

# Clustering Problem with 0–1 Quadratic Programming

**Khalid Haddouch, Ahmad El Allaoui, Abdelhafid Messaoudi,  
Karim El Moutaouakil and El Wardani Dadi**

**Abstract** The most unsupervised methods of classification suffer from several performance problems, especially the class number, the initialization start points and the solution quality. In this work, we propose a new approach to estimate the class number and to select a set of centers that represent, fiddly, a set of given data. Our key idea consists to express the clustering problem as a bivalent quadratic optimization problem with linear constraints. The proposed model is based on three criterions: the number of centers, the density data and the dispersion of the chosen centers. To validate our proposed approach, we use a genetic algorithm to solve the mathematical model. Experimental results applied on IRIS Data, show that the proposed solution selects an adequate centers and leads to a reasonable class number.

**Keywords** Bivalent quadratic optimization problem · Clustering problem · Data IRIS · Genetic algorithm

---

K. Haddouch (✉) · A.E. Allaoui · A. Messaoudi · K.E. Moutaouakil · E.W. Dadi  
National School of Applied Sciences of Al Hoceïma,  
University Mohammed First, Oujda, Morocco  
e-mail: haddouchk@yahoo.fr

A.E. Allaoui  
e-mail: hmad666@gmail.com

A. Messaoudi  
e-mail: messaoudi1968abdelhafid@gmail.com

K.E. Moutaouakil  
e-mail: yassirkarimimane@gmail.com

E.W. Dadi  
e-mail: wrd.dadi@gmail.com

## 1 Introduction

The classification problem consists of partitioning a set of data into clusters (classes, groups, subsets, ...) [1]. A cluster is described by considering the internal homogeneity and the external separation; in this sense, the elements of the same class should be similar to each other, while those belonging to different classes should be not.

The classification problem has been applied in a wide variety of fields, especially engineering, computer science, life and medical science, social science, and economy [2]. Several methods are proposed to solve the classification problem such as K-means, ISODATA, SVM, SOM, trees method and the Bayesian method [1–4]. Most of them require a priori specifying the number of classes  $k$ . Indeed, the quality of resulting clusters is, largely, depends on the estimated  $k$ . An algorithm can, always, generate a division, no matter whether the structure exists or not.

Determining the number of clusters in a data set is a frequent problem in data clustering and is a distinct issue from the process of solving the clustering problem [5]. This work aims to propose a new approach to estimate the number of classes  $k$  and to select a set of centers that represent fiddly a given data set. Our solution consists to express the clustering problem as a bivalent quadratic optimization problem with linear constraints. The proposed model is based on three criterions: the number of centers, density data and dispersion of the chosen centers. For its performance, the genetic algorithm is used in order to solve this model. In this context, we have proposed the classical mutation and crossover operators; the selection function is nothing but the objective function of the proposed model.

This paper is organized as follow: in Sect. 2, we present the 0–1 mathematical programming model for the unsupervised classification problem. The genetic algorithm for solving the proposed model is presented in Sect. 3. While in Sect. 5, the performances of this new method are evaluated by some experimental results. The last section concludes this work.

## 2 The 0–1 Mathematical Programming Model for the Unsupervised Classification Problem

Let  $D = \{d_1, \dots, d_n\} \subset \mathbb{R}^m$  be the set of data, where  $n$  and  $m$  are integer numbers. The Unsupervised Classification Problem (UCP) looks for a set  $S$ , called set of centers, of an optimal size that represents fiddly the set  $D$ . In this part, we express the UCP problem as a bivalent quadratic problem with linear constraints, such that  $S$  is a subset of  $D$ . To this end, we define some necessary concepts. From a computational point of view, the UCP is one of the most difficult optimization problems; it is an *NP*-hard problem and the existence of efficient heuristic approaches for the general case cannot be assured.

