Evolutionary computing for unsupervised clustering methods

Do, Anh Duc

2009

Do, A. D. (2009). Evolutionary computing for unsupervised clustering methods. Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/47494>

<https://doi.org/10.32657/10356/47494>

Nanyang Technological University

Downloaded on 26 Apr 2023 20:20:52 SGT

EVOLUTIONARY COMPUTING FOR UNSUPERVISED CLUSTERING METHODS

DO ANH DUC

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Master of Engineering

2009

ACKNOWLEDGMENTS

ACKNOWLEDGMENTS

This dissertation summarizes the research work completed during the author's Master of Engineering candidature at the School of Computer Engineering, Nanyang Technological University, Singapore. Over this period of time, the author has received considerable support from many personnel, but none more so than from his supervisor Dr. Cho Siu Yeung. His thoughtful guidance, invaluable suggestions and constant encouragement has always been the source of inspiration in his research. Even more, from his continual enthusiasm, passion and serious attitude towards research, the author has also grasped the art of being a professional researcher, for which he is extremely grateful.

He is grateful to his seniors – Nguyen Quang Huy, Wong Jia Jun and juniors – Huyen Ton Nu Hoang Lan, Pham Manh Tung for their insightful discussions and suggestions on relevant research fields. He would also like to take this opportunity to thank all personnel of the Emerging Research Laboratory, School of Computer Engineering and Nanyang Technological University for the support provided during his candidature.

Finally, he expresses his deepest gratitude to his parents Do Van Cuong and Nguyen Thi Hinh, and sisters Do Thi Thanh Hang and Do Thanh Minh for their selfless love and constant support.

ABSTRACT

ABSTRACT

Clustering represents a core research area of machine learning. It has been widely used in data processing and system learning where characteristics of the feature vectors, such as “localization” are defined or learned. Clustering algorithm attempts to organize unlabeled feature vectors into clusters such that within the same group, feature vectors are considered to be more similar than others of different groups. Among available clustering methods, Hard C-means (HCM) clustering represents non-overlapping clustering category while Fuzzy C-means clustering (FCM) represents the overlapping category. FCM enhances HCM with the introduction of fuzzy concept which is deemed closer to human cognition system.

In this research study, new approaches are explored to address the limitations of traditional clustering methods. Genetic Algorithm (GA) based clustering method with global search capability is proposed to overcome the problem of local optimality in traditional clustering methods that use gradient descent. On the other hand, Memetic Algorithm (MA), which is a hybridization of GA and Local Search (LS), inherits the advantages of both approaches in exploring and exploiting the search effectively and efficiently. Here, an MA based clustering approach is proposed to improve the performance of GA based clustering. In particular, a novel Multi-objective memetic algorithm based clustering method is presented to mitigate the limitations of traditional single objective clustering approaches. It extends the clustering problem from single objective optimization to one containing multiple criteria to be optimized simultaneously. The proposed Multi-objective memetic method is shown to outperform many existing clustering approaches, and is found to be more appropriate for complex real world data sets.

TABLE OF CONTENTS

Table of Contents

Chapter 1	Introduction	7
1.1	Background and Motivation.....	7
1.2	Objectives.....	10
1.3	Contributions of the Thesis and Publications	11
1.3.1	Contributions of the Thesis.....	11
1.4	Organization of the Thesis.....	12
Chapter 2	Clustering and Optimization	14
2.1	Clustering.....	14
2.1.1	Hard C-means Clustering.....	14
2.1.2	Fuzzy C-means Clustering.....	16
2.2	Optimization Algorithms	21
2.2.1	Non-Linear Programming Problem	21
2.2.2	The Optimal Solution and Local Minimum Issue	25
2.3	Evolutionary Computing	26
2.3.1	Genetic Algorithm	27
2.3.2	Memetic Algorithm.....	29
Chapter 3	Single-Objective Clustering	32
3.1	Genetic Algorithm Based Single-Objective Clustering	32
3.2	Memetic Algorithm Based Single-Objective Clustering	37
3.2.1	Local Search for Memetic Algorithm	38
3.3	Cluster Validity of Clustering Method	39
3.4	Experimental Results and Discussion	42
3.4.1	Data Sets and Preprocessing.....	42
3.4.2	Clustering Validation.....	44

TABLE OF CONTENTS

3.4.3	Performance of the Single-Objective Clustering Algorithms	50
3.5	Concluding Remarks.....	58
Chapter 4	Multiple-Objective Clustering.....	60
4.1	Multi-Objective Clustering Problem	61
4.2	Related Works	63
4.3	Multi-Objective Memetic Algorithm for Clustering (MOMACL).....	64
4.3.1	Objective Functions	66
4.3.2	Global Search in MOMA.....	67
4.3.3	Local Search in MOMA.....	71
4.4	Experimental Result and Analysis	74
4.4.1	The Performance of MOMACL with Validation Indices	74
4.4.2	Performance of Clustering Methods with Real Complex Data Sets	80
	MOMA Pareto Front Analysis	81
4.5	Concluding Remarks.....	89
Chapter 5	Conclusion and Future Works.....	91
5.1	Conclusion.....	91
5.2	Future Works.....	92
5.2.1	The Compliment with MOCK	92
5.2.2	Model Selection in MOMA	93
5.2.3	Improving the Scalability of MOMA and Its Application to Large Scale Real Data Sets	93
Publications		95
References		96

LIST OF FIGURES

List of Figures

Figure 2.1: Error surface –local and global minimum	25
Figure 2.3: Flow chart of GA	27
Figure 2.4: Flow diagram of memetic algorithm.....	31
Figure 3.1: A flow diagram of GA based clustering	33
Figure 3.2: MA-FCM algorithm flowchart.....	39
Figure 3.3: TEST-9 dataset without noise (left) and with noise (right).....	42
Figure 3.4: Global optimal and local optimal solutions of TEST9 data set	52
Figure 3.5: Global optimal and local optimal solutions of IRIS data set.....	53
Figure 3.6: FCM, GA-FCM and MA-FCM performances	56
Figure 3.7: Convergence comparison of GA-FCM and MA-FCM	58
Figure 4.1: Clustering criteria measures	62
Figure 4.2: Applying local search improvement on offspring population.....	65
Figure 4.3: Illustration of weighted sum approach.....	73
Figure 4.4: [a] Globally and locally Pareto-optimal [b] An ideal set of non-dominated solutions	81
Figure 4.5: Pareto front of MOMACL algorithm conducted by the simulation of IRIS dataset.....	82
Figure 4.6: Real data set performance – Best recall	84
Figure 4.7: Real data set performance – Best average	86
Figure 4.8: Real data set performance – Success rate.....	87
Figure 4.9: Real data set performance – Processing time	88

LIST OF TABLES

List of Tables

<i>Table 2.1: Membership condition of HCM and FCM</i>	<i>16</i>
<i>Table 3.1: Details of the experimental datasets.....</i>	<i>42</i>
<i>Table 3.2: Clustering validity by TEST-9 dataset under noise-free condition.....</i>	<i>45</i>
<i>Table 3.3: Clustering validity by TEST-9 dataset under noise condition.....</i>	<i>45</i>
<i>Table 3.4: Cluster validity on IRIS dataset.....</i>	<i>46</i>
<i>Table 3.5: Cluster validity on GLASS data set</i>	<i>48</i>
<i>Table 3.6: Cluster validity on VOWEL data set.....</i>	<i>49</i>
<i>Table 3.7: Parameter settings for GA-FCM and MA-FCM.....</i>	<i>50</i>
<i>Table 4.1: MOMAFC setup.....</i>	<i>75</i>
<i>Table 4.2: Clustering validation indexes of MOMAFCM algorithm on IRIS data set.....</i>	<i>76</i>
<i>Table 4.3: Clustering validation indexes of MOMAFCM algorithm on GLASS data set.....</i>	<i>77</i>
<i>Table 4.4: Clustering validation indexes of MOMAFCM algorithm on VOWEL data set.....</i>	<i>78</i>
<i>Table 4.5: Real complex data sets.....</i>	<i>80</i>

Chapter 1

Introduction

1.1 Background and Motivation

Clustering is used widely in data processing and system learning where the characteristics of data features, such as “localization”, are defined. Clustering algorithm attempts to organize unlabeled feature vectors into clusters such that within the same group, feature vectors are considered to be more similar than others of different groups. The similarity between data features is normally defined by some “distance” metric between data vectors in the feature space. Some pre-defined objective function(s) is then minimized which reflect the state of the clustering process. Clustering is generally distinguished from classification based on the unsupervised nature of the former. Classification uses prior knowledge on the data samples such as expected labels, to define regions with features of different labels. Clustering, does not use prior knowledge of data labels, instead it employs some similarity criteria to define the different groupings of the data set, where samples of the same group or cluster are expected to share the same label.

There are two types of algorithms for clustering. They are: Exclusive Clustering [4] [5] and Overlapping Clustering [2] [3]. In exclusive clustering, data belongs exclusively to only

CHAPTER 1. INTRODUCTION

one cluster, i.e., no single data vector may belong to more than one cluster simultaneously. K-means clustering [4] [5] and Hard C-Means (HCM) clustering represent instances of exclusive clustering method. In contrast, overlapping clustering employs the concept of fuzzy sets to cluster data in a manner that a data vector may belong to two or more clusters with different degrees of membership. Membership functions are used to present the “belongingness” of the data to each class. Fuzzy C-means Clustering (FCM) [2] [3] is a typical overlapping clustering technique and differs from HCM clustering since the latter considers a hard partitioning. In FCM, Fuzzy set [1] is used instead of Boolean set to present the membership function as a means to improve the accuracy of clustering. Some properties of HCM are also found in FCM. First, the method is guaranteed to converge to some stable state which may not be the global optimum. Secondly, FCM is sensitive to the initial conditions defined. Thirdly, FCM is also sensitive to noise and outlier existing in the data. Fourth, FCM requires pre-specifying the number of initial clusters used which may lead to non-optimum clustering results.

In traditional clustering algorithms, objective function(s) are commonly proposed to quantify the health or fitness of the clustering process. In a minimization problem, a smaller objective function or fitness value indicates a better clustering state. As most traditional clustering methods employ gradient descent in their optimization search, the problem of local optimality persists. Genetic Algorithm (GA), on the other hand, is a stochastic search technique that has been widely and successfully applied for solving non-linear programming problems [17] [18]. In GA, each population consists of chromosomes (individuals) representing candidate solutions for an optimization problem. New populations are

CHAPTER 1. INTRODUCTION

subsequently generated by transforming individuals of the current population through stochastic operators (or genetic operators) that emulate evolution in biological systems. Since the clustering problem can be viewed or formulated as a non-linear programming problem, GA may be used to replace the conventional gradient descent in the search to overcome problems of local optimality.

However, although GA is capable of exploring and exploiting promising regions of the search space, they can however, take a relatively long time to locate the local optimum in a region of convergence and may sometimes not find the optimum with sufficient precision. One of the recent growing areas of Evolutionary Computing research is Memetic Algorithm (MA) [20] [21] [26]. MA has materialized as a synergy of GA and individual learning, and has displayed remarkable record of success in solving feature/gene selection [22] [23] [24], molecular structure analysis [25] and aircraft design [26] [27].

In this thesis, a MA based clustering methodology is introduced for improving traditional GA based clustering approaches. One of the major design issues of MA is on how to establish the hybrid combination of GA and individual learning as to what, when, and how individual learning or LS should be applied to the population of individuals.

The shape of a cluster may vary not only simply in spherical but also in different kinds of structures [28] [29]. As a consequence, the plethora of clustering algorithms available in the literature have assumed diverse clustering criterion or cluster definition in their search for the optimum cluster set [30]. However unless a priori knowledge about the dataset of interest is available, it is impossible to establish in advance which clustering criterion would be most appropriate for capturing the implicit structure of the data. Moreover, the same data can have

CHAPTER 1. INTRODUCTION

more than one relevant structure, each one in accordance with a different cluster definition and/or with a different refinement level (e.g. number of clusters). The common application of cluster analysis to explore a data set generally focuses on the discovery of only one dominant structure that best fits the data according to a pre-defined criterion. Hence to discover unique structures in the dataset, separate clustering algorithms, each defined with a distinctive objective or criterion is used to search on the data independently. As an attempt to mitigate the above limitations, multi-objective clustering methods have been proposed [31] [32] [33] as an effective alternative. Multi-objective clustering algorithm is capable of refining the clustering structure of the data sets with respect to more than one criterion simultaneously. The approach offers a set of alternative structures that represent different interpretations of the data and is especially useful for dataset containing complex structures. The optimal number of cluster may also be directly optimization based on the diverse cluster validation indexes. In this thesis, some alternatives to multi-objective clustering methods are analyzed and presented. Subsequently, a novel multi-objective memetic algorithm (MOMA) based clustering algorithm is proposed for clustering real world complex datasets.

1.2 Objectives

The limitations of traditional clustering have been briefly discussed in section 1.1 with greater details provided subsequently in Chapter 2. The present thesis aims to address the core limitations of traditional clustering by embarking on the paradigm of modern stochastic evolutionary search for handling clustering problems; particularly the field of memetic algorithm is investigated. In memetic computation, clustering is formulated as a non-linear

CHAPTER 1. INTRODUCTION

optimization problem. Unlike traditional clustering methods which employs gradient descent in their search and hence prone to convergence at local optima when dealing with real world complex dataset, MA however is less sensitive to the presence of local optima and is capable of exploring and exploiting the search space effectively and efficiently. Further, since it is not known in advance which clustering criterion and cluster size would most best suit the current dataset in hand, we considered a multi-objective memetic methodology that searches on multiple clustering criteria simultaneously.

1.3 Contributions of the Thesis and Publications

1.3.1 Contributions of the Thesis

Different from traditional clustering algorithms, Genetic Algorithm and its variants introduce new approaches to solve clustering problems. Traditional methods focus on gradient search algorithms and objective function selection which have the limitation of single objective optimization. In this thesis, we address the clustering problem with global and multi-objective optimizations.

A clustering problem can be viewed as an optimization problem where the algorithm goes through set of iterations to refine the cluster solutions. Traditional clustering methods make use of local search methods which optimize predefined single objectives. There are known limitations of traditional clustering methods such as local search, being single objective, supervised and unrobust. These limitations make the result of the algorithm very subjective to prior settings of objective function and the number of clusters. In chapter 3 of this thesis, the proposed Clustering Genetic and Memetic Algorithms (MA) are described and

CHAPTER 1. INTRODUCTION

shown to generate improved solution at higher rate of success in obtaining the global optimal solution than traditional clustering approaches. The second contribution lies in the proposal and development of a novel multi-objective memetic algorithm (MOMA) for solving clustering problems. MOMA represents an extension of our single-objective MA by optimizing more than one objective functions or clustering criteria simultaneously.

