymbol{z}_{ij}^{\top} (\boldsymbol{p}_{ij} - u_{ij} \boldsymbol{q}_{ij}) \right] \\ + \frac{r}{2} \sum_{i,j} \left[\| \boldsymbol{q}_{ij} - \boldsymbol{x}_i + \boldsymbol{v}_j \|^2 + \| \boldsymbol{p}_{ij} - u_{ij} \boldsymbol{q}_{ij} \|^2 \right] \end{aligned} \tag{13}$$

where $r > 0$ is the penalty term, $\| \cdot \|$ is the Euclidean norm, $\boldsymbol{y}_{ij}$ and $\boldsymbol{z}_{ij}$ are the Lagrange multipliers associated with the constraints of the auxiliary variables (11) and (12), represented by $\mathbb{Y} = (\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{Z}})$.

3.2 Application of ADMM

We apply the ADMM method to the augmented Lagrangian (13) by the following iterative algorithm. Starting with $\mathbb{Q}^0 : (\boldsymbol{\mathcal{Q}}^0, \boldsymbol{\mathcal{P}}^0)$ and $\mathbb{Y}^0 : (\boldsymbol{\mathcal{Y}}^0, \boldsymbol{\mathcal{Z}}^0)$, we successively compute $\mathbb{U}^k : (\boldsymbol{U}^k, \boldsymbol{\mathcal{V}}^k, \boldsymbol{\mathcal{S}}^k)$, $\mathbb{Q}^k : (\boldsymbol{\mathcal{Q}}^k, \boldsymbol{\mathcal{P}}^k)$ and $\mathbb{Y}^k : (\boldsymbol{\mathcal{Y}}^k, \boldsymbol{\mathcal{Z}}^k)$ by the following procedure.

$$\mathbb{U}^{k+1} = \arg\min_{\mathbb{U}} \mathscr{L}_r(\mathbb{U}, \mathbb{Q}^k, \mathbb{Y}^k), \tag{14}$$

$$\mathbb{Q}^{k+1} = \arg\min_{\mathbb{Q}} \mathscr{L}_r(\mathbb{U}^{k+1}, \mathbb{Q}, \mathbb{Y}^k), \tag{15}$$

$$\boldsymbol{y}_{ij}^{k+1} = \boldsymbol{y}_{ij}^{k} + r(\boldsymbol{q}_{ij}^{k+1} - \boldsymbol{x}_i + \boldsymbol{v}_j^{k+1}), \tag{16}$$

$$\boldsymbol{z}_{ij}^{k+1} = \boldsymbol{z}_{ij}^{k} + r(\boldsymbol{p}_{ij}^{k+1} - u_{ij}^{k+1} \boldsymbol{q}_{ij}^{k+1}). \tag{17}$$

Note that the iterations of the ADMM method (14)-(17) admit exact updates if 1) the function is bi-convex, i.e., convex along $\mathbb{U}$ for $\mathbb{Q}$ fixed and reciprocally, and 2) if the constraints are bi-affine, i.e., affine in $\mathbb{U}$ for $\mathbb{Q}$ fixed and reciprocally [4]. In (10), $I_{\mathcal{S}_1}(\boldsymbol{\mathcal{S}})$ is non-convex because of the constraint $\det(\boldsymbol{S}_j) = 1$. To ensure the convergence of the method, it is sufficient to fix a number it_a of repetitions of the relaxation blocks (14)-(15), recommended $it_a = 5$, before updating the multipliers [11,16].

Solution of the subproblem (14) **in** $\mathbb{U}$

Assuming the auxiliary variables $\mathbb{Q}$ and the multipliers $\mathbb{Y}^k$ fixed, the problem (14) of the augmented Lagrangian (13) is decoupled according to each variable of $\mathbb{U}$, to be optimized separately.