### 2.1 The Sample Density

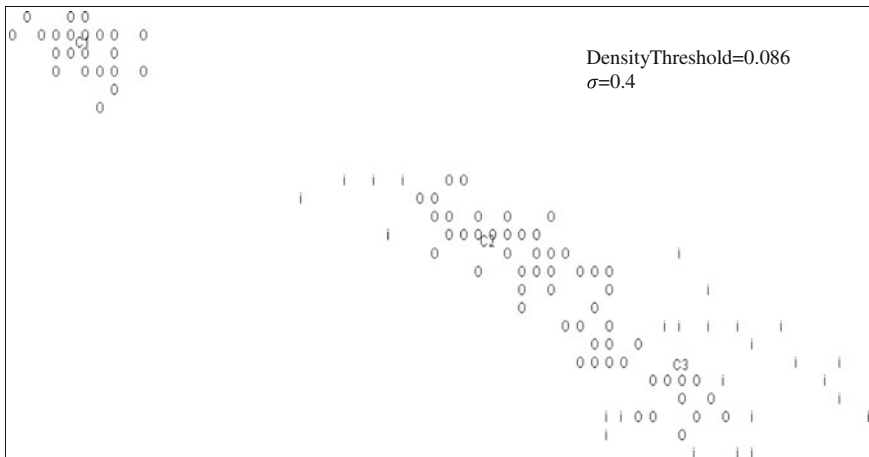
Let  $\sigma$  be a non negative number, for each sample  $d_i$ , we introduce the set  $A_i = \{d_j / \|d_i - d_j\| \leq \sigma\}$ . The *density parameter* of  $d_i$  is given by  $\alpha_i = |A_i|/n$ , where  $|A_i|$  is the size of the set  $A_i$ . In this sense, an isolated sample is the one that has a low density; an interior point is the one that has a large density. In this context, the Fig. 1 represents the IRIS Data for  $\sigma = 0.4$  and  $0.086$  as density threshold.

Noted that, the right centers of the IRIS Data are  $c1 = (1.464, 0.244)$ ,  $c2 = (4.242, 1.336)$ ,  $c3 = (5.57, 2.016)$ . Basing on the samples density, the problem UCP looks for one set  $S$  of center among  $D$  such that:

- The elements of  $S$  should have a large dispersion,
- The  $|density(S) - density(D)|$  is small as possible,
- The size of the center set  $S$  is small as possible.

### 2.2 The Decision Variable

Let  $S$  be a centers set among  $D$ ; for each sample  $d_i$ , we introduce the binary variable  $x_i$  that equals to 1 if the sample  $d_i$  is into  $S$ , 0 else. The vector decision is denoted by  $x = (x_1, \dots, x_n)^t$ .



**Fig. 1** The isolated point, represented by the character 'i', and the interior ones, represented by the character 'o', of the IRIS data

### 2.3 The Decision Dispersion

Let  $\beta_{ij} = \|d_i - d_j\|$  be the distance between the samples  $d_i$  and  $d_j$ . The deviation of the selected samples around the sample  $d_i$  is given by  $x_i \sum_{j=1}^n \beta_{ij} x_j$ . In this sense, for some vector decision  $x = (x_1, \dots, x_n)^t$ , we measure the dispersion of this decision by the quantity:

$$f_1(x) = \sum_{i,j=1}^n x_i \beta_{ij} x_j \quad (1)$$

*Remark 1* There exist several methods to measure this dispersion; such as interquartile range and standard deviation. These criterions of dispersion measure geometrically the dispersion of all data in order to specify automatically the center which represents these data. This type of dispersion is chosen for three reasons: It is less sensitive to extreme data, it represents all data and is easy to implement it.

### 2.4 The Decision Density

For a decision vector  $x = (x_1, \dots, x_n)^t$ , the total density of the selected centers is given by:

$$f_2(x) = \sum_{i=1}^n \alpha_i x_i \quad (2)$$

### 2.5 The Centers Set Size

For a decision vector  $x = (x_1, \dots, x_n)^t$ , the centers number is calculated by:

$$f_3(x) = \sum_{i=1}^n x_i \quad (3)$$

Basing on the criterions  $f_1$ ,  $f_2$  and  $f_3$ , we can define several bivalent quadratic problems with linear constraints. In our case, we fix some thresholds for the decision density, and then we look for one decision that maximizes the centers dispersion and minimizes the number of centers:

$$(P_\sigma) : \begin{cases} \text{Max} \left( \sum_{i,j=1}^n x_i \beta_{ij} x_j - \lambda \sum_{i=1}^n x_i \right) \\ \text{SC} : \\ \sum_{i=1}^n \alpha_i x_i \geq td \\ x_i \in \{0, 1\}, \quad i = 1, \dots, n \end{cases} \quad (4)$$

where  $\lambda$  is a penalty parameter. This penalty term is must be chosen in order to equilibrate the compromise between the dispersion and the number of centers terms.