1.4 Organization of the Thesis

This thesis comprises of five chapters. They are organized as follows:

Chapter 2 summarizes the basic concepts of clustering and optimization. These include a brief review of conventional local search, genetic algorithm (GA) and memetic algorithm (MA). The concept of fuzzy clustering are first analyzed and discussed. This is followed by a discussion on the strengths and limitations of some state-of-the-art conventional, modern stochastic evolutionary and memetic optimization methods in the literature.

Chapter 3 presents a study of two single-objective clustering methods, namely GA based and MA based clustering methods. Several clustering validity indexes commonly used in the literature for clustering validation are introduced. The use of GA and MA for single objective clustering is then discussed. This is followed by a detail description on the genetic operations of the proposed GA based clustering method. Empirical results of the GA and MA based clustering on real world complex dataset with comparisons to traditional FCM methods are also presented in the section.

Chapter 4 introduces our novel Multi-Objective Memetic Algorithm (MOMA) for solving clustering problems involving multiple objective functions or clustering criteria

CHAPTER1. INTRODUCTION

simultaneously. In particular, the basic components of the proposed method are introduced and described with great details in the chapter. The results obtained from empirical study on MOMA for clustering real world complex dataset are also reported and analyzed in the chapter.

Finally, chapter 5 draws the major conclusion of this thesis. It also summarizes the contributions of this work and proposes some possible directions for future research.

Chapter 2

Clustering and Optimization

2.1 Clustering

Clustering has long been a popular field of machine learning. The most important feature of clustering algorithm lies in the ability to determine the implicit structure of a data set without using any prior labeled knowledge. Clustering has been successfully used in many applications, including pattern recognition [16], classification [12], data mining [13] and image segmentation [9] [10]. It has also been employed for performing data analysis and modeling [6] [7] in various contexts.

2.1.1 Hard C-means Clustering

Hard C-mean Clustering (or K-mean Clustering) is among one of the earliest and simplest learning algorithms proposed for handling clustering problems. The algorithm begins with a definition on the number of cluster C to be used. The C cluster centroids can be defined randomly or heuristically in a stratified manner such that they are uniformly spread. The next step is to identify the membership of each vector point of the given data set by associating it to the nearest centroid. The membership of a vector point to a centroid is 1 if the point is

CHAPTER 2. CLUSTERING AND OPTIMIZATION

associated with that centroid and 0 otherwise. The algorithm continues with iterative steps of re-calculating the new centroids and updating the membership of each vector point. The new centroids are defined in order to improve the compactness of the cluster which is represented by the objective function in Eqn. 2.1 and terminates when there are no more changes of cluster centroids.

The objective function is commonly defined as a squared error function:

$$J = \sum_{j=1}^N \sum_{i=1}^C u_{ij} \left\| \mathbf{x}_j^{(i)} - \mathbf{v}_i \right\|^2, \quad (2.1)$$

where $\left\| \mathbf{x}_j^{(i)} - \mathbf{v}_i \right\|^2$ is a distance measure between a data point $\mathbf{x}_j^{(i)}$ and the cluster centre or centroid $\mathbf{v}_i$. It represents a similarity of the data points to the clusters. The Hard C-mean Clustering algorithm is outlined as follows:

Begin

Initialize: $\mathbf{V} = [\mathbf{v}_i]$ matrix within the border of the sample space, $\mathbf{V}^{(0)}$

Iteration $k = 0$

Do

Assign each data sample to the group that has the closest centroid.

Re-calculate the centers vectors $\mathbf{V}^{(k)} = [\mathbf{v}_i]$ as average of samples belong to it.

Iteration $k = k+1$

While ($\left\| \mathbf{V}^{(k+1)} - \mathbf{V}^k \right\| > \varepsilon$)

End

Hard C-mean Clustering is a simple methodology that employs crispy logic in the membership function. The belongingness of each sample to a cluster limits feasibility of Hard C-mean Clustering for handling complex clustering problems.

CHAPTER2. CLUSTERING AND OPTIMIZATION

2.1.2 Fuzzy C-means Clustering

Fuzzy clustering, on the other hand, represents a subfield of general clustering approaches that has been shown to be advantageous over crisp (or hard) clustering since the commitment of a vector to only one specific class is not enforced. One of the major factors that influence the categorization of appropriate groups of vector points is the “distance measure” chosen for the problem at hand. For instance, many analytic fuzzy clustering approaches were derived from Bezdek’s fuzzy C-means (FCM) clustering [8] and have been used successfully in many real world applications. FCM clustering is a probabilistic clustering technique and differs from hard C-Means which employs hard partitioning. FCM employs fuzzy partitioning such that a data point can belong to more than one group with membership grades between 0 and 1. Because of this characteristic, the conditions of the membership would be different between the Hard C-Means and Fuzzy C-Means. They are tabulated in Table 2.1.

Hard C-means Clustering	Fuzzy C-means clustering
1. $u_{ij} \in \{0,1\}$ for all i and j ,	1. $u_{ij} \in [0,1]$ for all i and j ,
2. If $u_{ij}=1$ then $\sum_{k=1, k \neq i}^C u_{kj} = 0$ for all j ,	2. $\sum_{i=1}^C u_{ij} = 1$ for all j ,
3. $0 \leq \sum_{j=1}^N u_{ij} \leq N$ for all i	3. $0 < \sum_{j=1}^N u_{ij} < N$ for all i

Table 2.1: Membership condition of HCM and FCM

CHAPTER2. CLUSTERING AND OPTIMIZATION

In Table 2.1, $\mathbf{X} = [\mathbf{x}_j] \mid j = 1 \dots N$ represents a given data set of N feature vectors, u_{ij} is the membership value of data sample $\mathbf{x}_j$ associated to cluster $\mathbf{v}_i$, C represents the number of clusters or centroids. The algorithm of FCM is typically derived by the minimization of the least mean squared function as shown in (2.2):

$$J_m = \sum_{j=1}^N \sum_{i=1}^C u_{ij}^m \|\mathbf{x}_j - \mathbf{v}_i\|^2, 1 \leq m \leq \infty \quad (2.2)$$

where m is weighting exponent (typically greater than 1), u_{ij}^m is the degree of membership of $\mathbf{x}_j$ in the cluster i , $\mathbf{x}_j$ is the j^{th} d -dimensional measured data, $\mathbf{v}_i$ is the d -dimension centroid of the cluster, and $\|\cdot\|$ is the norm representing the similarity between the measured data and the center.

On the other hand, the within-group criterion can be used as an objective function to optimize the clustering solution in the FCM algorithm. The objective function can be defined as the sum of square errors function as below:

$$J_m(\mathbf{U}, \mathbf{v}) = \sum_{j=1}^N \sum_{i=1}^C (u_{ij})^m (d_{ij})^2 \quad (2.3)$$

where $\mathbf{U}$ is a fuzzy C -partition of $\mathbf{X}$. $\mathbf{V} = (\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_C)$ and $\mathbf{v}_i$ are the cluster centers or prototypes of $\mathbf{u}_i$ with $1 \leq i \leq C$; $(d_{ij})^2 = \|\mathbf{x}_j - \mathbf{v}_i\|^2$ is defined as an inner product induced norm. $m \in [1, \infty)$ is a weighting exponent.

The dissimilarity measure $d_{ij} = \|\mathbf{x}_j - \mathbf{v}_i\|$ is defined as the distance between each data point $\mathbf{x}_j$ and fuzzy prototype $\mathbf{v}_i$. The distance is then weighted by $(u_{ij})^2 = (\mathbf{u}_i(\mathbf{x}_j))^m$, where

CHAPTER 2. CLUSTERING AND OPTIMIZATION

the m^{th} power of $\mathbf{x}_j$'s membership in fuzzy cluster $\mathbf{v}_i$. Since each term of J_m is proportional to $(d_{ij})^2$, J_m is a squared error clustering criterion, and the solutions of $\text{minimize}\{J_m(\mathbf{U}, \mathbf{v})\}$ are least-squared error of J_m . The following theorem is then deduced.

Theorem 1[8]: $(\mathbf{U}, \mathbf{v})$ is minimal for $J_m(\mathbf{U}, \mathbf{v}) = \sum_{j=1}^N \sum_{i=1}^C (u_{ij})^m (d_{ij})^2$ only if

$$u_{ij} = 1 / \left[\sum_{k=1}^C \left(\frac{d_{ij}}{d_{kj}} \right)^{2/m-1} \right] \quad (2.4)$$

$$\text{Or } \mathbf{v}_i = \sum_{j=1}^N (u_{ij})^m \mathbf{x}_j / \sum_{j=1}^N (u_{ij})^m \quad (2.5)$$

Proof [8]:

Assume $\mathbf{v}$ is fixed and we define $g_m(\mathbf{U}) = J_m(\mathbf{U}, \mathbf{v})$ for any $\mathbf{U}$. Since the columns of $\mathbf{U}$ are independent, therefore

$$\begin{aligned} \min \{g_m(\mathbf{U}, \mathbf{v})\} &= \min \left\{ \sum_{j=1}^N \sum_{i=1}^C (u_{ij})^m (d_{ij})^2 \right\} \\ &= \sum_{j=1}^N \left[\min_{\mathbf{u}_j \in \text{conv}(\mathbf{B}_c)} \left\{ \sum_{i=1}^C (u_{ij})^m (d_{ij})^2 \right\} \right] \end{aligned} \quad (2.6)$$

$$\text{where } \text{conv}(\mathbf{B}_c) = \left\{ \mathbf{u}_j \in \mathbf{R}^C \mid \sum_{i=1}^C u_{ij} = 1; u_{ij} \geq 0 \right\} \quad (2.7)$$

$$\text{Using Lagrange multipliers, let } g_{mj}(\mathbf{u}_j) = \sum_{i=1}^C (u_{ij})^m (d_{ij})^2 \quad (2.8)$$

CHAPTER 2. CLUSTERING AND OPTIMIZATION

The Lagrange function can be defined as:

$$F_j(\lambda, \mathbf{u}_j) = \sum_{i=1}^C (u_{ij})^m (d_{ij})^2 - \lambda \left(\sum_{i=1}^C u_{ij} - 1 \right) \quad (2.9)$$

By letting the gradient of the Lagrange function to be zero, we may get two conditions:

$$(A) \quad \frac{\partial F_j}{\partial \lambda}(\lambda, \mathbf{u}_j) = \left(\sum_{i=1}^C u_{ij} - 1 \right) = 0 \quad (2.10)$$

$$(B) \quad \frac{\partial F_j}{\partial u_{st}}(\lambda, \mathbf{u}_j) = \left[m(u_{st})^{m-1} (d_{st})^2 - \lambda \right] = 0 \quad (2.11)$$

Hence, a condition (C) can be defined:

$$(C) \quad u_{st} = \left[\frac{\lambda}{m(d_{st})^2} \right]^{1/m-1} \quad (2.12)$$

Using condition (A), we get

$$\begin{aligned} \sum_{k=1}^C u_{kt} &= \sum_{k=1}^C \left(\frac{\lambda}{m} \right)^{1/m-1} \left[\frac{1}{(d_{kt})^2} \right]^{1/m-1} \\ &= \left(\frac{\lambda}{m} \right)^{1/m-1} \left\{ \sum_{k=1}^C \left[\frac{1}{(d_{kt})^2} \right]^{1/m-1} \right\} = 1 \end{aligned} \quad (2.13)$$

$$\text{Thus,} \quad \left(\frac{\lambda}{m} \right)^{1/m-1} = 1 / \sum_{k=1}^C \left(\frac{1}{(d_{kt})^2} \right)^{1/m-1} \quad (2.14)$$

From (2.13), we get

CHAPTER2. CLUSTERING AND OPTIMIZATION

$$\begin{aligned}
 u_{st} &= \left\{ 1 / \sum_{k=1}^C \left[\frac{1}{(d_{kt})^2} \right]^{1/m-1} \right\} \left[\frac{1}{(d_{kt})^2} \right]^{1/m-1} \\
 &= 1 / \sum_{k=1}^C \left[\frac{d_{st}}{d_{kt}} \right]^{2/m-1}
 \end{aligned} \tag{2.15}$$

The FCM algorithm is summarized as follows:

Begin

Initialize: $U = [u_{ij}]$ matrix satisfy the membership condition in Table 2.1, $U^{(0)}$

Iteration $k = 0$

Do

Calculate the center vectors $V^{(k)} = [v_j]$ with $U(k)$ using the Eqn. (2.4) & (2.5)

Update $U^{(k)}$, $U^{(k+1)}$ using Eq. (2.4)

Iteration $k = k+1$

While ($\|U^{(k+1)} - U^k\| > \varepsilon$)

End

where ε represents a relatively small number that defines the minimum margin improvement of membership value that must be satisfied before the algorithm converges or terminates.

Although FCM algorithm has advantages over the HCM algorithm, some weaknesses of the HCM algorithm still exist which they have never been improved by the FCM algorithm. For instances,

1. Although the convergence is guaranteed to a certain state, it is not guaranteed that the solution is a global optimal solution.
2. Same as the HCM algorithm, the FCM algorithm is sensitive to initialization that the solution is dependent on the pre-defined numbers of clusters.
3. The problems of noise and outlier still exist in the FCM algorithm.

CHAPTER 2. CLUSTERING AND OPTIMIZATION

2.2 Optimization Algorithms

Optimization is the process of determining the optimal points of a certain domain which is usually represented by an objective function. The task can be viewed as solving a problem which consists of traversing a space of potential solutions to find an optimum. The measure of solution quality is often used as the objective function in an optimization algorithm. Finding the optimal point(s) of a large search space is often an NP-hard problem or non-linear programming problem, for instance the well known traveling salesman problem (TSP) and aerodynamic design problem, which has an exponentially increasing search space as the problem scales up.

2.2.1 Non-Linear Programming Problem

The unconstrained non-linear minimization problem can be stated as follows:

$$\min_{\mathbf{x}} \{f(\mathbf{x}) \mid \mathbf{x} \in R^n\} \quad (2.16)$$

where $f(\mathbf{x})$ is the objective function to minimize and $\mathbf{x} \in R^n$ is the set of decision variables

$$\mathbf{x} = (x_1, x_2, \dots, x_n).$$

If $\mathbf{x}^*$ is a local minimum, the following relationship holds:

$$f(\mathbf{x}^*) \leq f(\mathbf{x}) \quad (2.17)$$

and

$$0 \leq \|\mathbf{x} - \mathbf{x}^*\| \leq \varepsilon \quad (2.18)$$

and

CHAPTER2. CLUSTERING AND OPTIMIZATION

$$\mathbf{x} \in R^n \quad (2.19)$$

where

$$\|\mathbf{x} - \mathbf{x}^*\| = \sqrt{\sum_{i=1}^n (x_i - x_i^*)^2} \quad (2.20)$$

This means that in the neighborhood of $\mathbf{x}^*$ defined by size ε , there does not exist any vector $\mathbf{x}$ whose $f(\mathbf{x})$ is smaller than $f(\mathbf{x}^*)$.

