A NEW GENETIC ALGORITHM BASED
CLUSTERING FOR BINARY AND IMBALANCED
CLASS DATA SETS

Sabariah Saharan

A thesis submitted for the degree of Doctor of
Philosophy
Abstract

This research was initially driven by the lack of clustering algorithms that specifically focus on binary data. To overcome this gap in knowledge, a promising technique for analysing this type of data became the main subject in this research, namely Genetic Algorithm (GA). This type of algorithm has an intrinsic search parallelism that avoids getting stuck at the local optima and poor initialization. For the purpose of this research, GA was combined with the Incremental K-means (IKM) algorithm to cluster the binary data streams. However, prior to this proposed method, a well-known GA based clustering method, GCUK was applied to gauge the performance of this algorithm to cluster the binary data, with new application for binary data set. Subsequently, this led to a proposed new method known as Genetic Algorithm-Incremental K-means (GAIKM) with the objective function based on a few sufficient statistics that may be easily and quickly calculated on binary numbers. Different from the other clustering algorithms for binary data, this proposed method has an advantage in terms of fast convergence by implementing the IKM. Additionally, the utilization of GA provides a continuous process of searching for the best solutions, that can escape from being trapped at the local optima like the other clustering methods. The results show that GAIKM is an efficient and effective new clustering algorithm compared to the clustering algorithms and to the IKM itself. The other main contribution in this research is the ability of the proposed GAIKM to cluster imbalanced data sets, where standard clustering algorithms cannot simply be applied to this data as they could cause misclassification results. In conclusion, the GAIKM outperformed other clustering algorithms, and paves the way for future research in missing data and outliers and also by implementing the GA multi-objective optimization.
Acknowledgement

First and foremost, I am grateful to Almighty God for giving me many great opportunities in life, including this challenging yet wonderful PhD journey.

I am deeply thankful to Professor Jennifer Brown, Professor Roberto Baragona and A/Prof Marco Reale for their helpful supervision, generous assistance and encouragement to pursue this research that led to this thesis.

I owe very much to Professor Roberto Baragona for his thoughtful advice, knowledge and wisdom. His guidance and assistance were invaluable in helping me acquire the necessary skills to become a PhD researcher.

I would like to express my great appreciation to my parents, family and friends. It was hard being away from them, but their constant love and care touched me in many ways. Without their support, this work would not have been possible. To my closest friends in Christchurch, thank you for being an instant family to us these past years.

I am very fortunate to have my loving, understanding husband, Mohd Haneff Soid, and all my daughters, Irdina Haniss, Arissa Humaira, and Sarah Syuhada and my only son,
Imran Wafi, by my side throughout my study. I am forever in their debt for supporting me throughout countless weekends that were supposed to be reserved especially for them. Not forgetting my mum, Hajah Habibah Ma’som for her endless and countless pray, my siblings, and all my in laws. I would not have been able to complete this journey without their love, patience and pray.

I would like to acknowledge my employer, Universiti Tun Hussein Onn Malaysia (UTHM) and the Government of Malaysia for funding my PhD program and living expenses for my family and I in Christchurch. Without these financial support I would not have been able to undertake this program.

Finally I am also grateful to many others, all my friend in Christchurch and Malaysia who are really kind and helpful throughout this journey and all of whom can not be named here, but you are always in my heart!
Contents

1 Introduction 11
   1.1 Introduction ................................................................. 11
   1.2 Objectives of the Study ..................................................... 15
   1.3 Significance of the Study .................................................... 16
   1.4 Scope and Limitation of the Study ........................................ 18
   1.5 Overview of Thesis .......................................................... 19