$$\boldsymbol{\mathcal{V}}^{k+1} = \arg\min_{\boldsymbol{\mathcal{V}}} \sum_{i=1}^{n}\sum_{j=1}^{c} (\boldsymbol{y}_{ij}^k)^\top (\boldsymbol{q}_{ij}^k - \boldsymbol{x}_i + \boldsymbol{v}_j) + \frac{r}{2} \parallel \boldsymbol{q}_{ij}^k - \boldsymbol{x}_i + \boldsymbol{v}_j \parallel^2, \quad (18)$$

$$\begin{aligned} \boldsymbol{U}^{k+1} = \arg\min_{\boldsymbol{U}} I_{\mathcal{U}}(\boldsymbol{U}) + \sum_{i=1}^{n}\sum_{j=1}^{c} (\boldsymbol{y}_{ij}^k)^\top (\boldsymbol{p}_{ij}^k - u_{ij}\boldsymbol{q}_{ij}^k) \\ + \frac{r}{2} \parallel \boldsymbol{p}_{ij}^k - u_{ij}\boldsymbol{q}_{ij}^k \parallel^2, \end{aligned} \quad (19)$$

$$\boldsymbol{\mathcal{S}}^{k+1} = \arg\min_{\boldsymbol{\mathcal{S}}} \sum_{i=1}^{n}\sum_{j=1}^{c} (\boldsymbol{p}_{ij}^k)^\top \boldsymbol{S}_j \boldsymbol{p}_{ij}^k + I_{\mathcal{S}_1}(\boldsymbol{\mathcal{S}}). \quad (20)$$

The subproblems (18)-(20) are solved by taking the first-order optimality conditions, as for the AO method. Thus, the formulations obtained are quite close :

$$\boldsymbol{v}_j^{k+1} = \frac{1}{n}\sum_{i=1}^{n} \left(\boldsymbol{x}_i - \boldsymbol{q}_{ij}^k - \frac{1}{r}\boldsymbol{y}_{ij}^k \right), \quad (21)$$

$$u_{ij}^{k+1} = \frac{1}{r^2 \alpha_i^k \parallel \boldsymbol{q}_{ij}^k \parallel^2} \left[r\alpha_i^k (\boldsymbol{q}_{ij}^k)^\top \tilde{\boldsymbol{z}}_{ij}^k + 1 - \sum_{\ell=1}^{c} \frac{(\boldsymbol{q}_{i\ell}^k)^\top \tilde{\boldsymbol{z}}_{i\ell}^k}{\parallel \boldsymbol{q}_{i\ell}^k \parallel^2} \right], \quad (22)$$

$$\boldsymbol{S}_j^{k+1} = \det(\boldsymbol{\Sigma}_j^k)^{1/p} (\boldsymbol{\Sigma}_j^k)^{-1}, \quad (23)$$

with,

$$\tilde{\boldsymbol{z}}_{ij}^k = \boldsymbol{z}_{ij}^k + r\boldsymbol{p}_{ij}^k, \quad \alpha_i^k = \frac{1}{r}\sum_{j=1}^{c} \frac{1}{\parallel \boldsymbol{q}_{ij}^k \parallel^2}, \quad \boldsymbol{\Sigma}_j^k = \sum_{i=1}^{n} \boldsymbol{p}_{ij}^k (\boldsymbol{p}_{ij}^k)^\top.$$

Solution of the subproblem (15) **in** $\mathbb{Q}$

Now assuming the variables $\mathbb{U}$ and the multipliers $\mathbb{Y}^k$ are fixed. The sub-problem in $\mathbb{Q} : (\boldsymbol{\mathcal{Q}}, \boldsymbol{\mathcal{P}})$ is an unconstrained optimization problem. Since $\boldsymbol{Q} \mapsto F(\mathbb{Q}) = \mathscr{L}_r(\mathbb{U}^{k+1}, \mathbb{Q}, \mathbb{Y}^k)$ is quadratic, the unique solution is obtained by solving the

gradient equation $\nabla F(\mathbb{Q}) = 0$. A simple calculation allows to obtain the following linear system in $(\boldsymbol{q}_{ij}, \boldsymbol{p}_{ij})$.

$$r(1 + (u_{ij}^{k+1})^2)\boldsymbol{q}_{ij} - ru_{ij}^{k+1}\boldsymbol{p}_{ij} = u_{ij}^{k+1}\boldsymbol{z}_{ij}^{k} - \boldsymbol{y}_{ij}^{k} + r(\boldsymbol{x}_i - \boldsymbol{v}_j^{k+1}) \tag{24}$$

$$-ru_{ij}^{k+1}\boldsymbol{q}_{ij} + (2\boldsymbol{S}_j^{k+1} + r\mathbb{I})\boldsymbol{p}_{ij} = -\boldsymbol{z}_{ij}^{k} \tag{25}$$