In the next section, we will use the genetic algorithm to solve the problem  $P_\sigma$ .

### 3 Genetic Algorithm for Solving the Proposed Model

Since the proposed model is NP-complete, we use the genetic algorithm as optimization tool [6]. In this regard, we define our own coding, fitness function, selection mechanisms, crossover and mutation operators.

#### 3.1 Coding

As it is known, the UCP problem is a 0–1 quadratic programming. Then, it is natural to use the binary codes in order to produce our population; see Fig. 2. The size of each individual is equals to  $n$  (number of data).

#### 3.2 Fitness Function

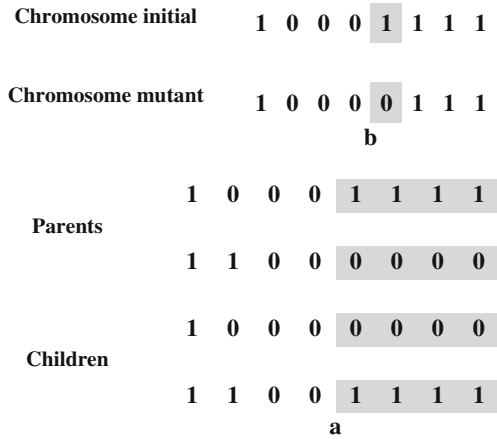
In order to evolve good solutions and to implement natural selection, the notion of fitness, which evaluates the solution, is used. In this case, the fitness function is no think but the objective function of our model:

$$f(x) = \sum_{i=1}^n \sum_{j=1}^n x_i \beta_{ij} x_j - \lambda \sum_{i=1}^n x_i \quad (5)$$

**Fig. 2** Example of the encoding individuals

0	1	0	0	0	1	0	0	1
---	---	---	---	---	---	---	---	---

**Fig. 3** Crossover and mutation operators



### 3.3 Mechanism of Selection Genetic Operators

At each step, a new population is created by applying the genetic operators: selection, crossover and mutation [7].

At the level of selection, the main idea is to prefer better solutions over worse ones. In this work, the type of the selection used is roulette-wheel selection. This type of selection guarantees a good luck to the good individual in order to be selected in the future generation, and a bad luck to the unsuitable individual targeted to be vanished in the future generation.

Crossover consists in building two new chromosomes from two old ones referred to as the parents, Fig. 3a. Mutation realizes the inversion of one or several genes in a chromosome, Fig. 3b.

After several experiments, the population size is fixed in function of the number  $n$  of variables. So, the size of population is equal to the number of data set  $n$ . In general, the probability of applying crossover operator is equal to 0.6 and the probability of applying mutation operator is equal to 0.02. The stopping criterion is based on a maximal number of iterations and/or when the performance function doesn't change.

## 4 Experimentation

To evaluate the performance of our method, the parameters  $\alpha$  and  $\beta$  are calculated from data iris. Several runs have been conducted for different values of the density parameter. To measure the performance of used algorithms, we used the following measures:

**Table 1** Results for different values of the density parameter

$\sigma$	Decision		Class 1				Class 2				Class 3			
	MCN	MTE	MCN	MInP	MIsP	MEC1	MCN	MInP	MIsP	MEC2	MCN	MInP	MIsP	MEC3
<b>0.4</b>	7	12.234	2	2	0	0.280	3	3	0	0.853	3	3	1	0.874
<b>0.36</b>	7	12.461	3	2	1	0.568	2	1	1	0.695	2	1	1	0.738
<b>0.32</b>	6	12.461	2	1	1	0.568	2	1	1	0.695	2	1	1	0.738
<b>0.28</b>	5	10.371	2	0	0	0.459	2	2	0	0.782	2	1	1	0.899
<b>0.24</b>	5	10.371	2	0	0	0.459	2	2	0	0.782	2	1	1	0.899
<b>0.2</b>	7	12.528	2	1	1	0.459	3	2	1	0.781	3	2	1	0.899
<b>0.18</b>	7	12.121	2	2	0	0.459	3	2	1	0.781	2	0	2	0.899
<b>0.16</b>	6	11.802	2	2	0	0.459	2	1	1	0.781	2	1	1	0.899
<b>0.12</b>	7	10.886	1	1	0	0.240	4	2	2	3.826	2	1	1	0.474
<b>0.11</b>	7	10.886	1	1	0	0.240	4	2	2	3.86	2	1	1	0.474

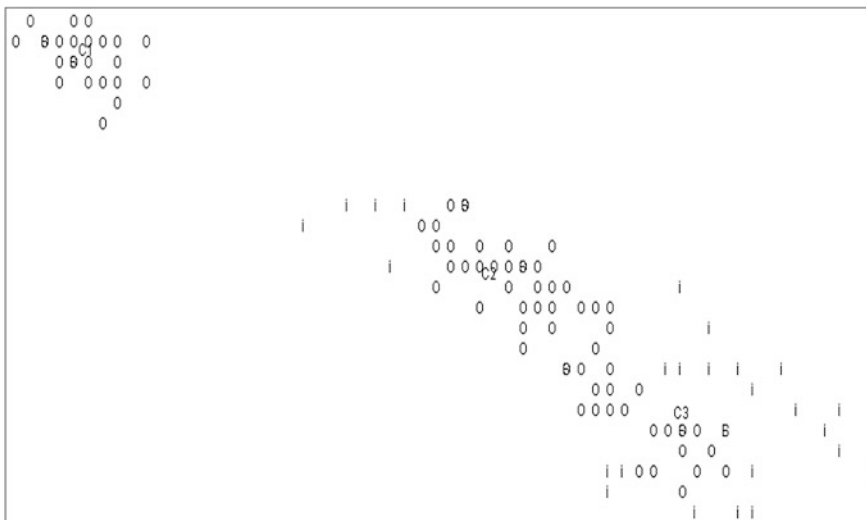
- MCN: Mean Class Number,
- MTE: Mean Total Error,
- MInP: Mean Interior Point,
- MIsP: Mean Isolated Point,
- MEC $i$ : Mean Error of the Class  $i$ .

The parameter  $\sigma$  is randomly chosen from the interval  $[0.4, 0.11]$ ; see the Table 1. The density parameter was fixed near to 70 %.

As shown in the Table 1, the mean number of selected centers using our approach is 6. The selected centers are equitably reparted between the three classes of the Iris Data; see the Figs. 4 and 5. In fact, our method assigns almost two centers to each class; see Table 1.

As shown in the Fig. 6, for  $\sigma > 0.15$ , the errors MEC1, MEC2 and MEC3 become almost constant. In this sense, the density has a low impact on the decision errors. Then, we can deduce that our method is consistent.

The most selected data are interiors samples; this means all the selected samples are representative ones. It should be noted that we can use the Euclidian distance to group the collected centers to obtain the right centers of the data under study [8].