2 Overview of Genetic Algorithms 21
   2.1 Introduction of Genetic Algorithms ...................................... 21
   2.2 Encoding ............................................................................. 24
      2.2.1 Point-based Encoding .................................................... 25
      2.2.2 Centre-based Encoding .................................................. 25
      2.2.3 Locus-based Adjacency Representation .............................. 26
   2.3 Fitness Function .................................................................... 27
   2.4 Elitism ................................................................................. 28
   2.5 Selection Operator ............................................................... 29
      2.5.1 Fitness Proportional / Roulette Wheel Selection .................. 30
      2.5.2 Stochastic Remainder Roulette Wheel Selection, (SRWS) ...... 31
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.5</td>
<td>Crossover Operator</td>
<td>62</td>
</tr>
<tr>
<td>4.2.6</td>
<td>Mutation Operator</td>
<td>62</td>
</tr>
<tr>
<td>4.2.7</td>
<td>Termination Process</td>
<td>63</td>
</tr>
<tr>
<td>4.3</td>
<td>Application to Real World Data Sets</td>
<td>63</td>
</tr>
<tr>
<td>4.4</td>
<td>Results</td>
<td>66</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Number of clusters, $K$</td>
<td>66</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Parameter Settings</td>
<td>72</td>
</tr>
<tr>
<td>4.4.3</td>
<td>GCUK-binary Results</td>
<td>75</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Summary</td>
<td>76</td>
</tr>
<tr>
<td>5</td>
<td>Genetic Algorithm with Incremental K-Means</td>
<td>79</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>79</td>
</tr>
<tr>
<td>5.1.1</td>
<td>Incremental K-Means, IKM</td>
<td>80</td>
</tr>
<tr>
<td>5.1.2</td>
<td>Data Representation</td>
<td>81</td>
</tr>
<tr>
<td>5.1.3</td>
<td>Fitness Function</td>
<td>82</td>
</tr>
<tr>
<td>5.1.4</td>
<td>Selection Process for GAIKM</td>
<td>82</td>
</tr>
<tr>
<td>5.1.5</td>
<td>Crossover Process for GAIKM</td>
<td>83</td>
</tr>
<tr>
<td>5.1.6</td>
<td>Mutation Process for GAIKM</td>
<td>84</td>
</tr>
<tr>
<td>5.1.7</td>
<td>Termination Criteria</td>
<td>85</td>
</tr>
<tr>
<td>5.1.8</td>
<td>The Process of GAIKM</td>
<td>85</td>
</tr>
<tr>
<td>5.2</td>
<td>A Simulation Study</td>
<td>86</td>
</tr>
<tr>
<td>5.3</td>
<td>Application to Real World Data Sets</td>
<td>93</td>
</tr>
<tr>
<td>5.3.1</td>
<td>Results of Parameter Settings</td>
<td>94</td>
</tr>
<tr>
<td>5.3.2</td>
<td>Results of GAIKM</td>
<td>97</td>
</tr>
<tr>
<td>5.4</td>
<td>Summary</td>
<td>98</td>
</tr>
</tbody>
</table>
6 Conclusion

6.1 GCUK-binary ....................................................... 100

6.2 GAIKM ............................................................... 102

6.3 Recommended Future Studies .................................... 103

APPENDIX

.1 Appendix A ............................................................ 114
List of Figures

2.1 Basic steps in GA ......................................................... 23
2.2 Locus-based Adjacency Representation illustration .......................... 26
2.3 Roulette Wheel Selection Illustration ........................................ 30
2.4 Stochastic Roulette Wheel Selection Illustration .............................. 32
2.5 Stochastic Universal Sampling Illustration ..................................... 33
2.6 Binary Tournament Selection Illustration ...................................... 34
2.7 Single Crossover Illustration .................................................... 35

3.1 Confusion Matrix ............................................................... 46
3.2 Illustration of Pareto-front ..................................................... 49
3.3 Illustration of Minimum Spanning Tree ......................................... 52