It follows that at each iteration, we solve nc linear systems of size $2p$

$$\boldsymbol{A}_{ij}^{k} \begin{bmatrix} \boldsymbol{q}_{ij}, \\ \boldsymbol{p}_{ij} \end{bmatrix} = \boldsymbol{b}_{ij}^{k} \tag{26}$$

$$\boldsymbol{A}_{ij}^{k} = \begin{bmatrix} r(1 + (u_{ij}^{k+1})^2)\mathbb{I} & -ru_{ij}^{k+1}\mathbb{I} \\ -ru_{ij}^{k+1}\mathbb{I} & 2S_j^{k+1} + r\mathbb{I} \end{bmatrix}, \boldsymbol{b}_{ij}^{k} = \begin{bmatrix} u_{ij}^{k+1}\boldsymbol{z}_{ij}^{k} - \boldsymbol{y}_{ij}^{k} + r(\boldsymbol{x}_i - \boldsymbol{v}_j^{k+1}) \\ -\boldsymbol{z}_{ij}^{k} \end{bmatrix}.$$

Algorithm

Algorithm 2 summarises the ADMM method. The stopping criterion is now the relative error on all primal and dual variables less than a threshold fixed at 10^{-3}. For t iterations, the complexity of our method is the same $O(tnc^2p)$. We initialize ADMM with random $\mathbb{U}$, same then AO, and construct all other variables $\mathbb{Q}$ (11)-(12). The Lagrange multipliers are initialized by solving the first order optimality condition (10)-(12), deriving the Lagrangian according to the variables $\boldsymbol{\mathcal{Q}}, \boldsymbol{\mathcal{P}}$: $\boldsymbol{z}_{ij}^{0} = 2\boldsymbol{S}_j^{0}\boldsymbol{p}_{ij}^{0}, \boldsymbol{y}_{ij}^{0} = u_{ij}^{0}\boldsymbol{z}_{ij}^{0}, \forall i, j$.

Algorithm 2 ADMM

Intput : Number of clusters c, penalty term r
$err = 1, k = 0,$
Random initialization or through ADMM(euclidean).
while $err > 10^{-3}$ **do**
  $k \leftarrow k + 1$
  for 1 until 5 **do** *(it_a repetitions of the relaxation blocks)*
    $\boldsymbol{\mathcal{V}}^k$,$\boldsymbol{\mathcal{S}}^k$ and $\boldsymbol{U}^k$ respectively according to (21), (23) et (22)
    $\boldsymbol{\mathcal{Q}}^k, \boldsymbol{\mathcal{P}}^k$ solving the system (26)
  end for
  $\boldsymbol{\mathcal{Y}}^k, \boldsymbol{\mathcal{Z}}^k$ respectively according to (16) et (17)
  $err \leftarrow \| (\mathbb{U}, \mathbb{Q})^k - (\mathbb{U}, \mathbb{Q})^{k-1} \| / \| (\mathbb{U}, \mathbb{Q})^k \|$
end while
Output : $\boldsymbol{U}^k$,$\boldsymbol{\mathcal{V}}^k$,$\boldsymbol{\mathcal{S}}^k$

4 Numerical experiences

In this section, we studied the performance of our ADMM method for the FCM problem with Mahalanobis distance. We used Matlab (R2021). The penalty term

r influences the performance of ADMM. In order to fine-tune this parameterization, we normalized all the data between $[-1, 1]$. Thus we take for r default the product of the dimensions to ensure the coordination of the variables, $r_d = 4cnp$. To find the optimal value r^*, we test several values and keep the one that converges the fastest in term of iterations.

In our study, we have fixed $m = 2$ and $\rho_j = 1, \forall j \in [1, c]$, such as [13]. We compare the three following algorithms:

- **FCM-GK**, the original method with alternating optimization on the GK model.
- $\mathbf{ADMM}_{r^*}$, ADMM applied to the augmented Lagrangian (13), with the optimal penalty value r^*.
- $\mathbf{ADMM}_{r_d}$, the same algorithm with the default value r_d.