**Fig. 4** The selected centers, for  $\sigma = 0.4$ , are represented by the character 's'



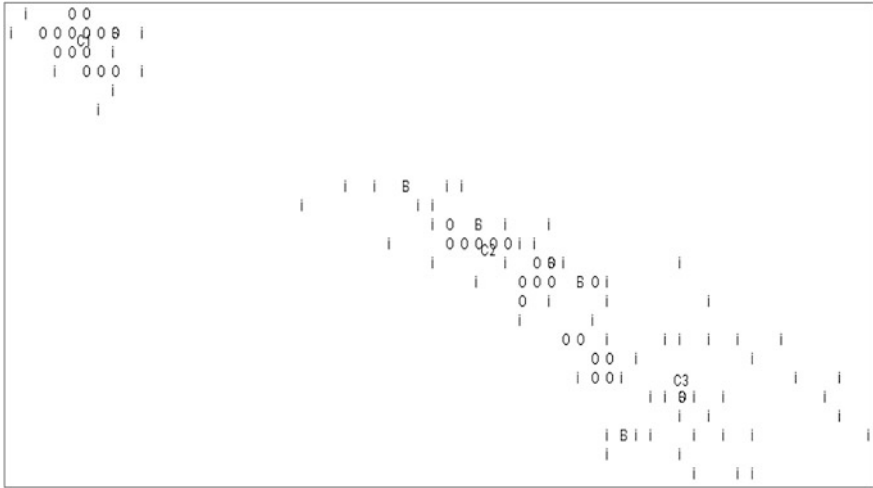


Fig. 5 The selected centers for  $\sigma = 0.12$  are represented by the character 's'

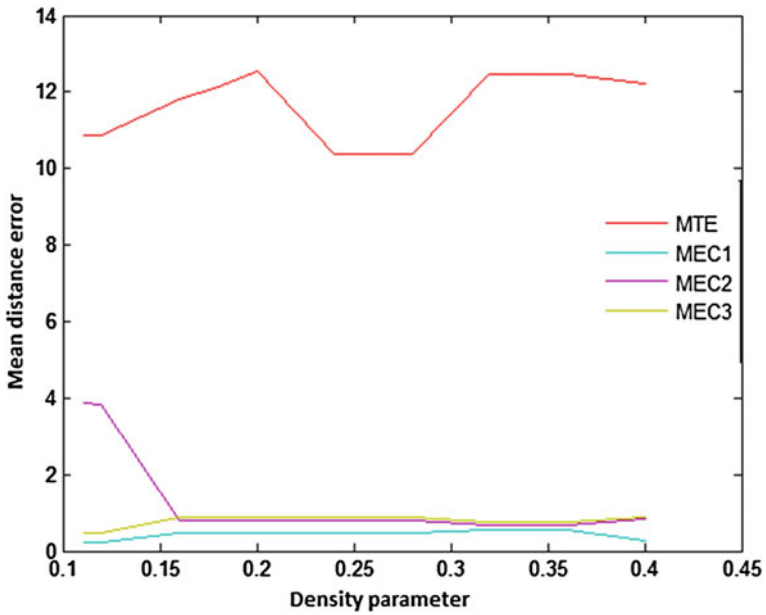


Fig. 6 Mean errors density versus density parameter  $\sigma$

## 5 Conclusion

In this work, we have proposed a new approach to estimate the class number and to select a set of centers that represent fiddly the D set. Our approach consists expressing this problem as a bivalent quadratic problem with linear constraints ensuring a large dispersion of the chosen centers, respects the information percent imposed by the experts of different domains. In the future, we will combine our approach with some performance labeling method to construct a new system able to solve classification problem.

## References

1. Everitt, B., Landau, S., Leese, M.: Cluster Analysis. Arnold, London (2001)
2. Xu, R., Wunsch, D.: Survey of clustering algorithms neural networks. *IEEE Trans. Neural Netw.* **16**(3), 645–678 (2005)
3. Hall, L.Q., Özyurt, I.B., Bezdek, J.C.: Clustering with a genetically optimized approach. *IEEE Trans. Evol. Comput.* **3**(2), 103–110 (1999)
4. Ball, G., Hall, D.: A clustering technique for summarizing multivariate data. *Behav. Sci.* **12**, 153–155 (1967)
5. Abascal, F., Valencia, A.: Clustering of proximal sequence space for the identification of protein families. *Bioinformatics* **18**, 908–921 (2002)
6. Holland, J.: *Adaptation in Natural an Artificial Systems*, 2nd edn. Press, M.I.T (1992)
7. Goldberg, D.E.: *Design of Innovation: Lessons From and for Competent Genetic Algorithms*. Kluwer Academic Publishers, Boston, MA (2002)
8. Jain A.K., Dubes, R.C.: *Algorithms for Clustering Data*. Prentice Hall (1988)