Basic structure of Local Search methods

The fundamental structure of local search techniques for solving unconstrained minimization problem is described in this section. The search begins with a random initialization. Subsequently, a direction of movement is prescribed according to some algorithm, and a line search or trust region approach is performed to determine an appropriate next step. The process is repeated at the new solution and the algorithm continues to run until a local minimum is found or the termination criteria are reached. Schematically, the local search method can be outlined as follows:

Begin**Initialize:**

Supply an initial guess $\mathbf{x}_0$;

Iteration $k=0$;

Repeat

Perform testing $\mathbf{x}_k$ for convergence;

Calculate search direction $\mathbf{p}_k$;

Determine an appropriate step length λ_k (or modified step $\mathbf{s}_k$)

Set $\mathbf{x}_{k+1} = \mathbf{x}_k + \lambda_k \mathbf{p}_k$ or $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{s}_k$;

Until (Convergence);

End

CHAPTER2. CLUSTERING AND OPTIMIZATION

Termination condition can be determined by many ways such as the accuracy referring to the precision of the final solution. A common stopping condition is used if the best solution is found while such solution does not get improved further by the algorithm running in a certain number of training steps.

Hill climbing strategies

Hill climbing strategies are a set of local search methods in which the choice of the neighbor solution is done by taking the one locally minimizing the optimization criterion. The local subset of the solution space can be considered as a hill. Traditionally, the strategies come under the general heading of hill climbing strategies as maximization methods. They are classified as:

1. Direct search method:

The most common strategies of direct search method are: coordinate strategies, pattern search strategy and DSC strategy. The Davies-Swann-Campey (DSC) strategy [51] was proposed in 1964. It was shown to be more effective than the pattern search strategy by Hooke and Jeeves. However, with the optimization problem of high dimensional space, costly orthogonalization procedure employed by DSC makes it less efficiency. Computational operations and storage requirement of DSC is of order $O(n^2)$. The most two famous types of DSC strategy are DSCG which employs the Gram and Schmidt orthogonalization procedure, and DSCP which employs the Palmer orthogonalization procedure.

CHAPTER2. CLUSTERING AND OPTIMIZATION

2. Gradient method:

Gradient method orientates their choice of search directions according to the first partial derivatives $\nabla F(\mathbf{x})$ of the objective function. The rate of convergence of gradient search is generally faster than that of direct search. However, the number of evaluations in gradient search is higher than that of direct search because the evaluation of additional n partial derivatives is required every iteration, where n representing the number of variables. One limitation of this method is that they can only be applied with differentiable functions, which are difficult to satisfy in real world situations.

The basic idea of gradient descent is to move the search direction toward negative gradient, similar to moving downhill in a function. In this way, the search moves towards the (local) minimum of the function. This method, however, may be stuck at some local optimum or saddle point where $\nabla F(\mathbf{x})$ is equal to 0.

3. Newton method:

Newton method makes use of the second partial derivatives $\nabla^2 F(\mathbf{x})$. There are several classes, namely, discrete Newton, quasi Newton, and truncated Newton. All Newton methods are based on approximating the objective function locally by a Taylor quadratic model. Basic Newton method conducts the search in the direction $-\left[\nabla^2 F(\mathbf{x}_k)\right]^{-1}\nabla F(\mathbf{x}_k)$, called Newton direction, where $\mathbf{x}_k$ is the current search point. The Newton strategy is often better than the direct and gradient searches in terms of convergent rate.

CHAPTER2. CLUSTERING AND OPTIMIZATION

2.2.2 The Optimal Solution and Local Minimum Issue

Local search algorithms have numerous advantages to solve the optimization problem. They are easy to implement and are guaranteed to obtain local optimal in a reasonable computational time. However, local search algorithms are typically ineffective for global optimization as the search may stop even if the solution found by the algorithm is not the global optimal. Moreover, these algorithms are sensitive to the initial estimates of the unknown if the criterion function has more than one optimum. In this case, the algorithms might converge to a local optimum instead of the desired global optimum. Figure 2.1 illustrates an example of error functional space with E_1 as local minimum and E_2 the global minimum.

Figure 2.1: Error surface –local and global minimum

It is impossible to analytically determine where the global minimum of the error surface is, and so the algorithm is essentially an exploration of the error surface. From an initially

CHAPTER2. CLUSTERING AND OPTIMIZATION

random configuration of the parameters (i.e., a random point on the error surface x_1 or x_2), the training algorithms incrementally seek for the global minimum. Typically, the gradient (or slope) of the error surface is calculated at the current point, and used to make a downhill move. Eventually, the algorithm would converge to the local minimum (E_1), or global minimum (E_2) depending strongly on the initialization starting at either x_1 or x_2 .

2.3 Evolutionary Computing

Evolutionary Computing (EC), inspired by Darwin's "survival of the fitness" law in the biological world, is efficient and broadly applicable for global optimization. Typical evolutionary algorithms include *differential evolution*, *evolutionary strategy*, *evolutionary programming*, *genetic algorithm* and *genetic programming*. They are similar in spirit, but different from the procedures of their implementation and the natures of problem to be applied.

ECs begin with a set of k randomly or heuristically generated candidate solutions, called population (where k is the population size). Each candidate solution is called an individual and may be encoded as a finite length string over some finite alphabet set and represents the parameters of the objective problem. For example, if we wish to optimize some function $F(\mathbf{x})$ where $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$, ($x_i \in R$), the parameters $(x_1, x_2, \dots, x_n)$ will be encoded as a finite length string. Fitness of each individual in the population is defined as some quantity that is related (normally proportional) to the objective value of that individual. A fitness function may be designed to return higher values for fitter individuals. EC work in an iterative manner called generations. In each generation, a new population is created from (or

CHAPTER2. CLUSTERING AND OPTIMIZATION

evolved from) the old population by means of simulated-biological operators, i.e., selection, crossover and mutation. Biological simulation is applied with the hope that new population contains better individuals than the old population.

2.3.1 Genetic Algorithm

Genetic Algorithm (GA) is the most commonly used algorithm of EC research. In GA, a population of individuals is initialized. This is followed by the selection, crossover and mutation operators to generate a new population. The skeleton of a GA is outlined as shown in Figure 2.3.

Figure 2.3: Flow chart of GA

One widely used stopping condition for GA is the maximum number of generations evolved. Other conditions such as time bound, maximum number of computation efforts (for example number of function evaluation) can also be used. Five issues are addressed when using GAs, these included defining the encoding scheme, selection strategy, crossover and mutation

CHAPTER2. CLUSTERING AND OPTIMIZATION

operators, and replacement strategy. The effectiveness of genetic algorithm in solving complex optimization and machine learning problem has been reported in [17] [33].

Encoding scheme

Defining an appropriate encoding scheme is the first issue of using genetic algorithm. Ideally, the encoding scheme should be flexible enough to represent all feasible solutions, but also restrictive enough to weed out any infeasible solutions. There are many encoding schemes such as binary encoding, real number encoding, tree encoding etc.

Selection strategy

It is used to select fitter individuals as parents to undergo recombination. Whatever selection method may be used, the fitter individuals are given a higher chance to be selected. Roulette wheel selection, tournament selection and rank-based selection are some of the commonly used selection methods. Competitive selection is used to expect that good parents can produce fitter offspring when they combined.

Crossover operator

Crossover is crucial for the success of a GA. It is probabilistically applied to generate offspring from the selected parents. Crossover is a genetic operator that splices information from two or more parents to form a composite offspring that has genetic material from all parents. It is expected that new offspring will be fitter than their parent. One point, two points and uniform crossover operations are widely used for continuous optimization. Partially

CHAPTER2. CLUSTERING AND OPTIMIZATION

matched crossover, order crossover, cycle crossover, edge recombination crossover and edge assembly crossover operations are commonly used as crossover techniques for combinatorial optimization.

Mutation

It is another genetic operator that is used to alter one or more genes in an individual. Mutation increases diversity of the population. This operator is applied with low probability for preventing GAs from stagnating at local optima.

Selection/Replacement strategy

This strategy determines the offspring that replace existing individuals to produce new population. Different variants of GAs may be distinguished by the use of different replacement schemes. For example, simple genetic algorithm creates an entirely new population of individuals. Steady-state genetic algorithm replaces only a specified portion of population in each generation. Incremental genetic algorithm merges only one offspring or two into the old population. Deme genetic algorithm, on the other hand, evolves multiple populations in parallel using a steady-state algorithm and migrates some of the individuals from each population to one of the other populations.

2.3.2 Memetic Algorithm

As mentioned earlier, although ECs often have a strong ability to converge to the global optimum, the convergence process is generally slow. Meanwhile, local search procedures can

CHAPTER2. CLUSTERING AND OPTIMIZATION

find the local optimum of a small region of the searching space, but they are typically poor in finding a global optimal. It is often a promising direction to combine global and local search in any search algorithm by considering the merits and deficiencies of each of them. These hybridizations are commonly known as memetic algorithms.

Memetic Algorithms (MAs) are evolutionary algorithms that apply a separate local search process to refine individuals (i.e. improve their fitness by hill-climbing). These methods are inspired by models of adaptation in natural systems that combine evolutionary adaptation of population of individuals with individual learning within a lifetime. MAs are inspired by Richard Dawkin's concept of meme [34], which represents a unit of evolution that can exhibit local refinement. Moscato [19] utilized this term to name his genetic algorithm associated with a local search operator performed by a simulated annealing algorithm. Under different contexts and situations, MAs are also known as hybrid ECs, genetic local searchers, Baldwinian EAs or Lamarckian EAs.

Unlike traditional EC's approaches, MAs are concerned with exploiting all available knowledge about the problem under study. This is not as an optional mechanism, but as a fundamental feature. From an optimization point of view, MAs are hybrid of ECs that combine global and local search by using an EC to perform exploitation while the local search method performs exploration. In single objective optimization, combining global and local search is a strategy used by many successful global optimization approaches, and the MAs have in fact been recognized as a powerful algorithmic paradigm for evolutionary computing. The characterization of a meme suggests that in culture evolution processes, information is not simply transmitting unaltered between individuals. In contrast, it is

CHAPTER2. CLUSTERING AND OPTIMIZATION

processed and enhanced by the communicating parts. This characteristic is accomplished in MAs by incorporating heuristics, approximate algorithms, local search techniques, specialized recombination operators, truncated exact methods, etc. Basically, most MAs can be regarded as a search strategy in which a population of optimizing agents cooperate and compete with each other. Figure 2.4 shows a flow diagram of a typical memetic algorithm.

Figure 2.4: Flow diagram of memetic algorithm

Chapter 3

Single-Objective Clustering

In the previous chapter, some traditional clustering methods as well as the associated optimization methods have been discussed. In this chapter, both Genetic Algorithm (GA) and Memetic Algorithm (MA) are employed to solve the single objective clustering problem. This chapter focuses on single objective clustering, where the solution is obtained by applying an optimization process to solving a problem with only one objective function.

3.1 Genetic Algorithm Based Single-Objective Clustering

The traditional clustering methods involve minimizing some kinds of objective function using the gradient based method. However, the method may encounter the problem of local optimality when inappropriate assumption is made for clustering the data set. In our study, GA based clustering is employed to obtain the global optimum solution [17]. The objective of the GA is to attain the minimum solution that gives optimal clustering centers on the data feature space. Figure 3.1 illustrates the workflow of the GA-based clustering processes. An initial population is first generated for further evolution. The evolution terminates when the fitness satisfies the stopping condition or the population has converged.

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

Figure 3.1: Workflow diagram of GA based clustering

Chromosome representation

The genetic algorithm may employ binary strings or real numbers to encode the population of potential solutions, often termed as chromosomes. The core merit of a binary representation lies in its simplicity and generality. It is straightforward to apply the classical crossover and mutation to binary strings. Although the encoding for binary representation is simple, the dimensionality involved is higher making the GA operations less efficient. Furthermore, different binary chromosomes may map to the same membership values, thus resulting in premature search convergence. To avoid such problems, real valued representations are considered in the present study on GA based clustering. Let N be the feature size and C the

CHAPTER 3. SINGLE-OBJECTIVE CLUSTERING

cluster size. The length of a chromosome is then $N \times C$. The solution of this clustering problem is a set of cluster centers, where each individual string is a sequence of real numbers representing a cluster center as:

$$Chromosome = \left\{ \underbrace{v_{11} \dots v_{1f}}_{\mathbf{v}_1} \underbrace{v_{21} \dots v_{2f}}_{\mathbf{v}_2} \dots \underbrace{v_{c1} \dots v_{cf}}_{\mathbf{v}_c} \right\}_{c \times f} \quad (3.1)$$

where $\mathbf{v}_j$, $j = 1 \dots C$, corresponds to a f -dimensional feature vector of cluster j .

Fitness Computation

The fitness computation process consists of two core phases. In the first phase, the chromosomes are decoded to obtain the corresponding cluster prototypes. The membership values and distances, d'_{ij} , of each data vector point $\mathbf{x}_i$, $i = 1 \dots N$ to the cluster prototypes $\mathbf{v}'_j$ are computed as

$$d'_{ij} = \sqrt{(\mathbf{x}_i - \mathbf{v}'_j)^T (\mathbf{x}_i - \mathbf{v}'_j)} \quad (3.2)$$

for $j = 1 \dots C$, and $i \neq j$.

Upon computing the distance measures, d'_{ij} , the membership values, u_{ij} , for each data vector point $\mathbf{x}_i$, are then computed using Eq. (2.4).

Selection and reproduction

In our GA, the rank-based roulette-wheel selection scheme is adopted. Chromosome of higher fitness value is ranked higher and thus has larger chance surviving. According to the fitness value of the chromosome, each chromosome in the population is associated with a

CHAPTER 3. SINGLE-OBJECTIVE CLUSTERING

sector in a virtual wheel proportional to the probability of it surviving to the next generation.

The probability of selecting a chromosome is defined as:

$$p_k = q_{\max} - \left(\frac{k-1}{n-1} \right) (q_{\max} - q_{\min}) \quad (3.3)$$

where k denotes the index of the chromosome and n is total number of chromosomes in the population. $q_{\max}$ and $q_{\min}$ are the upper-bound and lower-bound probabilities for the best and the worst chromosomes, respectively. These upper-bound and lower-bound probabilities must satisfy the following relationship:

Let the total probability of all the selected chromosomes be $P = \sum_1^n p_k = 1$, so:

$$\begin{aligned} \sum_1^n \left(q_{\max} - \left(\frac{k-1}{n-1} \right) (q_{\max} - q_{\min}) \right) &= 1 \\ \Rightarrow nq_{\max} - (q_{\max} - q_{\min}) \sum_1^n \left(\frac{k-1}{n-1} \right) &= 1 \\ \Rightarrow \frac{n}{2} (q_{\max} + q_{\min}) &= 1 \\ \Rightarrow q_{\max} + q_{\min} &= \frac{2}{n} \end{aligned} \quad (3.4)$$

From the above condition, the upper-bound and lower-bound probabilities are set as $q_{\min} = 0$

and $q_{\max} = 2/n$ respectively.