In order to evaluate these different methods, we will use an external evaluation criterion that measures the similarity between two hard partitions : the clustering result and the ground truth. It is however necessary to transform the fuzzy partition into hard partition by assigning each object to the cluster with the highest membership. We used the Ajusted Rand Index (ARI) introduced by Hubert et al[14]. The ARI value is between 0 and 1, 1 corresponding to identical partitions.

We used 11 data sets. The first five corresponds to real data from the UCI library [1] : IRIS, WINE, SEEDS, WDBC, and DRYBEAN. We also used six synthetic data sets [2] : A1, A3, DIM32, DIM64, S1, and S3. We have referenced Table 1 their characteristics, i.e. the number of classes c, objects n and attributes p, as well as the optimal penalty parameter r^*, and by default r_d. For insensitivity of the results to the initialization for every algoithms, we first ran ADMM with the Euclidean distance ($\mathbf{ADMM}_{eu}$) with $r = 2.5$ and set a maximum number of iterations to 50 starting with random $\boldsymbol{U}^0$.

Table 1: Characteristics of data sets.

	IRIS	WINE	SEEDS	WDBC	DRYBEAN	A1	A3	DIM32	DIM64	S1	S3
c	3	3	3	2	7	20	50	16	16	15	15
n	150	178	210	569	13611	3000	7500	1024	1024	5000	5000
p	4	13	7	30	16	2	2	32	64	2	2
r^*	13	30	480	710	2.10^5	2000	1000	300	50	100	800
r_d	7200	27768	17640	136560	6.097.728	$4,8.10^4$	3.10^6	2^{21}	2^{22}	6.10^5	6.10^5

Table 2 shows that the ADMM methods perform better overall than the FCM-GK method, except for DRYBEAN, where FCM-GK is better. It seems that the larger number of individuals per class and the ratio between the number of clusters and the number of individuals explain this behavior.

[1] https://archive.ics.uci.edu/ml/datasets.php
[2] https://cs.joensuu.fi/sipu/datasets/

Table 2: ARI score (UCI).

	IRIS	WINE	SEEDS	WDBC	DRYBEAN
FCM-GK	0.74	0.34	0.72	0.41	**0.70**
ADMM$_{r^*}$	**0.78**	0.81	0.71	**0.74**	0.32
ADMM$_{r_d}$	0.72	**0.90**	0.71	**0.74**	0.32

Table 3, corresponding to the results for the synthetic data, confirms this characteristic: the greater the number of clusters (A1, A3) the lower the ARI score. On the other hand, the greater the number of dimensions (DIM32, DIM64), the better the score.

Although their complexity is of the same order of magnitude, the ADMM method is the fastest especially with the default penalty r_d. Table 4 lists the number of iterations for some data, measured for ten different random initializations (ADMM$_{eu}$).

Table 3: ARI score (Synthetic data).

	A1	A3	DIM032	DIM064	S1	S3
FCM-GK	**0.90**	**0.93**	0.44	0.18	**0.97**	**0.66**
ADMM$_{r^*}$	0.23	0.16	**0.57**	**0.68**	0.33	0.24
ADMM$_{r_d}$	0.20	0.16	**0.57**	**0.68**	0.33	0.26

Table 4: Number of iterations (mean ± standard deviation).

	IRIS	WINE	SEEDS	WDBC	A1	S1
ADMM$_{eu}$	30±1	33±3	30 ±4	26 ±1	10 ±2	10 ±0
FCM-GK	67±0	113±0	41±0	35±0	197±75	92 ±37
ADMM$_{r^*}$	35±0	41±0	6±0	7±0	4±0	3±0
ADMM$_{r_d}$	**2**±0	**2**±0	**4**±0	**3**±0	**2**±0	**2**±0

5 Conclusion

We have proposed an application of the ADMM method for the FCM clustering model with the Mahalanobis distance. The interest of this method is to divide the problem into a sequence of simpler sub-problems, easy to solve. Convergence to the same minimum, assumed to be global, is ensured. The results obtained on several data sets (real or synthetic) show good performances, in terms of ratios of well-classified samples, when the number of clusters is not too large or when the number of dimensions is significantly higher. To simplify the use of our method, we have proposed a default value for the penalty term (hyperparameter), whose convergence is assured and close to that of the optimal value. Our methods need less iterations than FCM-GK to converge and consequently are faster regarding the execution time.