4.1 Silhouette index for Nursery data, k=4 ....................................... 69
4.2 Silhouette index for Nursery data, k=5 ....................................... 69
4.3 Silhouette index for Nursery data, k=6 ....................................... 69
4.4 Silhouette index for Car Evaluation data, k=3 ................................ 69
4.5 Silhouette index for Car Evaluation data, k=4 ................................ 69
4.6 Silhouette index for Car Evaluation data, k=5 ................................ 69
4.7 Silhouette index for Bank Marketing data, k=3 ................................ 70
4.8 Silhouette index for Bank Marketing data, k=4
4.9 Silhouette index for Bank Marketing data, k=5
4.10 Silhouette index for CRTA data, k=5
4.11 Silhouette index for CRTA data, k=6
4.12 Silhouette index for CRTA data, k=7
4.13 Silhouette index for CRTA2 data, k=4
4.14 Silhouette index for CRTA2 data, k=5
4.15 Silhouette index for CRTA2 data, k=6
4.16 Silhouette index for URTA data, k=5
4.17 Silhouette index for URTA data, k=6
4.18 Silhouette index for URTA data, k=7

1 Number of generations of the best individual for Nursery data set
2 Number of generations of the best individual for Car Evaluation data set
3 Number of generations of the best individual for Bank Marketing data set
4 Number of generations of the best individual for CRTA data set
5 Number of generations of the best individual for CRTA2 data set
6 Number of generations of the best individual for URTA data set
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Terms in Genetic Algorithms</td>
<td>22</td>
</tr>
<tr>
<td>3.1</td>
<td>Operational Taxonomic Units</td>
<td>45</td>
</tr>
<tr>
<td>4.1</td>
<td>Clusters of variables obtained from the GCUK algorithm</td>
<td>67</td>
</tr>
<tr>
<td>4.2</td>
<td>Silhouette index for GCUK-binary for all the data sets</td>
<td>68</td>
</tr>
<tr>
<td>4.3</td>
<td>Different parameter settings and the $DB$ value for Nursery data</td>
<td>73</td>
</tr>
<tr>
<td>4.4</td>
<td>Different parameter settings and the $DB$ value for Car Evaluation data</td>
<td>73</td>
</tr>
<tr>
<td>4.5</td>
<td>Different parameter settings and the $DB$ value for Bank Marketing data</td>
<td>73</td>
</tr>
<tr>
<td>4.6</td>
<td>Different parameter settings and the $DB$ value for CRTA</td>
<td>74</td>
</tr>
<tr>
<td>4.7</td>
<td>Different parameter settings and the $DB$ value for CRTA2</td>
<td>74</td>
</tr>
<tr>
<td>4.8</td>
<td>Different parameter settings and the $DB$ value for URTA</td>
<td>75</td>
</tr>
<tr>
<td>4.9</td>
<td>Result for GCUK-binary</td>
<td>76</td>
</tr>
<tr>
<td>5.1</td>
<td>Cluster prototypes for the &quot;hand-made&quot; clustering structure of the first data set</td>
<td>87</td>
</tr>
<tr>
<td>5.2</td>
<td>Results from Experiment 1</td>
<td>89</td>
</tr>
<tr>
<td>5.3</td>
<td>Probabilities that variables assume value 1 according to a Bernoulli scheme</td>
<td>90</td>
</tr>
<tr>
<td>5.4</td>
<td>Results from Experiment 2</td>
<td>91</td>
</tr>
<tr>
<td>5.5</td>
<td>Correlated variables in each cluster of each data set for Experiment 3</td>
<td>92</td>
</tr>
</tbody>
</table>
5.6 Results from Experiment 3 ......................................................... 93
5.7 Different parameter settings and the $qRW$ value for Nursery data .. 94
5.8 Different parameter settings and the $qRW$ value for Car Evaluation data . 95
5.9 Different parameter settings and the $qRW$ value for Bank Marketing data . 95
5.10 Different parameter settings and the $qRW$ value for CRTA .............. 96
5.11 Different parameter settings and the $qRW$ value for CRTA2 ............ 96
5.12 Different parameter settings and the $qRW$ value URTA ............... 97
5.13 Results for GAIKM ................................................................. 98
Chapter 1

Introduction

1.1 Introduction

The increasing volume of data collected by individuals, firms and organizations at a specific range of time has triggered the urge to explore and process this type of information. Many statistical techniques have been proposed by researchers to analyze this large volume of data such as analysis of variance, linear regression, discriminant analysis, multi-dimensional scaling, classification and clustering.