The chromosome selection process can be summarized by the following steps:

1. Calculate the accumulative probability AP_k of the k^{th} chromosome to be examined:

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

$$AP_k = \sum_{i=1}^k p_i \quad (3.5)$$

where p_i is the probability of the selected chromosome calculated using Eq. (3.3).

2. Generate a random number r within $[0, 1]$
3. Select the k^{th} chromosome such that $AP_{k-1} < r < AP_k$

Subsequently the reproduction process proceeds to generate new offspring using the crossover and mutation operations. The crossover operation generates new offspring from two parent chromosomes before it is perturbed using the mutation operator. The details of the two reproduction operations are described in what follows:

Crossover and Mutation

Crossover is a probabilistic process that exchanges information between two parent chromosomes for generating two corresponding offspring. The crossover probability threshold pC defines the degree or amount of crossover to take place in each generation. At each iteration, arithmetic crossover is performed if a randomly generated number r , *i.e.*, of range $[0, 1]$, is found to be larger than pC . In arithmetic crossover, two offspring $\mathbf{x}'$ and $\mathbf{y}'$ are generated as $\mathbf{x}' = \lambda_1 \mathbf{x} + \lambda_2 \mathbf{y}$ and $\mathbf{y}' = \lambda_1 \mathbf{y} + \lambda_2 \mathbf{x}$, respectively, where λ_1, λ_2 represent some scalar values. To ensure that the generated offspring are likely to remain in the feasible search region, convex crossover is used with $\lambda_1 + \lambda_2 = 1$ and $\lambda_1, \lambda_2 > 0$. In our GA, $\lambda_1 = \lambda_2 = 1/2$ is defined.

Mutation introduces variations in the chromosomes. It provides a global searching capability for the GA through altering the chromosomes probabilistically. Each chromosome

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

undergoes mutation with a fixed probability of p_M . An arithmetic mutation operation used and described as follows:

Let $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_k \ \dots \ x_m]$ be the original individual, $\mathbf{x}' = [x_1 \ x_2 \ \dots \ x_k' \ \dots \ x_m]$ is the mutated offspring from $\mathbf{x}$. The value of x_k' can be derived either by $x_k' = x_k + r(b_k - x_k)\left(1 - \frac{t}{T}\right)^b$ or $x_k' = x_k - r(x_k - a_k)\left(1 - \frac{t}{T}\right)^b$, where a_k and b_k are the upper and lower bounds respectively. r denotes a uniform random number that lies within $[0, 1]$, t denotes the current generation number, T is the maximum number of generations, and b is the degree of non-uniformity. The degree of non-uniformity defines the degree of perturbation by the mutation operator across generations. This is an important strategy as we expect a higher degree of perturbation at the initial generations, which reduces with higher search generations. At $b = 1$, the degree of perturbation varies linearly. Here b is set to 2 for a reasonable degree of expansion to avoid any overly destructive effects of mutation.

Termination criterion

The algorithm terminates when either the maximum number of generations is reached or the change in fitness value over one generation is lower than the user-specified threshold value. Here the threshold value is set at 0.001% of the fitness value.

3.2 Memetic Algorithm Based Single-Objective Clustering

Similar to GA-FCM, MA-FCM is expected to outperform the traditional clustering methods with the ability of converging to the global optimum. One of the main steps in GA algorithm

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

is to evaluate the fitness of the current population which can be computationally expensive. Hence, the convergence speed of GA is dependent upon the complexity of the data set. Memetic algorithm (MA) is a hybridization of global search and local search strategies where, with global search phase, the ability of reaching the global optimal solution is preserved, and at the same time, with local search phase, the speed convergence process is enhanced. Figure 3.2 illustrate the work diagram of the MA-FCM algorithm.

3.2.1 Local Search for Memetic Algorithm

Although local search helps the search to converge quickly to local optimum, it consumes many computational steps which can be computationally expensive. Hence it may not be the best efficient to apply local search directly on all individuals in the GA population. Here, we define a fraction of the population that will be further subjected to local search refinement. To control when and how the local search should be applied, two parameters, namely, tLS and pLS are defined. tLS defines how long the local search is applied while pLS denotes the probability of applying local search. A general local search algorithm is outlined as follows:

```

Begin
  //Check condition: if current generation is applied local search
  If ( Mod( Generation-counter,  $tLS$  ) == 0 ) then
    For each child  $i$ 
      Generate a random number  $r$  of the range [0, 1]
      If (  $r < pLS$  ) then
        Apply local search
      End If
    End For
  End If
End

```

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

Figure 3.2: MA-FCM algorithm workflow

3.3 Cluster Validity of Clustering Method

There exist many potential methods to evaluate the performance of clustering methods on complex data sets. Quantitative measures are used to indicate how accurate the result represents the structure of the data set. This is one of the most important yet complicated issues of the clustering problem. Initially, the clustering method requires the number of

CHAPTER 3. SINGLE-OBJECTIVE CLUSTERING

clusters to be pre-defined which is one of the important problems when *a priori* knowledge of the data set is unavailable. An optimization algorithm is required to determine the optimal number of clusters for any data that maximizes the performance of the clustering process. The performance measures for clustering process are referred to as validity indices and described here as follows:

Partition coefficient [2]:

This index is defined as

$$PC(c) = \frac{1}{n} \sum_{i=1}^c \sum_{j=1}^n u_{ij}^2 \quad (3.6)$$

where $1/c \leq PC(c) \leq 1$. The optimal number of clusters is subjected to the condition $\max_{2 \leq c \leq n-1} PC(c)$. A larger value of $PC(c)$ indicates better clustering performance.

The PE [3, 4]:

This index is defined as

$$PE(c) = -\frac{1}{n} \sum_{i=1}^c \sum_{j=1}^n u_{ij} \log_2 u_{ij} \quad (3.7)$$

where $0 \leq PE(c) \leq \log_2 c$. The optimal number of clusters is found by solving $\min_{2 \leq c \leq n-1} PE(c)$. There is a special case when $u_{ij} = 1/c$ for all i and j that makes $PC(c) = 1/c$ and $PE(c) = \log_2 c$ [11].

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

The FS [12]:

This index is defined as

$$\begin{aligned} FS(c) &= \sum_{i=1}^c \sum_{j=1}^n u_{ij}^m \|\mathbf{x}_j - \mathbf{v}_i\|^2 - \sum_{i=1}^c \sum_{j=1}^n u_{ij}^m \|\mathbf{v}_i - \bar{\mathbf{v}}\|^2 \\ &= J(\mathbf{U}, \mathbf{v}) + K(\mathbf{U}, \mathbf{v}) \end{aligned} \quad (3.8)$$

where $\bar{\mathbf{v}} = 1/c \sum_{i=1}^c \mathbf{v}_i$. The term $J(\mathbf{U}, \mathbf{v})$ is clustering objective function which measures the compactness of the data structure and $K(\mathbf{U}, \mathbf{v})$ measures the separation. The optimal value of c is found by solving $\min_{2 \leq c \leq n-1} FS(c)$.

The XB [13, 14]:

This index is defined for $m = 2$

$$\begin{aligned} XB(c) &= \frac{\sum_{i=1}^c \sum_{j=1}^n u_{ij}^m \|\mathbf{x}_j - \mathbf{v}_i\|^2}{n \min_{i,j} \|\mathbf{v}_i - \mathbf{v}_j\|} \\ &= \frac{J(\mathbf{U}, \mathbf{v})/n}{Sep(a)} \end{aligned} \quad (3.9)$$

similarly, $J(\mathbf{U}, \mathbf{v})$ represents the compactness while $Sep(a) = \min_{i,j} \|\mathbf{v}_i - \mathbf{v}_j\|$ is a cluster separation measure. The optimal value of c is found by solving $\min_{2 \leq c \leq n-1} XB(c)$.

These four indices are originally defined as cluster validity for fuzzy C-means clustering method. They determine the compactness of each cluster and separation between clusters. Since each index provides a unique measure of the clustering performance, improved solution may be attained by combining and considering all four indices instead of using only one, since it is often not known *a priori* which index would fit the dataset set best.

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

3.4 Experimental Results and Discussion

3.4.1 Data Sets and Preprocessing

In this section, experiments were taken to examine the performances in terms of clustering validity as well as classification accuracy. The performances of clustering methods were validated using a synthetic dataset, namely TEST-9 and three real data sets, namely IRIS, GLASS, and VOWEL. The datasets differ in terms of sample size, feature size and optimal cluster size. Table 3.1 tabulates the details of these four data sets.

Data set	# Samples n	# Features f	# Clusters c
TEST9	900	2	9
IRIS	150	4	3
GLASS	214	9	6
VOWEL	990	10	11

Table 3.1: Details of the experimental datasets

Figure 3.3: TEST-9 dataset without noise (left) and with noise (right)

TEST-9 data set is an artificially generated two-dimensional data set with a sample size of 900. The data set contains nine clusters and each cluster contains 100 points. Data points

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

in each group are uniformly generated from each cluster. Two sets of data (with and without noise) were generated as shown in Figure 3.3. IRIS dataset is commonly used for benchmarking and it consists of three separate clusters with two clusters having slight overlapping. There is also no noise in the IRIS data. GLASS data set has a similar number of samples as the IRIS dataset but has twice the cluster size with smaller members. A practical observation shows that there are outliers in GLASS data set. VOWEL data set consists of a relatively larger number of both samples and feature sizes. There are overlapping clusters and different cluster sizes in VOWEL which makes optimal clustering hard to achieve.

For an f -dimensional data set, each dimension refers to one characteristic of the data and the degree of dispersions for each characteristic is generally different. The result of clustering to minimize the total dispersion measure will likely discard the effects of some characteristics. Those features with high dispersion will dominate those with small degree and hence the obtained result may be biased. To use all the features sufficiently in the clustering process, data normalization is essential to equalize the dispersions of each feature. Hence, all dataset $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n\}$ and data point $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{if})$, data normalization is applied to scale x_{ij} to x'_{ij} :

$$x'_{ij} = \frac{x_{ij} - \sum_{k=1}^n x_{kj} / n}{\sqrt{\sum_{k=1}^n \left(x_{kj} - \sum_{k=1}^n x_{kj} / n \right)^2 / (n-1)}} \quad (3.10)$$

with $k = 1, \dots, f$ and $i = 1, \dots, n$

CHAPTER 3. SINGLE-OBJECTIVE CLUSTERING

After normalization, each feature of the data would have the same sample mean and dispersion measure (i.e., its variance).

3.4.2 Clustering Validation

The following experiments were conducted to examine the cluster validation method with both FCM and GA-FCM. Those indices as mentioned in the previous part of this thesis are used to define the optimal number of clusters and describe the structure of the data sets.

Cluster Validity on synthetic dataset (TEST-9)

We generated the validity indices stated as PC, PE, FS and XB from Eqns (3.6-3.9) respectively, to evaluate the performance of FCM and GA-based FCM clustering algorithms. All of them are validated using the synthetic TEST-9 dataset and the obtained results are tabulated in Table 3.2.

According to the results shown in Table 3.2, the optimal number of clusters is 9. FCM performed well on this validation for since all indices show an optimal cluster size of $c=9$. On the other hand, the proposed GA-FCM algorithm also converged to $c=9$. Hence, it is observed that GA-FCM is able to maintain good clustering performance on noiseless dataset.

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

c	FCM				GA-FCM			
	PC	PE	FS	XB	PC	PE	FS	XB
3	0.61	0.98	-246.29	0.14	0.53	1.17	-173.21	0.26
4	0.59	1.12	-701.51	0.08	0.55	1.24	-930.24	0.07
5	0.59	1.20	-777.69	0.14	0.42	1.70	-159.76	0.21
6	0.61	1.21	-856.74	0.09	0.50	1.56	-647.60	0.13
7	0.66	1.13	-1063.66	0.08	0.55	1.47	-892.47	0.09
8	0.68	1.11	-1044.86	0.06	0.56	1.46	-772.58	0.08
9	0.74	0.96	-1276.32	0.03	0.64	1.28	-1033.17	0.04
10	0.70	1.08	-1234.88	0.39	0.58	1.45	-975.89	794.13
11	0.67	1.20	-1193.57	0.52	0.53	1.67	-890.76	794.57
12	0.65	1.24	-1173.29	0.34	0.53	1.64	-895.40	2.27
13	0.63	1.33	-1155.56	0.44	0.51	1.70	-937.86	1.75
14	0.61	1.42	-1095.23	0.45	0.47	1.86	-844.03	383.82
15	0.60	1.47	-1063.42	0.42	0.43	2.05	-754.65	25597.13

Table 3.2: Clustering validity by TEST-9 dataset under noise-free condition

c	FCM				GA-FCM				
	PC	PE	FS	XB	P-PC	PC	PE	FS	XB
3	0.62	0.96	-348.26	0.14	0.29	0.37	1.50	684.95	325.71
4	0.59	1.14	-812.41	0.09	0.25	0.54	1.27	-1051.82	0.07
5	0.59	1.20	-976.13	0.11	0.28	0.51	1.43	-911.92	0.10
6	0.59	1.26	-1091.07	0.12	0.29	0.50	1.54	-944.79	0.15
7	0.63	1.23	-1291.01	0.09	0.27	0.51	1.58	-989.64	0.11
8	0.66	1.18	-1343.47	0.06	0.24	0.54	1.54	-1072.81	0.07
9	0.61	1.35	-1166.89	0.32	0.20	0.58	1.44	-1177.70	0.05
10	0.65	1.23	-1428.66	0.19	0.22	0.52	1.65	-1061.29	1105.19
11	0.57	1.52	-1219.27	0.50	0.23	0.48	1.76	-1014.01	1909.19
12	0.61	1.40	-1347.99	0.32	0.24	0.46	1.87	-992.09	1119.26
13	0.59	1.47	-1317.04	0.30	0.25	0.42	1.99	-904.75	17394.88
14	0.58	1.52	-1293.40	0.29	0.27	0.40	2.11	-854.46	3862.74
15	0.56	1.62	-1238.06	0.30	0.27	0.39	2.16	-844.88	4625.61

Table 3.3: Clustering validity by TEST-9 dataset under noise condition

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

The alternative validation for this dataset would be conducted under noisy condition. Two hundred noisy points with uniform distribution were added to corrupt the original dataset as shown on the right-hand side of Figure 3.3. The noisy data points are distributed uniformly in the ranges between [0, 5] along x-axis and between [0, 10] along y-axis. The validation results are tabulated in Table 3.3. As shown in the table, the clustering validities based on the four indices are unable to arrive at the optimal solution when using the FCM clustering method on the noisy data set. On the other hand, GA-FCM performs well to generate consistent validation results under noisy condition.