The results are very encouraging. To confirm them, we wish to apply our method to a biology dataset, where many objects to be classified have a large number of attributes. To facilitate the use of our method, a formulation with an adaptive penalty is envisaged to replace the study of the optimal r. Finally, our study opens the possibility to apply the ADMM method to other clustering methods, having a non-convex objective function particularly those using alternating optimization.

References

1. Anter, A.M., Hassenian, A.E., Oliva, D.: An improved fast fuzzy c-means using crow search optimization algorithm for crop identification in agricultural. Expert Systems with Applications **118**, 340–354 (2019)
2. Bezdek, J.C.: Fuzzy Mathematics in pattern classification. Cornell University (1973)
3. Bezdek, J., Dunn, J.: Optimal fuzzy partitions: A heuristic for estimating the parameters in a mixture of normal distributions. IEEE Transactions on Computers **100**(8), 835–838 (1975)
4. Boyd S., Parikh N., Chu E., Peleato B., Eckstein J.: Distributed optimization and statistical learning via alternating direction method of multipliers. Foundation and Trends in Machine Learning **3**, 1–122 (2011)
5. Cai, W., Zhai, B., Liu, Y., Liu, R., Ning, X.: Quadratic polynomial guided fuzzy c-means and dual attention mechanism for medical image segmentation. Displays **70**, 102106 (2021)
6. Eckstein, J.: Parallel alternating direction multiplier decomposition of convexprograms. Journal of Optimization Theory and Applications (1), 39–62 (1994)
7. F. Yu, X.C., Zhao, L.: ADMM for efficient deep learning with global convergence. In: KDD 19: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery Data Mining. pp. 111–119 (2019)
8. Fortin, M., Glowinski, R.: Augmented Lagrangian Methods: Applications to the Numerical Solution of Boundary-Value Problems. North-Holland, Amsterdam (1983)
9. Gabay, D., Mercier, B.: A dual algorithm for the solution of nonlinear variational problems via finite element approximations. Computers and Mathematics with Applications **2**, 17–40 (1976)
10. Ghosh, S., Dubey, S.K.: Comparative analysis of k-means and fuzzy c-means algorithms. International Journal of Advanced Computer Science and Applications **4**(4) (2013)
11. Glowinski R., Le Tallec P.: Augmented Lagrangian and Operator-splitting Methods in Nonlinear Mechanics. SIAM, Philadelphia (1989)
12. Glowinski R., Marocco A.: Sur l'approximation par éléments finis d'ordre un, et la résolution par pénalisation–dualité, d'une classe de problèmes de Dirichlet non linéaires. RAIRO-AN **9**(2), 41–76 (1975)
13. Gustafson, D., Kessel, W.: Fuzzy clustering with a fuzzy covariance matrix. In: 1978 IEEE conference on decision and control including the 17th symposium on adaptive processes. pp. 761–766. IEEE (1979)
14. Hubert, L., Arabie, P.: Comparing partitions. Journal of classification **2**(1), 193–218 (1985)

15. Jain, A.K., Dubes, R.C.: Algorithms for clustering data. Prentice-Hall, Inc. (1988)
16. Koko J.: Parallel Uzawa method for large-scale minimization of partially separable functions. Journal Optimization Theory Applications **158**, 172–187 (2013)
17. Koko J., Jehan-Besson S.: An augmented lagrangian method for TV_g+L^1-norm minimization. Journal Mathematica Imaging Vision **38**, 182–196 (2010)
18. Masson, M.H., Denoeux, T.: Ecm: An evidential version of the fuzzy c-means algorithm. Pattern Recognition **41**(4), 1384–1397 (2008)
19. Pal, N.R., Bezdek, J.C.: On cluster validity for the fuzzy c-means model. IEEE Transactions on Fuzzy systems **3**(3), 370–379 (1995)
20. Pal, N.R., Pal, K., Keller, J.M., Bezdek, J.C.: A possibilistic fuzzy c-means clustering algorithm. IEEE transactions on fuzzy systems **13**(4), 517–530 (2005)
21. Yin, S., Li, H.: Hot region selection based on selective search and modified fuzzy c-means in remote sensing images. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing **13**, 5862–5871 (2020)