One of the most popular statistical techniques used to explore and process this increasingly large volume of information is clustering. Clustering can be referred to as an unsupervised classification whereby the data points are clustered using the only information that is available in the data set and the number of clusters, $K$, may be known a priori or not. This technique is deemed one of the most challenging problems in machine learning [Hruschka et al., 2009]. There are more than a thousand different scientific papers and theses published related to this topic, and until today, there is a continuous proposal of new clustering algorithms.
The aim of clustering is to divide the data points into \( K \) clusters, whereby the characteristics of the data within the cluster should be as similar as possible, and they should be different from the other data in other clusters. The higher the similarity within the cluster and the higher the difference between clusters, the better the clustering. In this study, the interest was to cluster the variables to see a similar pattern of these variables. For example given a set of \( n \) variables, \( X = x_1, x_2, \ldots, x_n \) to be clustered with each of these \( X_i \in \mathbb{R}^P \) is an attribute vector used to describe the variables. These variables will be clustered into a set of clusters, \( C = C_1, C_2, \ldots, C_K \) where \( K \) is the number of clusters. The clusters are disjoint \( C, C_i \cap C_j = \emptyset \) for \( i \neq j \). The numbers of \( K \) may be known a priori or not. Let \( \bar{x}_k \) be the mean of the cluster \( C_k \). The main goal of clustering is to find the minimum distance between the \( x_i \) to the closest center \( \bar{x}_k \) as follows:

\[
\sum_{k=1}^{K} \sum_{x_i \in C_k} \| x_i - \bar{x}_k \|^2
\]

A well-known clustering method is K-means [Macqueen, 1967] algorithm which uses an iterative process. This algorithm is easy to implement and efficient [Jain, 2010, Jain et al., 1999]. The method of K-means can be described by the following steps:

1. Choose the initial centers \( (\bar{x}_1, \ldots, \bar{x}_K) \) for each of the clusters \( C_1, \ldots, C_K \).

2. Find the new cluster membership by assigning each data point to the closest centres.

3. Recalculate the clusters centres.

4. Stop the process if all the cluster centres do not change. Otherwise repeat the process from Step 2.
Even though the K-means algorithm is computationally efficient, there are a few critical weaknesses. The quality of the clustering is totally dependent on the initial cluster centres and is sensitive to outliers. Moreover, it can also be easily trapped in a local minimum [Lu et al., 2004b, Ordonez, 2003, Pham et al., 2004]. The other weakness of K-means is that it is time consuming when applied to a large data set [Sanjay Chakraborty, 2011].

Incremental K-means (IKM) [Ordonez, 2003, Pham et al., 2004, Sanjay et al., 2011, Sanjay Chakraborty, 2011] was proposed to improve the performance of the standard K-means with different objectives. Chakraborty and Nagwani [2011], Sanjay et al. [2011], Sanjay Chakraborty [2011] proposed an IKM where the new data are clustered by comparing their smallest distance from the means of the existing clusters. The result is same as if a standard K-means was run for the whole data set, but this new proposed method needed less computation time. The proposed IKM showed better performance when compared with the standard K-means. However, when the threshold value or the % of δ change in the database exceeded 57%, standard K-means outperformed IKM.

Pham et al. [2004] proposed an IKM variant that applied the device of allowing a cluster center to jump to a completely different point. This device may enable the cluster centres to move to reduce a cluster distortion. When the value of K becomes larger, the search process is slowed down because the algorithm needs to find all possible beneficial places to insert the new clusters.

The IKM method proposed by Ordonez [2003] deals with large binary data sets specifically. This method may be viewed as a compromise method between the online K-means and
the standard K-means algorithms. Online K-means, most commonly known as sequential K-means, allows to update the model as new data is received. However, unlike these two methods, this IKM will not iterate until convergence, and only the values of the sufficient statistics $N$ and $M$ have to be updated at each step. Let $n$ be the number of cases, then $N_j, j = 1, \ldots, k$ is the number of cases that are assigned to cluster $j$ and $M_{ij}$ is the sum of the instances that equal to 1 recorded for the factor $i$ for all the cases that are assigned to cluster $j$. The performance of IKM outperformed standard K-means, Scalable K-means and On-line K-means in most cases. However, when this algorithm was applied to very sparse matrices and very high dimensional data set, it was found that the Scalable K-means had slightly better solutions than IKM.