Cluster Validity on IRIS, GLASS and VOWEL datasets

In this section, the performance of various clustering methods using the validity indices is examined. According to the cluster validity results obtained from the TEST-9 data set, it is shown that both PC and XB indexes were found to arrive at the optimal number of clusters more often than the other indices considered. In the rest of the simulations involving the IRIS, GLASS and VOWEL datasets, PC and XB indexes would be employed to evaluate the performance of the clustering algorithms.

c	FCM						GA-FCM					
	m=2		m=2.5		m=3		m=2		m=2.5		m=3	
	PC	XB	PC	XB	PC	XB	PC	XB	PC	XB	PC	XB
2	0.834	0.113	0.731	0.092	0.661	0.071	0.789	0.120	0.667	0.105	0.599	0.089
3	0.707	0.222	0.572	0.172	0.490	0.126	0.608	0.472	0.474	0.436	0.411	0.357
4	0.615	0.314	0.468	0.228	0.385	0.162	0.492	0.698	0.400	0.375	0.334	0.264
5	0.539	0.554	0.416	0.163	0.330	0.102	0.457	0.479	0.301	0.847	0.254	0.533
6	0.512	0.434	0.366	0.274	0.268	0.212	0.389	0.879	0.272	0.608	0.215	0.794

Table 3.4: Cluster validity on IRIS dataset

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

The clustering validity results obtained on the IRIS dataset are reported in Table 3.4. The validity indices were computed with various exponents. As shown in the table, both FCM and GA-based FCM clustering algorithms were able to attain the optimum number of clusters, i.e., at $c=2$, for various values of m . It is illustrated that the characteristic of IRIS dataset with two overlapping clusters indicated only slight difference between data samples belonging to those clusters. Overall, both values of $c=2$ and 3 were regarded as acceptable for modeling the structure of the IRIS data.

On the contrary, because of the noise added into the GLASS dataset, the FCM clustering algorithm failed to obtain the desired result optimal cluster size. On the other hand, the proposed GA-based FCM still remain capable of converging to the optimal cluster size of 6, as shown in Table 3.5. This highlights that FCM does not perform well on noisy dataset while GA-based FCM method performed well consistently.

Table 3.6 presents the cluster validity for the VOWEL dataset. Both FCM and GA-FCM clustering methods failed on this validity test. The optimal number of clusters obtained by both FCM and GA-FCM is 5 with various settings of exponents. It is difficult to match the optimal number of clusters at 11 since the VOWEL dataset seems to be relatively unstructured clustering such that the FCM clustering method is unable to model well.

c	FCM						GA-based FCM					
	m=2			m=2.5			m=3			m=2		
	PC	XB		PC	XB		PC	XB		PC	XB	
5	0.2	11918851		0.2	1.5E+08		0.2	2.4E+08		0.2	3.54E+08	
6	0.167	78958768		0.167	2.04E+08		0.167	4018570		0.167	3.93E+09	
7	0.143	1.16E+09		0.143	81543451		0.143	1.7E+08		0.143	7.33E+08	
8	0.125	9.9E+08		0.125	8.97E+08		0.125	1.5E+07		0.125	1.08E+11	
9	0.111	2.17E+10		0.111	1.56E+08		0.111	1.9E+07		0.111	3.4E+11	
10	0.1	5.86E+09		0.1	85812518		0.1	1111155		0.1	1.33E+10	
11	0.091	5.74E+08		0.091	98675102		0.091	4592583		0.091	1.79E+09	
12	0.083	36261631		0.083	54506538		0.083	1144505		0.083	1.89E+12	
13	0.077	1.82E+08		0.077	13853304		0.077	3204681		0.077	1.24E+11	
14	0.071	8.56E+08		0.071	47946515		0.071	2375654		0.071	3.76E+10	

Table 3.6: Cluster validity on VOWEL data set

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

3.4.3 Performance of the Single-Objective Clustering Algorithms

Before performing the training process for the GA-FCM and MA-FCM clustering algorithms, the parameter settings which have determined experimentally are reported in Table 3.7.

Data set	Pop size	pC	pM	nEva	LocalSearch
TEST9	30	0.7	0.03	50000	DSCG
IRIS	30	0.7	0.03	30000	DSCG
GLASS	30	0.7	0.03	100000	DSCG
VOWEL	30	0.7	0.03	100000	DSCG

Table 3.7: Parameter settings for GA-FCM and MA-FCM

In the MA-based FCM, the population is initialized with an initial set of population with high diversity over the search space. Hence, the exploring feature of MA indicated by mutation probability is set to be rather small at 0.03, while crossover probability is set higher at 0.7 to increase convergence speed. Since the data structure of GLASS and VOWEL data sets is complex, the number of evaluations is set to be relatively higher at 100,000 while the other data sets at 50,000, in order to ensure the algorithm converges to near global optimum. The local search method used in the MA-based FCM is DSCG as previous introduced in the thesis.

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

Clustering performance with local optimal problem

The TEST-9 dataset is firstly used to show the clustering performance with the problem of local optimal. The TEST9 data set consists of 9 separated clusters and noises on the left half of the space. Figure 3.4 shows the solutions obtained by the MA-FCM algorithm. Figure 3.4(a) shows the global optimal solution for the data set with the desired center's position and with the lowest value of the objective function. The other three graphs (i.e., Figure 3.4(b), (c) and (d)) show the results of the three local optimal solutions with center's positions moved due to the added noise.

Figure 3.5 shows the results of clustering on the IRIS data set. The result on the left-hand side shows the global optimal case, whereas the result on the right-hand side shows another clustering solution with different center's positions.

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

Figure 3.4: Global optimal and local optimal solutions of TEST9 data set (a) optimal solution (b-c-d) local optimal solutions

CHAPTER 3. SINGLE-OBJECTIVE CLUSTERING

Figure 3.5: Global optimal and local optimal solutions of IRIS data set

Benchmarking results

For validation, TEST9 data set consists of a large number of samples, whereas GLASS and VOWEL data sets have larger number of clusters, feature sizes and samples that make the chromosome length longer and requires demanding computations. The length of the chromosome and the complexity of the search space is affected strongly by the performance of global search in terms of convergence speed. This issue challenges the performance of GA-FCM and MA-FCM, since success rate and computational time are contradictory. All the computations were performed under a Pentium-4 3.0GHz with 1GB RAM of personal computer. The success rate which represents the rate of reaching the global optimal result may be improved with a larger number of evaluation settings, which leads to more expensive computation. On the other hand, with a limited number of evaluation settings, the method may not be able to reach the global solution with random initialization. The trade-off

CHAPTER 3. SINGLE-OBJECTIVE CLUSTERING

between convergence speed and success rate can be adjusted in MA-FCM with the aid of local search. The employed local search boosts up the convergence speed of the algorithm. Figure 3.6 shows the performance of the MA-FCM algorithm benchmarking against both the traditional FCM and the GA-FMC clustering algorithms under different consideration factors.

As shown in Figure 3.6(a), the success rates of the three clustering methods are dependent on the data sets used. The clustering methods always perform well on simple data sets, e.g. TEST9 with small dimension and IRIS with simple structure. It is observed that GA-FCM performs worse with the GLASS and VOWEL data sets than the other two methods. The reason mainly is because of the limitation of GA, which requires substantial computational resource. Hence, with limited number of evaluations, GA might not yet be able to reach the global optimal solution. However for simple data structure of IRIS, GA shows higher success rate which indicates the algorithm has converged within the limited setting. Figure 3.6(b) shows the average time, which is needed for the method to reach the best solution, i.e., in terms of fitness over the population. It shows that traditional FCM takes a shorter time to obtain the best solution while MA-FCM takes longer time and GA-FCM takes time even longer. The reason for significant convergence time of traditional FCM is because the method uses purely local search, whereas the both GA-FCM and MA-FCM are global stochastic search methods that require more computational time. Generally, MA-FCM has less convergence time on the average as compared to GA-FCM. However, there is a significant difference between GA-FCM and MA-FCM when they are applied to VOWEL data set. The reason for worse convergence time of GA-FCM is the effect of both

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

chromosome length and search space complexity of VOWEL data set. They also affect the performance of MA-FCM as it can be observed from the Figure 3.6(a) that the success rate of MA-FCM is smaller than that of the traditional FCM.

To sum up, the characteristic of the data set affects strongly with the performance of the methods. For simple and low dimensional data sets such as TEST9 and IRIS, all the methods perform well with high success rate and small amount of computation time. For the data sets with a large number of sample data such as TEST9 and VOWEL, GA-FCM and MA-FCM may face the problem of long convergence time. However, for a large number of features and clusters e.g. VOWEL, the chromosome length is large and this affects the performance of the GA-FCM significantly. Overall, MA-FCM performs better and outperforms both traditional and GA based FCM clustering algorithms.

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

Figure 3.6: FCM, GA-FCM and MA-FCM performances (a) Success rate, (b) Average time, (c) Success rate vs. number of features, (d) Average time vs. number of samples

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

Convergence Performance of GA-FCM vs. MA -FCM

In this section, a comparison between GA-FCM and MA-FCM in terms of convergence speed is conducted. Figure 3.7 shows the convergence curves of both methods applied to four different data sets. The figures show the log scale of the fitness values over the number of evaluations performed. It can be observed from the figures that the curves of GA-FCM tend to converge steadily, while MA-FCM achieves improvement in fitness values implied by the local search phase. The local search operator has tremendous performance in simple data sets of TEST9 and IRIS with sharp curves indicated convergence obtained less than 2000 evaluations. The local search operation in the experiments with these two data sets performs very well with significant improvement.

For more complex data sets, the MA-FCM also encounters the problem of large chromosome length and search space. The curves show no difference in GLASS data between MA-FCM and GA-FCM. The flat horizontal line portion of the curve shows the local search phase which takes lots of evaluation due to high dimensional problem. This feature can be seen clearly on the last case of VOWEL data set while the local search phase takes large number of evaluations for small improvement of fitness. The effect of a large chromosome shows significant difference in the two curves, which shows GA-FCM in turn has better performance compared to MA-FCM. Since a long chromosome affects the performance of the local search which takes a lot of evaluations per iteration, it makes the performance of MA-FCM even worse than GA-FCM.

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

Figure 3.7: Convergence comparison of GA-FCM and MA-FCM (a) TEST9 data (b) IRIS data (c) GLASS data and (d) VOWEL data

3.5 Concluding Remarks

In this chapter, a MA-based FCM clustering algorithm is introduced for clustering. It serves to replace the traditional gradient search in FCM through the use of evolutionary search. In

CHAPTER3. SINGLE-OBJECTIVE CLUSTERING

MA-FCM the clustering problem is formulated as an optimization problem. MA-FCM overcomes the problem of local optimum with ability of reaching the global optimal solution using evolutionary search. Experiments were conducted to show some cases where FCM get stuck at local optimal solution while MA-FCM converges to the global optimal solution.

The experiments conducted with real data sets included IRIS, GLASS and VOWEL datasets highlighted the high success rate obtained by GA-FCM and MA-FCM algorithms. With evolutionary search, the GA-FCM improves the rate of reaching the global optimal solution compared to traditional FCM. However, one major drawback of GA-FCM lies is the size of chromosome length which requires a significant amount of computational resources to process. There is a need to improve the scalability of GA-FCM in order to process complex large data set at tractable computational time.

MA-FCM is presented as an improvement of GA-FCM which inherits the evolutionary search feature and employs local search simultaneously. Since MA-FCM improves the convergence speed compared to GA-FCM, however, in most of the cases, the computational time of MA-FCM can sometimes be higher than that of the traditional FCM. Thus, the appropriate design issue on the combination of global and local search represents an important issue of MA-FCM that would warrant further research.

Chapter 4

Multiple-Objective Clustering

In the previous chapter, genetic algorithm based and memetic algorithm based clustering methods were introduced and compared with the traditional clustering methods. It has shown the proposed evolutionary clustering methods were able to attain good performances in terms finding the global optimal solution or optimal cluster size at higher success rate. In this chapter, we will further develop an approach to improve the “unsupervised” feature of clustering. Previous methods have attempted to conduct clustering using a single objective approach. The objective function is usually selected based on some prior knowledge about the data set, such as cluster structure or number of clusters to use.

In this chapter, a novel clustering method which takes into consideration more than one objective function for optimization is proposed. The method employs a multi-objective memetic algorithm approach as an extension of the single-objective memetic algorithm to solve an optimization problem pertaining to more than one objective function. The method is expected to describe the clustering structure of complex data sets more precisely when *no a priori* knowledge of the data is available. In the later part of this chapter, experimental results are demonstrated to illustrate the advantages and disadvantages of the new methodology and compared to previous methods using different types of data sets.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

In this research, we employ multi-objective memetic algorithm, called MOMA, to solve a clustering problem. However, we are not going to focus on the advanced methodology of MOMA, but rather on analyzing the clustering results with a simple MOMA configuration.

4.1 Multi-Objective Clustering Problem

Most existing clustering methods attempt to optimize just one property of the clustering structure and usually make a priori assumption on the type of clusters which can later be identified. This binding to a particular clustering criterion is one of the reasons for the failure to describe the clustering structure of random data set where the criterion employed is inappropriate. Furthermore, real clustering problem usually involves data with more than one cluster structure. While refining the cluster structure with respect to pre-defined criterion, the method fails to detect the other structures.

A multi-objective optimization problem (MOOP) deals with more than one objective function and involves simultaneous optimization of several possibly conflicting objectives. In a single-objective optimization, the optimal solution is usually clearly defined. It does not hold for multi-objective optimization problems. Instead of a single optimum, there is a set of alternative trade-offs, generally known as Pareto-optimal solutions. These solutions are optimal in the sense that no other solution in the search space is considered totally better when all objectives are considered. The various conventional clustering methods mainly distinguish from each other by the objectives that they want to optimize. However, while several valid properties may be used to describe a good clustering, they may be partially conflicting and/or inappropriate in a particular context.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

There are several valid properties that may be ascribed to a good partitioning, but these are partly in conflict and are generally difficult to express by objective functions. Despite this, existing clustering criteria/algorithms do fit broadly into three fundamental categories:

Compactness: This concept is generally implemented by keeping intra-cluster variations small. This category includes algorithms like K-means [46], average link agglomerative clustering [47] or model-based clustering approaches [48]. The resulting methods tend to be very effective for spherical or well-separated clusters, but they may fail to detect more complicated cluster structures [49], [50], [14], [15].

Connectedness: This is a more local concept of clustering based on the idea that neighboring data items should share the same cluster. Algorithms implementing by this principle are density-based methods [35], [36] and other methods such as single link agglomerative clustering [37]. They are well-suited for the detection of arbitrarily shaped clusters, but lack of robustness when there is little spatial separation between clusters.

Spatial separation: Spatial separation on its own is a criterion that gives little guidance during the clustering process and can easily lead to trivial solutions. It is therefore usually combined with other objectives, most notably measures of compactness or balance of cluster sizes. The resulting clustering objectives can be tackled by general-purpose meta-heuristics (such as simulated annealing, tabu search and evolutionary algorithms [38, 39]).

Figure 4.1: Clustering criteria measures
- 62 -

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

Examples of data sets exhibiting compactness, connectedness and spatial separation, respectively, are shown in Figure 4.1. Clearly, connectedness and spatial separation are related concepts. In principle, the cluster structure in the data sets B and C can be identified by a clustering algorithm based on either connectedness or on spatial separation, but not by one based on compactness.

In order to develop a clustering algorithm that simultaneously considers several complementary aspects of clustering quality, our approaches make use of Multi-objective optimization. Specifically, we employ the MOMA package to optimize several available and selective clustering measures as objectives to obtain a set of trade-off solutions which represent a good approximation of the Pareto front.

4.2 Related Works

The most closely related work is reported in [30]. This work presented a multi-objective evolutionary algorithm, called MOCK (Multi-Objective Clustering with automatic K-determination) that is able to simultaneously optimize two complementary objectives: overall deviation and connectivity. MOCK returns a set of different trade-off partitions over a range of different cluster numbers and automatically determines the best ones.

Another interesting work is reported in [31]. The authors employed several clustering algorithms to find different clusters and then combine these results to form a partition. In this approach, each cluster can be derived from different algorithms. The clusters of the resultant partition are selected with an independent goodness function based on the stability of the

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

clusters under the re-sampling of the dataset. By doing so, their algorithm chooses the best objective function for different parts of the feature space.

On the other hand, the work in [32] is one of the most popular approaches for cluster ensemble. They formalized the cluster ensemble problem as a combinatorial optimization problem in terms of shared mutual information. Then, to tackle the combinatorial complexity of the problem, they proposed three algorithms (or consensus functions): CSPA (Cluster-based Similarity Partitioning Algorithm), HGPA (Hyper-Graph Partitioning Algorithm) and MCLA (Meta-Clustering Algorithm).

Another cluster ensemble method based on graph partitioning was proposed in [40]. It constructs a bipartite graph from a set of partitions to be combined, modeling objects and clusters simultaneously as vertices, and later partitioning the graph by a traditional graph partitioning technique.

4.3 Multi-Objective Memetic Algorithm for Clustering (MOMACL)

The Multi-Objective Memetic Algorithm (MOMA) is a hybrid algorithm combining Multi-Objective Genetic Algorithm (MOGA) with Local Search (LS) algorithm. The MOGA is employed by the fast Non-dominated Sorting Genetic Algorithm (NSGA-II) [42] [43], which has been shown to outperform other methods in terms of convergence speed and diversity of solution set. NSGA-II is a revised version of the original NSGA which maintains the solutions of the best front, including them into the next generation. The rank of each individual is based on the concept of the level of non-domination corresponding to the

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

NSGA. Moreover, this algorithm introduces a faster processing to sort the population that takes m computations, where m is the number of objectives and N is the number of population members.

Traditional LS algorithms can only optimize single objective optimization problems, whereas the optimum of multi-objective optimization problem is a set of solutions which is termed a Pareto-optimal set. Some specifications are needed while we apply the LS algorithm into MOMA algorithm. The objective of applying the local search operator is to improve the convergence speed of Pareto solutions to the global Pareto front. So, after a new offspring population is created by genetic operations, we choose to apply local search to this new population to improve the “quality” of these new-born individuals. Improved offsprings are then combined with individuals in the parent population to create a mixed population. Figure 4.2 shows the pipeline of the improvement on offspring population. This mixed population is then resized by Pareto-dominant ranking to obtain the next population.

Figure 4.2: Applying local search improvement on offspring population

Instead of finding a single clustering solution, clustering can also be considered as a multi-objective optimization problem. The aim of the multi-objective clustering is to find an approximation to the Pareto front, which is the set of partitioning that are Pareto optimal with respect to the optimized objective function. Here, two objective functions are to be minimized and the approximation set is represented by the corresponding attainment surface.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

This is the boundary in the objective space separating those points that are dominated by or equal to at least one of the data points in the approximation set.

4.3.1 Objective Functions

In the proposed MOMA based clustering algorithm, we select the complementary criteria that reflects different aspects of a good clustering solution. Two types of complementary objectives are selected. They are connectedness and compactness, as the concept of connectedness is also applicable for data set with spatial separation features.

- 1 **Compactness:** In order to express compactness, we compute the overall deviation of a partitioning. This is simply computed as the overall summation of distances between data items and their corresponding cluster center:

$$Dev(c) = \sum_{i=1}^c \sum_{j=1}^n d(\mathbf{x}_j, \mathbf{v}_i) = \sum_{i=1}^c \sum_{j=1}^n \|\mathbf{x}_j - \mathbf{v}_i\|, \quad (4.1)$$

where c is the set of all clusters, $\mathbf{v}_i$ is the centroid of cluster i and $d(\mathbf{x}_j, \mathbf{v}_i)$ is the distance function.

- 2 **Connectedness:** the measure used is connectivity which evaluates the degree to which neighboring data-points have been placed in the same cluster. It is computed as:

$$Conn(c) = \sum_{j=1}^N \left(\sum_{k=1}^L x_{j,nn_j(k)} \right), \quad (4.2)$$

$$\text{where } x_{r,s} = \begin{cases} 0 & \text{if } \exists C_t : r \in C_t \wedge s \in C_t \\ 1/k & \text{otherwise} \end{cases} \quad (4.3)$$

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

$nn_j(k)$ is the k^{th} nearest neighbor of data sample j , and L is a parameter determining the number of neighbors that contribute to the connectivity measure. As the objective captures local densities to be minimized, it can detect arbitrarily shaped clusters such as shell shape. However, it is not robust toward clusters that are overlapped.

4.3.2 Global Search in MOMA

In the proposed MOMA algorithm, NSGA-II algorithm is employed to perform global search while local search helps to accelerate the convergence to a local optimum within a searching region in MOMA. The chromosome representation, the selection, crossover and mutation operations are described as below.

Chromosome representation

In our study, real valued representation is employed in the GA based clustering algorithm for the same reason discussed in section 3.1. Let N be the number of features and C be the number of clusters. Then the length of chromosome would be $N \times C$. The solution of this clustering problem will be a set of cluster centers, with each individual string being a sequence of real numbers representing each cluster center's information. The chromosome is represented as:

$$Chromosome = \left\{ \underbrace{v_{11} \dots v_{1f}}_{\mathbf{v}_1} \underbrace{v_{21} \dots v_{2f}}_{\mathbf{v}_2} \dots \underbrace{v_{c1} \dots v_{cf}}_{\mathbf{v}_c} \right\}_{c \times f}, \quad (4.4)$$

where $\mathbf{v}_j$, $j=1 \dots C$ corresponds to an f -dimensional feature vector of the cluster prototype i .

Selection operator

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

A crowded comparison operator ($<_n$) guides the selection process towards a uniformly spread-out Pareto-optimal front. Every individual i in the population has two attributes:

- Non-domination rank (i_{rank}), which is associated with non-dominated front that this individual belongs to.
- Crowding distance (i_{distance}), which is calculated by the crowding-distance-assignment algorithm.

According to the crowded comparison operator $<_n$, that “solution $\mathbf{i}$ is considered better than solution $\mathbf{j}$ ” is defined by:

$$\mathbf{i} <_n \mathbf{j} \text{ if } (i_{\text{rank}} < j_{\text{rank}}) \text{ or } ((i_{\text{rank}} = j_{\text{rank}}) \text{ and } (i_{\text{distance}} > j_{\text{distance}})), \quad (4.5)$$

which means that for two solutions with differing non-dominated ranks, the solution with lower rank is better; otherwise, if both solutions belong to the same front, the solution which is located in a less crowded region is better.

After all solutions are assigned to a non-domination rank and a crowding distance, the selection operator is used to find good solutions as parents for evolving. A stochastic binary tournament selection operator is used based on the crowded comparison operator. This is to maintain the diversity of the population as well as to choose good solutions for reproduction.

Crossover operator

Suppose that two parent solutions $\mathbf{x}_1$ and $\mathbf{x}_2$ obtained from selection operator are crossed with a probability p_c . For crossover, the solutions are crossed variable-by-variable to create new children solutions $\mathbf{y}_1$ and $\mathbf{y}_2$. Depending on the relative distance between the parent parameter

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

values, children solutions are created by using a polynomial probability distribution. Each variable is crossed with a probability of 0.5 using the following steps:

Step 1: Create a random number u between 0 and 1.

Step 2: Calculate β_q as follows:

$$\beta_q = \begin{cases} (u\alpha)^{\frac{1}{\eta_c+1}}, & \text{if } u \leq \frac{1}{\alpha} \\ \left(\frac{1}{2-u\alpha}\right)^{\frac{1}{\eta_c+1}}, & \text{otherwise} \end{cases}, \quad (4.6)$$

where $\alpha = 2 - \beta^{-(\eta_c+1)}$ and β is calculated as:

$$\beta = 1 + \frac{2}{x_{2i} - x_{1i}} \min \left\{ (x_{1i} - x_i^l), (x_i^u - x_{2i}) \right\}, \quad (4.7)$$

x_i^l and x_i^u are denoted as lower bound and upper bound of the parameter x_i . The parameter η_c is the distribution index and can take any non-negative value. A small value of η_c allows solution far away from parents to be created as children solutions and a large value restricts only near-parent solutions to be created as children solutions.

Step 3: The children solutions are then calculated as follows:

$$y_{1i} = 0.5 \left[(x_{1i} + x_{2i}) - \beta_q |x_{2i} + x_{1i}| \right], \quad (4.8)$$

$$y_{1i} = 0.5 \left[(x_{1i} + x_{2i}) + \beta_q |x_{2i} + x_{1i}| \right]. \quad (4.9)$$

Mutation operator

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

In the mutation operation, a polynomial probability distribution is used to create a solution z_{ji} from a parent solution y_{ji} . The following procedure is used for each variable with a probability p_m .

Step 1: Create a random number u between 0 and 1.

Step 2: Calculate the parameter as follows:

$$\delta_q = \begin{cases} [2u + (1-u)(1-\delta)^{\eta_m+1}]^{\frac{1}{\eta_m+1}} - 1 & , \text{ if } u \leq 0.5 \\ 1 - [2(1-u) + 2(u-0.5)(1-\delta)^{\eta_m+1}]^{\frac{1}{\eta_m+1}} & , \text{ otherwise} \end{cases} \quad (4.10)$$

where $\delta = \frac{\min\{(y_{ji} - y_{li}), (y_{ui} - y_{ji})\}}{y_{ui} - y_{li}}$. The parameter η_m is the distribution index for mutation

and takes any non-negative value. We used $\eta_m = 100 + t$ (where t is the iteration number) here.

Step 3: Calculate the mutated child as follows:

$$z_{ji} = y_{ji} + \delta (y_{ui} - y_{li}). \quad (4.11)$$

The mutation probability p_m is linearly varied from $1/P$ (P is the number of variables in the optimization problem) till 1.0, so that, on the average, one parameter gets mutated at the beginning and all parameters get mutated at the end of a simulation run.

The algorithm

Initially, a random parent population P_o is created. The population is sorted based on the non-domination. Each solution is assigned to a fitness value equal to its non-dominated level. The smaller the value fitness assigned, the better the non-domination. In the meantime, tournament selection, recombination and mutation operators are employed to create the child population Q_o of size N . Because elitism is introduced by comparing current population with

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

the previously-found best non-dominated solution, the procedure is different after the initial generation. The algorithm is outlined as below:

Begin
Initialize:
 $R_t = P_t \cup Q_t$; combine parent and children population
 $F = \text{fast-nondominated-sort}(R_t)$; F is set of non-dominated front of R_t
 $P_{t+1} = 0$ and $i = 1$
Repeat
 Crowding-distance-assignment (F_i) ; calculate crowding distance in F_i
 $P_{t+1} = P_{t+1} \cup F_i$; include i^{th} non-dominated front in parent pop
 $i = i + 1$
Until $|P_{t+1}| < N$
 Sort ($F_i, <_n$) ; sort in descending order using $<_n$
 $P_{t+1} = P_{t+1} \cup F_i[1 : (N - |P_{t+1}|)]$; choose the first $(N - |P_{t+1}|)$ elements of front F_i
 $Q_{t+1} = \text{make-new-pop}(P_{t+1})$; use selection, crossover, mutation to create a new pop
 $t = t + 1$; increment the new generation counter
End

4.3.3 Local Search in MOMA

MOMA employs GA to guide the search direction towards the global region whereas local search approach provides to find the local optimum quickly because of their speedy convergences. When hybrid algorithms are applied to a multi-objective optimization problem, we have to specify an objective function to be optimized by local search. This specification is straightforward in the application for a single-objective optimization problem because the single objective can be used for both genetic search and local search. However, the specification of objective functions for local search is not straightforward in the case of multi-objective optimization. Weighted sum aggregation is a traditional way of solving this issue.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

Weighted sum aggregation method

Weighted sum approach is a method to port local search to multi-objective optimization problems (see Figure 4.3). This method takes each objective function and multiplies it by a fraction of one, namely "weighting coefficient". The modified functions are then added together to obtain a single objective function, which can easily be solved using any single objective optimization method. When dealing with the multi-objective memetic algorithm, weighted sum aggregation method is used as a local search operator. All objectives of an individual solution are aggregated and summed them together. The single objective summation is optimized by local search. New solution found by local search is updated as an improved individual solution.