Another approach to improve the performance of the standard K-means is by using Genetic Algorithms. Genetic Algorithms or GA was an idea from John Holland and his friend in the early of 1970’s. It is a subclass of the Evolutionary Algorithms which use stochastic searching techniques [Goldberg, 1989]. These techniques are inspired by the process of biological evolution; selection, crossover and mutation processes. The continues process for searching the best population of the problem solutions, gives GA the ability to escape from stopping at the local optima. This can make GA more efficient compared to other local optimization methods.

Some of the researchers that use GA to improve the performances of the standard K-means, are Bandyopadhyay and Maulik [2002b] (Genetic Clustering for Unknown K), Krishna et al. [1999] (Genetic K-means Algorithm ), Lu et al. [2004a] (Fast Genetic K-Means Algorithm), Lu et al. [2004b] Incremental Genetic K-means Algorithm , Arunprasath et al. [2010] (Genetic Algorithm for the K-means Initialization[Kwedlo and Iwanowicz, 2010], Rapid Genetic
Algorithm), and Guo et al. [2006], Li et al. [2010], Murthy and Chowdhury [1996], Tan and Lu [2009](An Improved K-means with Combined with the Genetic Algorithm). The details for the Genetic Clustering Algorithm will be explained in Chapter 3.

Based on these few examples, the idea of using GA for K-means improvement is not new. However, many of these studies focus on quantitative data (discrete and continuous). Therefore the development of clustering for categorical data has not grown as tremendously as that for numerical data. It may be due to the definition of clustering for categorical data being not as clear as the problem for numerical data [Duda and Hart, 1974]. For numerical data, the problem of high dimensionality [Ordonez and Omecinski, 2002], data sparsity [Guha et al., 1998] and noise [Bradley et al., 1998] make clustering for this type of data sets more challenging compared to categorical data.

Even though the idea of using GA to improve the performance of K-means has proposed by many researchers, most of them used numerical data sets as their test data. Thus, the main idea in this study, is to cluster the binary data sets by exploiting the advantages in GA to provide a good population and then cluster it by using computing devices taken from the efficient and effective IKM.

1.2 Objectives of the Study

This thesis proposes a new GA based clustering algorithm that focuses on clustering binary data. In particular, the aims of this thesis are to:
1. exploit the most characteristics properties of binary data by using the existing GA based clustering methods.

2. to improve the performance of GCUK to cluster the binary data and imbalanced class data sets.

3. compare the performance of GCUK-binary with \( K \)-means clustering algorithm.

4. propose a new clustering algorithm from a combination of GA and IKM that specifically focusing on binary data and imbalanced class data sets.

5. to compare the performance of GAIKM with the existing clustering methods, IKM, SKM, GCUK and NSGAII.

1.3 Significance of the Study

GA is a well-known optimization method that has been shown by many researchers to be capable of successfully carrying out the clustering task. Unlike the other traditional clustering algorithms, GA provides populations that are filled with only the best problem solutions. Each of the populations have a chance to be picked as the best solution. The continuous process to search for the best solution gives an advantage to GA to escape from being trapped at the local optima.

One advantage of GA, is it uses the entire space as a solution to the objective questions. GA can be used either for single objective optimization or multi objective optimizations. The well-known GCUK [Bandyopadhyay and Maulik, 2002b] is one of the examples of GA that use the single objective optimization and one of the algorithms that improved the performance of K-means. It was proven as an efficient and effective algorithm to cluster large data
In spite of the effectiveness of GCUK to cluster large data sets, this algorithm has a weakness as it is uses the length of the data point as a chromosome representation. This results in increased time to converge if it involves large high dimensional data sets. It should be noted that GA must iterate over every string in the chromosome and basically the longer data point (string) the longer it takes to converge. The GCUK also has the capability to provide the number of clusters during the process. However, this advantage has a drawback as GCUK can give empty clusters. We show this in the experiment in Chapter 4 (GCUK-binary).