For an N -objective optimization problem, a weight vector is specified as

$$\mathbf{w} = (w_1, w_2, \dots, w_n) \text{ with } 0 \leq w_i \leq 1 \text{ and } \sum_{i=1}^n w_i = 1 \quad (4.12)$$

The scalar fitness function for applying local search is the weighted sum of objective functions:

$$f(\mathbf{x}) = w_1 f_1(\mathbf{x}) + w_2 f_2(\mathbf{x}) + \dots + w_n f_n(\mathbf{x}) \quad (4.13)$$

And the initial solution to do local search with $f(\mathbf{x})$ is $\mathbf{x}_o$.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

Figure 4.3: Illustration of weighted sum approach

It is noticed that with the weighted aggregation as a local search operator, the current solution does not move to any dominated neighbors because the weights are specified as $w_i \geq 0$ for all $i = 1, 2, \dots, N$ in the scalar fitness function. Therefore, the local search does not degrade the current solution in the sense of the Pareto-dominance relation. The specification of the weight vector $w = (w_1, w_2, \dots, w_n)$ corresponds to the local search direction in the objective space. In the literature, there are two ways to choose the weight vector: static and dynamic weights. In this thesis, we employ static weight method due to its simplicity.

The algorithm

Basically, local search in MA is conducted in an iterative manner. A search direction and an appropriate step length are calculated from the current candidate each iteration. A new search point is then created in that direction. The fitness between the original and newly created candidate is then retained. Local search methods differ from each other with regards to the rules, which govern the process of generating search direction and constructing new search point. A typical local search method can be outlined as below:

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

Begin**Initialize:** Start from some initial candidate solution $\mathbf{x}_0$ Iteration $k = 0$ **While** (stopping conditions are not satisfied)Generate search direction $\mathbf{v}_k$ and step length s_k from current candidate $\mathbf{x}_k$ Generate new candidate $\mathbf{x}'_k = \mathbf{x}_k + s_k \times \mathbf{v}_k$ Compare $\mathbf{x}_k$ and $\mathbf{x}'_k$ retain better one as candidate for next iterationIteration $k = k+1$ **End while****End**

Stopping conditions for a local search can be defined by time bound or maximum number of objective function evaluation. The required accuracy ε is achieved if $|\mathbf{x}_{k+1} - \mathbf{x}_k| < \varepsilon$ at some iteration k . Rate of convergence is defined as the rate at which the required accuracy can be achieved. In our research, we employed Davies-Swann-Campey (DSC) strategy [51] due to better performance compared to other methods (referring to Section 2.2.1).

4.4 Experimental Result and Analysis

4.4.1 The Performance of MOMACL with Validation Indices

To compare with the previous work on chapter 3 of the thesis which applied to the sphere-like data sets, we used two performance measures out of the four validity indices: they are PC and XB (see Chapter 3). The PC index reflects the compactness of the partitioning while the XB index is the quotient of compactness and spatial separation.

$$\begin{aligned} f_1 &= PC(c) \\ f_2 &= XB(c) \end{aligned} \tag{4.14}$$

The application of MOMACL applied to fuzzy clustering is called MOMAFC. The equations used to computing membership values and the distance values are the same as the traditional

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

FCM method. The experimental setup is shown in Table 4.1. The parameter settings are obtained by reusing the values from Table 3.7. The values of w_1 and w_2 imply to consider the tradeoff of each objective function as in Eq. (4.13).

Data set	Pop size	pC	pM	nEva	LS	w_1	w_2
TEST9	30	0.7	0.03	50000	DSCG	0.5	0.5
IRIS	30	0.7	0.03	30000	DSCG	0.5	0.5
GLASS	30	0.7	0.03	100000	DSCG	0.5	0.5
VOWEL	30	0.7	0.03	100000	DSCG	0.5	0.5

Table 4.1: MOMAFC setup

c	FCM (single objective)						MOMA-FC (multi-objective)											
	m=2			m=2.5			m=3			m=2			m=2.5			m=3		
	PC	XB		PC	XB		PC	XB		PC	XB		PC	XB		PC	XB	
2	0.834	0.113		0.731	0.092		0.661	0.071		0.701	0.190		0.598	0.259		0.728	0.238	
3	0.707	0.222		0.572	0.172		0.490	0.126		0.741	0.156		0.606	0.206		0.724	0.160	
4	0.615	0.314		0.468	0.228		0.385	0.162		0.688	0.387		0.541	0.301		0.458	0.235	
5	0.539	0.554		0.416	0.163		0.330	0.102		0.603	0.618		0.480	0.227		0.394	0.166	
6	0.512	0.434		0.366	0.274		0.268	0.212		0.593	0.515		0.447	0.355		0.349	0.293	

Table 4.2: Clustering validation indexes of MOMAFCM algorithm on IRIS data set

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

Analysis

Clustering validation indexes are used and directly optimized that the MOMAFC shows the different results with significant effects on the determination of the optimal number of clusters. For IRIS data set as shown in Table 4.2, the expected optimal number of clusters is correctly defined by the validation indices. There are three clusters existing in the IRIS data set in which 2 of them are overlapped. Therefore, the previous methods of FCM and GA-FCM are unable to define the optimal as $c=3$ but found it as $c=2$ instead. The MOMAFC uses another objective function as a spatial separation measure which helps to distinguish the two overlapped clusters.

Similarly to GLASS data set in Table 4.3, MOMAFC correctly defines the expected optimal number of clusters as $c=6$. For this data set, FCM finds $c=3$ or $c=4$, which is affected by the non-spherical cluster feature of GLASS data. As for GLASS data is simple and has a clear cluster structure, MOMAFC and GA-FCM obtained similar results as shown in Table 4.4. However, in the case of non-structured data as VOWEL data set, all other previous methods are unable to define the expected number of clusters. In this case, MOMAFC has a very significant improvement since it can define the number of clusters to be around the optimal number as $c=11$. The result defines as $c=9$, $c=10$ or $c=11$, which are close or equal to the expected optimal number.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

4.4.2 Performance of Clustering Methods with Real Complex Data Sets

Real world complex data sets

Name	Variant	Samples n	Features f	Clusters c	Chromosome length l
ABALONE		4177	8	3	24
LETTER		20000	16	26	416
YEAST		1484	8	10	80
HEART DISEASE	Cleveland	303	14	5	70
	Hungarian	294	14	5	70
	Switzerland	123	14	5	70
	Long Beach VA	200	14	5	70
FLARE		1066	10	3	30
MARKETING		8993	13	9	117

Table 4.5: Real complex data sets

In this section, experiments were conducted to examine the performances of clustering methods with real complex data sets. The data sets are collected from UCI Machine Learning Repository [41] with a high number of samples, features and clusters. Heart disease data sets have small number of samples but with a large chromosome length. Abalone, Yeast, and Flare data sets have an average high number of samples. Marketing and Letter have a significantly high number of samples and long chromosome presentation. Both data sets are known to be highly overlapped and have multiple clustering structures. In the genetic based algorithms, chromosome length, numbers of samples and data structures are dominant to the performance of clustering methods.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

MOMA Pareto Front Analysis

In this section, we present a brief discussion on the Pareto front and its related performance criteria.

Figure 4.4: [a] Globally and locally Pareto-optimal [b] An ideal set of non-dominated solutions

In Figure 4.4[a], the shorter curve represents the Locally Pareto Optimal front, because none of its points have any neighbor which dominates any member of this set. The longer curve represents the Globally Pareto-optimal front. None of the other points in the search space dominates any member of this set. The two main goals in multi-objective optimization are to find a set of solutions as close as possible to the true Pareto-optimal front, i.e., the convergence and a set of solutions as diverse as possible. In Figure 4.4[b], we show the performance of an ideal MOMA on a hypothetical problem. It is clear that all of the obtained non-dominated solutions lie on the Pareto-optimal front and they also maintain a uniform-like distribution over the entire Pareto-optimal region.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

The scope of MOMA in this thesis focuses on the final solution set which demonstrates directly to the performance of clustering algorithm. Hence, the selection of solutions is based on the objective function's values. This applies to first performance criterion of MOMA which is to find a set of solutions as close as possible to the true Pareto-optimal front. The diversity criterion reflects the performance of MOMA in terms of exploring as many different solutions as possible. However, analysis of MOMA's diversity performance is not the focus of this thesis.

In this section, we examine the obtained best solution set, where the proposed MOMACL is applied to IRIS data set with different initialization settings. The figure below shows the Pareto front plot for IRIS data set.

Figure 4.5: Pareto front of MOMACL algorithm conducted by the simulation of IRIS dataset

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

As shown in Figure 4.5, the solution set forms a part of the Pareto front simulated by IRIS data set. As we select a solution set based on the convergence criterion, this is the best solution set to be obtained. The obtained solutions are expected to be closest to the optimal Pareto front of the data set.

The low diversity of the result is a known issue in this research. Since the focus of this research is to optimize the solution for clustering, the diversity performance of MOMA will not be analyzed in detail. The MOMA implementation employed in this thesis is a standard and simple version. Previous studies which focus on MOMA for data clustering were conducted with synthetic data and not yet applicable for real and complex data. The synthetic data sets used normally have a small number of variables which make the plotting and analysis of the Pareto front in more feasible. With real complex data set and a large number of variables, the performance of this MOMA version is not optimal and needs to be improved in future research.

Experiments and results analysis

In the previous experiments, clustering indices have also been used for MOMACL in order to show the comparison in terms of clustering validity. In this experiment, objective functions of Dev and Conn (as shown in Eqn. 4.1 and 4.2 respectively) are used. These objectives cover two important measures for clustering problem.

$$\begin{aligned} f_1 &= Dev(c) \\ f_2 &= Conn(c) \end{aligned} \tag{4.15}$$

The obtained results are shown under different comparison measures: best recall, best average, success rate and processing time. Best recall and best average are measured in terms

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

of classification rate. The values imply how well the methods classify the data sets after the learning process. Success rate is measured as a rate of obtaining a global optimal solution over n , the number of runs of the method. Processing time is the average of time taken over n numbers of runs.

Best recall

“Recall” rate is the rate applied to train the whole data samples such that, the training set will be the same as the testing set. The results will be compared with known labels of each data sample. The meaning of “recall” reflects how good the algorithm learns from the data set. However, the value does not show the ability of the performance of the system with new data sample.

Figure 4.6: Real data set performance – Best recall

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

As shown in Figure 4.6, all methods obtained a similar recall rate from different data sets. Yeast data is known to be the most unstructured among other data sets. The “recall” rates for Yeast data are from 70% to 80%, which reflect the learning performance of the methods can learn from an unstructured data, whereas other data sets are from 80% to 95%. For MOMACL, there is a noticeable improvement of 5% on average. By optimizing multiple objectives, the method learns the structure of the data set from more than one aspect. Hence, MOMACL obtains a solution which is able to describe the data structure better than the solution obtained by the single-objective methods.

Best average

A K-fold validation is employed for this experiment. As setting $K=4$, the data set is divided the number of data samples randomly with three fourth used for training and one fourth for testing. For the traditional FCM method, the classification rate of each K-fold is obtained from single solution. For genetic based methods, the classification rate for each K-fold is the average value over the population result set. The average values over K folds will be obtained and analyzed.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

Figure 4.7: Real data set performance – Best average

As shown in Figure 4.7, we observe similar patterns as both methods have better classification rate for all data sets except the Yeast data. However, since there are separate sets of training data and testing data, the values obtained are slightly smaller than the recall values. The values range from 60% to 70% for Yeast data and 70% to 90% for other data sets. In this test, since the testing data samples are not included in the learning process, it is difficult for the methods to capture the structure of the data. As can be seen from the figure, the improvement of MOMACL is around 10% which is more than the previous test of 5%. So, in most practical situations when testing data samples are not learned, MOMACL shows significantly better performance than single-objective clustering methods.

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

Success rate

In this test, we examine the performance of clustering methods in terms of the capability to reach the global optimal solution. Each method will be executed by 50 times and the rate of converging to a global optimal solution will be observed. The optimal solution of each method is selected based on the lowest objective function that obtains after 50 runs. For the multi-objective method, the objective will be the accumulative value obtained using the weighted sum aggregation method.

Figure 4.8: Real data set performance – Success rate

From Figure 4.8, we can observe the performance improvement between the traditional FCM and genetic based methods. MA-FCM and GA-FCM have a similar success rate and

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

are slightly better than FCM. In this case, we also observe the improvement of MOMACL over the GA-FCM and MA-FCM. The improvement is achieved to have an average of 7%. The result shows that there are low values for “letter recognition” and “heart disease” data sets where the length of a chromosome is large.

Processing time

In this test, we examine the performance of the clustering methods in terms of processing time. As for the data with a higher number of samples, the method requires more time for processing. For practical data sets, they used to have a very large number of samples which require a much longer time. Hence, processing time is one of the most important measures for clustering methods in applications for real data sets.

Figure 4.9: Real data set performance – Processing time

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

From Figure 4.9, the time taken for FCM is significantly lower compared to other GA based methods. As the traditional FCM uses gradient search for optimization, the processing time is linearly dependent on the number of samples and features of the data. As discussed in Section 2.3.2, MA-FCM is studied to have one important advantage over GA-FCM which is faster in processing time. As shown in Figure 4.9, MA-FCM shows significantly slower processing times compared to GA-FCM. Since both MA-FCM and MOMACL employ the hybrid of genetic operations and local search, the performances are observed to be similar. For ‘abalone’ and ‘heart disease’ data sets, the size of samples and chromosome is relative small. Hence, the method takes less time to complete the process compared with other data sets. There is a slight improvement of MOMACL over MA-FCM for special data set such as ‘letter recognition’ and complex data set such as ‘yeast’.

4.5 Concluding Remarks

In this chapter, MOMA based clustering is introduced and the experiments conducted show interesting results. The first focus of MOMACL is to improve the “unsupervised” feature of the clustering method which is the ability to define the optimal number of clusters. Since MOMACL uses a direct approach to optimize the validation indices, the result shows significant improvement on both synthetic and real world complex data sets compared to the traditional FCM. In the second experiment, we used two objective functions for MOMACL. The obtained results are presented in terms of the best recall, best average, success rate and processing time to show that there are 5% to 10% improvement by the MOMACL algorithm. The success rate of MOMACL was also observed to be around 7% better than the previous

CHAPTER 4. MULTIPLE-OBJECTIVE CLUSTERING

GA-FCM and MA-FCM. The achieved processing time showed around 25% improvement compared to the previous best MA-FCM.

Generally, the synthetic data sets for a multi-objective memetic problem have a lower number of variables. In this study, the optimal Pareto is obtained with a single variable. Hence, it is convenient to plot the optimal Pareto front for the MOMA analysis. In this research, we used real data sets with a large number of samples, features and cluster sizes. However, studying the optimal Pareto front with other criteria of MOMA is not the main focus of this research hence it is not covered with greater details in this thesis.

As mentioned in Section 4.2, there are related works on multi-objective memetic algorithm and clustering. However, the approach and criteria used are different from our proposed method. For example, MOCK [44] follows the approach of locus-based adjacency encoding scheme which has major advantages on synthetic and simple data sets. However, the method faces an issue with large scale data sets when each chromosome representation is too large for processing. The trial test of MOCK with large data set is failed to perform due to large memory demanding such that it is unable to compare with our method. In this research, we analyzed the proposed method and compare it with traditional and other relevant methods with benchmarking data sets and standard performance criteria. The comparison has shown the improvements by the proposed method.