The objective of the version of GCUK implemented in this study was to exploit the most characteristic properties of binary data by using the well-known GA based clustering algorithm. Some new improvements were added in this algorithm to suit the binary environment as GCUK has never been used to cluster these types of data sets. Based on the findings, advantages and weaknesses from the results were used to develop a new clustering method.

The other significance output in this study was to improve the performance of K-means and Incremental K-means algorithm which was proposed by [Ordonez, 2003]. Although K-means was already known as efficient and provided an easy way to interpret the results, this algorithm has its drawbacks such as: time consuming, easily trapped at the local optima, extremely dependent on the initial values and sensitive to outliers. To overcome these problems, some researchers proposed an Incremental $K$-means (IKM) to enhance the performance of the standard $K$-means. There are varieties of IKM based on the objectives of the studies. Ordonez [2003] for example, proposed an IKM that specifically improves the performance
of the standard $K$-means to cluster the binary data streams. His study proved that this algorithm is efficient and effective when compared with the Scalable $K$-means and On-line $K$-means. However, in some cases, where the data sets have some issues like sparsity and high dimensionality, the Scalable $K$-means slightly outperformed the IKM.

Based on this weakness, this study used the advantages and the efficiency of the IKM from the idea of Ordonez [2003] and combine with the GA. Single objective optimization was chosen as from the findings and results, there are some gaps from this method that can be improve in this study.

The method used in this study has been used by other researchers before, and many of them have proposed a new method based on the improvement from the original method. Nevertheless, this study is still important and contributes something new. Apart from the different types of data sets that were used, the other contribution is the application to imbalanced class data sets in this study which have never been tested with GCUK. The findings and results from this study can provide a new reference to future research involving GA and imbalanced class data.

1.4 Scope and Limitation of the Study

There are many efficient GA clustering algorithms that have been proposed by researchers, but in this study a well-known existing GA based clustering algorithms was applied; Genetic Clustering for Unknown $K$ (GCUK) proposed by Bandyopadhyay and Maulik [2002b]. This algorithm has been used for numerical data sets and has never been applied to binary data
or imbalanced class data sets, except in an improvement of the GCUK method by Lin et al. [2005].

The data involved in this study are four data sets that were retrieved from the UCI Machine Learning database [Lichman, 2013] and two imbalanced data sets; Christchurch Road Traffic Accidents (CRTA) and United Kingdom Road Traffic Accidents (URTA). All the data sets from the UCI Machine Learning database contain categorical data which were then recoded into binary attributes.

This study does not focus on the natural clusters for each of the variables as the intention is to see the performance of the proposed method when compared to other clustering methods (standard K-means, GCUK, NSGAI, IKM and SKM). The time taken to converge by the proposed method was not recorded in this study and is suggested for future research.

1.5 Overview of Thesis

The aim of this thesis is to propose a new clustering method that can be applied specifically for binary and imbalanced class data sets. It is organised as follows:

**Chapter 2** provides a brief explanations on GA.

**Chapter 3** provides a literature reviews on GA based clustering.

**Chapter 4** implementation of GCUK that reflect some modifications on the data representation to suit the binary environment. Here, the performance of GA to imbalanced data was tested too. These results, have been discussed at the conferences at the Otago Uni-
versity, University of Wellington and University of Auckland. A paper that explained
some part of the results has been submitted to International Conference on Education,
Mathematics and Sciences 2016 (ICEMS2016) and The Asian Mathematical Conference
(AMC 2016 Bali), Bali, Indonesia.

Chapter 5 introduces the new proposed GA based clustering method. This is a combination
of simple GA with the IKM and Scalable K-means. A part of the analysis of the
proposed method was given in Saharan and Baragona [2013].