Chapter 5

Conclusion and Future Works

5.1 Conclusion

In this thesis, we have attempted to cover a wide research of clustering methods. Extensive experiments were conducted to analyze the performances of different clustering algorithms, such as, FCM, GA based method, MA based method and MOMA based method.

Traditional clustering methods are supervised approaches that are dependent on the prior knowledge on the number of appropriate clusters that suits a data set. To achieve unsupervised clustering, evolutionary algorithm represents an appropriate alternative for solving clustering problems. Genetic Algorithm and Memetic Algorithm based clustering methods have been proposed in this thesis for defining the optimal number of cluster for general complex data sets. Evolutionary clustering algorithm also possess ability to locate the global optimal. Since traditional clustering methods make use of gradient search, they venerable to getting stuck at local optimal solution, especially when dealing with complex datasets. Experimental results showed that the performance of evolutionary clustering algorithm is generally better than traditional clustering method that employs gradient search. Memetic Algorithm first materialized as a form of evolutionary and local search hybrid.

CHAPTER 5. CONCLUSION AND FUTURE WORKS

Comparing to traditional evolutionary algorithms such as Genetic Algorithm, Memetic Algorithm has been shown to improve the search processing time significantly.

In the clustering literature, different criteria have been proposed and used to describe the clustering structure of datasets. Since each criterion generally leads to very different clustering result, existing clustering methods usually make assumptions on the availability of prior knowledge about the data set. However, such assumption does not apply well on complex data sets containing multiple unique sub-structures. A multi-objective clustering methodology is hence introduced to address this issue by optimizing multiple criteria simultaneously in the present thesis. The conducted experiments showed that the proposed multi-objective clustering algorithm generated significant improvements in converging to optimal cluster size as well as the quality of clustering on several well known merit commonly used in the literature.

5.2 Future Works

5.2.1 The Compliment with MOCK

The MOCK method makes use of locus-based adjacency representation proposed in [44] to encode the chromosome. For the graph-based representation, each individual has a length of N genes $g = g_1, g_2, \dots, g_N$ where N is the size of the data set. Each gene g_i can take any values j in the range $\{1, \dots, N\}$. A value of j assigned to the i^{th} gene is interpreted as a link between data sample i and j ; then in clustering solution they will be in the same cluster.

CHAPTER 5. CONCLUSION AND FUTURE WORKS

The encoding scheme has important advantages in the sense that MOCK makes use of “connectedness” measure as one of the optimizing objective. The encoding scheme also requires totally different set of genetic operators for crossover and mutation.

However, apart from the strength in MOCK, the encoding scheme leads to a significant disadvantage that when the size of data set is large, the length of the chromosome becomes large too. The data sets used in [44] are synthetic data sets with quite small sizes. However, the size of data set is always large for real data sets, especially for biological data. The method has a limitation in scalability as also mentioned in [45].

On the other hand, in this context, we may use a different scheme of encoding based on the number of clusters and number of features which are limited compared to the number of samples in the data set. The chromosome length is considered smaller and more suitable for evolutionary algorithm and local search.

5.2.2 Model Selection in MOMA

The use of FCM into MOMA requires a customized methodology to combine two algorithms (global and local search algorithms) together and to control the proportion of each method based on its current performance in the generation. A research work in [44] introduces various ways for model selection in the multi-objective clustering problem.

5.2.3 Improving the Scalability of MOMA and Its Application to Large Scale Real Data Sets

We can make use of the locus-based adjacency representation as proposed in [44] which has many proven advantages for MOMA clustering. However, instead of directly mapping all the

CHAPTER5. CONCLUSION AND FUTURE WORKS

data samples into each individual that may lead to a large chromosome size, we can introduce a pre-processing method which defines a sub-cluster in the data set. Then, instead of mapping each data sample, we map the sub-cluster to one gene in the chromosome. The samples belonging to each sub-cluster is assumed to be in the same cluster for the clustering result.

This pre-processing can be done with a simple neighboring mapping method or by a simple fuzzy clustering. The short chromosome improves the scalability of MOMA in order to deal with the real data set containing large number of data samples.

PUBLICATIONS

Publications

1. Anh-Duc Do, Siu-Yeung Cho, and Yew-Soon Ong, "A Multi-Objective Memetic Algorithm Based Clustering Method", World Congress in Computer Science Computer Engineering and Applied Computing, International Conference on Artificial Intelligence, July 2008, Las Vegas, USA.
2. Anh-Duc Do and Siu-Yeung Cho, "Memetic Algorithm Based Fuzzy Clustering", IEEE Congress on Evolutionary Computation CEC2007, Sept. 2007, Singapore.
3. Siu-Yeung Cho, Anh Duc Do, and Chai Quek, "Possibilistic Clustering for Neural Network based Breast Cancer Detection in Thermograph", in 2006 ACM Symposium on Applied Computing, April 2006, Dijon France.

REFERENCES

References

- [1] G. J. Klir and B. Yuan, "Fuzzy sets and fuzzy logic, theory and applications", Prentice Hall, 1995
- [2] M. Razaz, "A fuzzy C-means clustering placement algorithm", Circuits and Systems, IEEE International Symposium, Vol.3, pp. 2051-2054, 1993
- [3] N. Watanabe and T. Imaizumi, "Fuzzy k-means clustering with crisp regions", Fuzzy Systems, IEEE International Conference, Vol.1, pp. 199-202, 2001
- [4] A. Baraldi and P. Blonda, "A survey of fuzzy clustering algorithm for pattern recognition - Part I", IEEE Transactions on Systems, Vol.29, 1999
- [5] A. Baraldi and P. Blonda, "A survey of fuzzy clustering algorithm for pattern recognition - Part II", IEEE Transactions on Systems, Vol.29, 1999
- [6] P. Teppola, S. P. Mujunen, and P. Minkinen, "Adaptive fuzzy c-means clustering in process monitoring", Chermometrics and Intelligent Laboratory System, Vol.45, pp. 22-38, 1999
- [7] X. Chang, W. Li, and J. Farrell, "A C-means clustering based fuzzy modeling method", Fuzzy Systems, IEEE International Conference, Vol.2, pp. 937-940, 2000
- [8] J. C. Bezde, "Pattern recognition with fuzzy objective function algorithm", Plenum Press, New York, 1981
- [9] W. E. Phillips, R. P. Velthuisen, S. Phuphanich, L. O. Hall, L. P. Clark, and M. L. Silbiger, "Application of fuzzy c-means segmentation technique for tissue differentiation in MR images of a hemorrhagic glioblastoma multifrome", Magnetic Resonance Imaging, Vol.13(2), pp. 277-290, 1995
- [10] M. R. Rezaee, P. M. J. Zwet, B. P. E. Lelieveldt, R. J. Geest, and J. H. C. Reiber, "A multiresolution image segmentation technique based on pyramidal segmentation and fuzzy clustering", IEEE Transactions On Image Processing, Vol.97, pp. 1238-1248, 2000
- [11] J. Yu, Q. Cheng, H. Huang, "Analysis of the weighting exponent in the FCM", IEEE Transactions on Systems, Vol.34, pp. 634-638, 2004
- [12] N. S. Iyer, A. Kandel, and M. Schneider, "Feature-based fuzzy classification for interpretation of mammograms", Fuzzy Sets and Systems, Vol.114, pp. 271-280, 2000

REFERENCES

- [13] K. Hirota and W. Pedrycz, "Fuzzy computing for data mining", Proceedings of the IEEE, Vol.87(9), pp. 1575-1600, 1999
- [14] T. Hastie, R. Tibshirani, and J. Friedman, "The elements of statistical learning: data mining, inference and prediction", Springer-Verlag, 2001
- [15] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review", ACM Computing Surveys, Vol.31, pp. 264-323, 1999
- [16] R. Dubes and A. K. Jain, "Algorithms for clustering data", Prentice-Hall, 1988
- [17] E. D. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Addison-Wesley, 1989
- [18] J. H. Holland, "Adaptation in Natural and Artificial Systems", The University of Michigan Press, 1975
- [19] P. Moscato, "On evolution, search, optimization, genetic algorithms and martial arts: Toward memetic algorithms", Caltech concurrent Computation program, Tech. Rep. 790, California Institute of Technology, Pasadena, 1989
- [20] Y. S. Ong and A.J. Keane, "Meta-Lamarckian Learning in Memetic Algorithm", IEEE Transactions On Evolutionary Computation, Vol. 8, No. 2, pp. 99-110, April 2004
- [21] Y. S. Ong, M. H. Lim, N. Zhu and K. W. Wong, "Classification of Adaptive Memetic Algorithms: A Comparative Study", IEEE Transactions On Systems, Man and Cybernetics - Part B, Vol. 36, No. 1, pp. 141-152, February 2006
- [22] Z. Zhu, Y. S. Ong and M. Dash, "Markov Blanket-Embedded Genetic Algorithm for Gene Selection", Pattern Recognition, Vol. 40, no. 11, pp. 3236-3248, Nov 2007
- [23] Z. Zhu, Y. S. Ong and J. Zurada, "Identification of Full and Partial Class Relevant Genes", IEEE/ACM Transactions on Computational Biology and Bioinformatics, Accepted & In Press, 2008
- [24] Z. Zhu, Y. S. Ong and M. Dash, "Wrapper-Filter Feature Selection Algorithm Using A Memetic Framework", IEEE Transactions On Systems, Man and Cybernetics - Part B, Vol. 37, no. 1, pp. 70-76, Feb 2007

REFERENCES

- [25] X. F. Fan, Z. Zhu, Y. S. Ong, Y. M. Lu, Z. X. Shen, and Jer-Lai Kuo, "A Direct First Principle Study on the Structure and Electronic Properties of $\text{BexZn}_{1-x}\text{O}$ ", *Applied Physics Letter* 91, 121121, September 2007
- [26] Z. Z. Zhou, Y. S. Ong, P. B. Nair, A. J. Keane and K. Y. Lum, "Combining Global and Local Surrogate Models to Accelerate Evolutionary Optimization", *IEEE Transactions On Systems, Man and Cybernetics - Part C*, Vol. 37, No. 1, Jan. 2007, pp. 66-76
- [27] Y. S. Ong, P.B. Nair and A.J. Keane, "Evolutionary Optimization of Computationally Expensive Problems via Surrogate Modeling", *American Institute of Aeronautics and Astronautics Journal*, 2003, Vol. 41, No. 4, pp. 687-696
- [28] R. N. Dave, "New measure for evaluating fuzzy partitions induced through C-shells clustering", *Intelligent Robots and Computer Vision X: Algorithms and Techniques*, Proceedings of SPIE Conference, pp. 406-414, 1991
- [29] R. Krishnapuran, O. Nasraoui, and H. Frigui, "The fuzzy C spherical shells algorithm: A new approach", *IEEE Transactions on Neural Networks*, Vol.3, pp. 663-671, 1992
- [30] J. Handl and J. Knowles, "Exploiting the trade-off - The benefits of multiple objectives in data clustering", *Evolutionary Multi-Criterion Optimization*, pp. 547-560, Springer-Verlag, 2005
- [31] M. Law, A. Topchy, and A. K. Jain, "Multiobjective data clustering", *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol.2, pp. 424-430, 2004
- [32] A. Strehl and J. Ghosh, "Cluster ensembles - a knowledge reuse framework for combining multiple partitions", *Journal on Machine Learning Research*, Vol.3, pp. 583-617, 2002
- [33] K. C. Tan, T. H. Lee, D. Khoo, and E. F. Khor, "A multi-objective evolutionary algorithm toolbox for computer-aided multi-objective optimization", *IEEE Transactions on Systems*, Vol.31, 2001
- [34] R. Dawkins, "The Selfish Gene", Claredon Press, Oxford, 1976
- [35] M. Ankerst, M. Breunig, H. P. Kriegel, and J. Sander, "Optics: Ordering points to identify clustering structure", *Proceedings of International Conference on Management of Data*, pp. 49-60, ACM Press, 1999

REFERENCES

- [36] M. Ester, H. P. Kriegel, and J. Sander, "A density-based algorithm for discovering clusters in large spatial databases with noise", Proceedings of the Second International Conference on Knowledge Discovery and Data-Mining, AIII Press, 1996
- [37] E. Vorhees, "The effectiveness and efficiency of agglomerative hierarchical clustering in document retrieval", PhD thesis, Department of Computer Science, Cornell University, 1985
- [38] S. Bandyopadhyay and U. Manlik, "Nonparametric genetic clustering: comparison of validity indices", IEEE Transactions on Systems, Vol.31, pp.120-125, 2001
- [39] V. J. Rayward-Smith, I. H. Osman, C. R. Reeves, and G. D. Smith, "Modern heuristic search methods", John Wiley and Son Ltd, 1996
- [40] X. Z. Fern and C. E. Brodley, "Solving cluster ensemble problems by bipartite graph partitioning", International Conference on Machine Learning, pp. 281–288, ACM Press, 2004
- [41] UCI Machine Learning Repository: <http://mllearn.ics.uci.edu/MLRepository.html>
- [42] K. Deb, A. Pratab, S. Agarwal, and T. Meyarivan, "A fast and elitist multi- objective genetic algorithm: NSGA-II", IEEE Transactions on Evolutionary Computation, Vol.6, pp. 182–197, 2002
- [43] J. D. Knowles and D. Corne, "Approximating the non-dominated front using the Pareto archived evolution strategy", Evolutionary Computation, Vol.8(2), pp. 149–172, 2000
- [44] J. Handl and J. D. Knowles, "An Evolutionary Approach to Multiobjective Clustering", IEEE Transactions on Evolutionary Computation, Vol. 11(1), 2007
- [45] J. Handl and J. D. Knowles, "Improvements to the scalability of multiobjective clustering", Proceedings of IEEE Congress on Evolutionary Computation, 2005
- [46] L. MacQueen, "Some methods for classification and analysis of multivariate observations", Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, pp.281-297, University of California Press, 1967
- [47] E. Vorhees, "The effectiveness and efficiency of agglomerative hierarchical clustering in document retrieval", PhD thesis, Department of Computer Science, Cornell University, 1985
- [48] G. McLachlan and T. Krishnan, "The EM algorithm and extensions", John Wiley and Son Ltd,

REFERENCES

1997

- [49] R. O. Duda, P. E. Hart, and D. G. Stork, "Pattern classification", Second Edition, John Wiley and Son Ltd, 2001
- [50] B. S. Everitt, "Cluster analysis", Edward Arnold, 1993
- [51] H. Ishibuchi, T. Yoshida, and T. Murata, "Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling", IEEE Transactions on Evolutionary Computation, Vol.7(2), pp. 204-223, 2003