Chapter 6 concludes this thesis by summarizing all the results from the study and discussing
all the possible extensions from the modified and proposed method.
Chapter 2

Overview of Genetic Algorithms

2.1 Introduction of Genetic Algorithms

GAs are one of the best ways to solve a problem for which little is known. These algorithms can solve both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution. GAs work within a search space that contains all feasible solutions. Each point in the search space represent one feasible solution and each of these feasible solutions will be marked by its fitness value. The set of solutions is called a population. One solution is taken from the population \((P)\) to form a new population \((P')\). A strong possible solution will survive to the next generation, but weak solutions will be eliminated from the population. The critical problem in the GAs process is to assign the fitness function and the initialization of the starting point in the search space [S.N. Sivanandam and S.N. Deepa, 2008].

Before we can further discuss GAs, there are some of the terms that are usually used in GAs literature. These terms are shown in Table 2.1:
Table 2.1: Terms in Genetic Algorithms

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>String</td>
<td>The individual part to form the chromosome, e.g: Chromosome: 43571 : String: 4, 3, 5, 7, 1</td>
</tr>
<tr>
<td>Chromosomes</td>
<td>A set of strings and one chromosome represent one solution</td>
</tr>
<tr>
<td>Population</td>
<td>The set of the chromosomes</td>
</tr>
<tr>
<td>Fitness</td>
<td>The value that measured how far or close the chromosomes from the solution.</td>
</tr>
<tr>
<td>Fitness Function</td>
<td>A particular type of objective function that is used to summarize and measure how close a given design solution is to achieving the set aims.</td>
</tr>
<tr>
<td></td>
<td>This is the problem specific in the GA and very crucial.</td>
</tr>
</tbody>
</table>

The solutions in the search space are nothing without the process of the combination and crossover. The new population will hopefully contain better solutions from the current population and this is only done after the two solutions are combined. Furthermore, to make it more unique, some of the characters in the solutions will be changed or mutated with a very small probability. The objective of these two operators is to make sure that the new population will be fitter compared to the old population [Goldberg, 1989, S.N. Sivanandam and S.N. Deepa, 2008].

GA is an iterative process of three major operators of selection, crossover and mutation as illustrated in Figure 2.1. The process of GA iterations will be continued until the stopping criterion satisfied.
GA applies an artificial process of evolution to construct a robust search which requires minimal information. It starts with the initialization of the population or the potential solutions of the problems. This initialization is represented by the chromosomes which is a set of genes, with each gene carrying the characteristics of the data set. These chromosomes have their own fitness values depending on the objectives of the clustering, so it is very important to determine the right objective function(s). The process of evolution or building a new generation is a series of selection, crossover and mutation processes. The selection process, is based on the value of the fitness. If the chromosome is fit enough (has achieved the objective
function), then it can be selected and the weakest chromosomes will be eliminated from the population. Then, these chromosomes will go into the mating pool and two chromosomes will be randomly selected to mate and combine some of their genes to produce a new offspring. A small change (mutation) of a gene in a child may happen with a lower probability. The idea is to make the child more unique or have a high quality compared with their parent. The mutation process also has a role in preventing the algorithm from drop into the local minimum. Thus, as new generations develop, the quality of the chromosomes or the possible solutions are also increased. This process is repeated until some criteria are met or some predefined number of generations is achieved.

2.2 Encoding

The first step in GA is to choose the suitable encoding to represent the chromosomes or the possible solutions. The users will have to encode their problem’s variables into the chromosomes environment. Usually GA uses a binary encoding, but many researchers such as Bandyopadhyay and Pal [2007], Baragona et al. [2006], Jie et al. [2003], Maulik and Bandyopadhyay [2000], Mukhopadhyay et al. [2009] used real numbers for encoding in GA. The binary encoding only contains 0’s (absent) and 1’s (present) and it was first used because of its relative simplicity [Goldberg, 1989]. These chromosomes will have a length of $l$ that represent solutions for every objective. The number of solutions that can be obtained is then clearly $2^l$.

There are three methods of encoding: point-based encoding, centre-based encoding and locus-based adjacency. Of these point-based and centre-based encoding are the methods that widely


Villar, P., Fernández, A., Carrasco, R. a., and Herrera, F. (2012). Feature Selection and Granularity Learning in Genetic Fuzzy Rule-Based Classification Systems for Highly Im-


