Clustering for Binary Data Sets by Using Genetic Algorithm-Incremental K-means

To cite this article: S Saharan et al 2018 J. Phys.: Conf. Ser. 995 012038

View the article online for updates and enhancements.

Related content
- A heuristic approach to handle capacitated facility location problem evaluated using clustering internal evaluation
  G R Sutanto, S Kim, D Kim et al.
- A clustering algorithm based on maximum entropy principle
  Yang Zhao and Fangai Liu
- PERFORMANCE ANALYSIS OF COMBINED METHODS OF GENETIC ALGORITHM AND K-MEANS CLUSTERING IN DETERMINING THE VALUE OF CENTROID
  Putra Adya Zizwan, Muhammad Zarlis and Erna Budhiarti Nababan
Clustering for Binary Data Sets by Using Genetic Algorithm-Incremental K-means

S Saharan¹, R Baragona², M E Nor¹, R M Salleh¹ and N M Asrah¹

¹Department of Statistics and Mathematics, University of Tun Hussein Onn Malaysia, 86400 Johor Malaysia
²Department of Communication and Social Research, Sapienza University of Rome, Italy

Abstract. This research was initially driven by the lack of clustering algorithms that specifically focus in 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 Algorithms (GA). For the purpose of this research, GA was combined with the Incremental K-means (IKM) algorithm to cluster the binary data streams. In GAIKM, the objective function was based on a few sufficient statistics that may be easily and quickly calculated on binary numbers. The implementation of IKM will give an advantage in terms of fast convergence. The results show that GAIKM is an efficient and effective new clustering algorithm compared to the clustering algorithms and to the IKM itself. In conclusion, the GAIKM outperformed other clustering algorithms such as GCUK, IKM, Scalable K-means (SKM) and K-means clustering and paves the way for future research involving missing data and outliers.

1. Introduction
The increasing volume of data that were collected by individuals, organizations or either firms has triggered the initiative to process and analyse this type of information. Many researchers have been proposed a new statistical method such as discriminant analysis, multi-dimensional scaling, clustering and classification. Clustering and classification is one of the most well-known statistical techniques used to process this large volume of data. Clustering mostly known as unsupervised classification as the prior knowledge of the number of clusters, \(K\) maybe known or not. [1] has stated that this technique can be consider as one of most challenging problem in machine learning. Until today, there are more than a thousand new techniques have been proposed and there is always a continuous new idea from the researchers.

One of the most popular clustering algorithms is K-means which is computationally efficient. However, it still has a few critical weaknesses which is totally dependent on the initial cluster centres and very sensitive with the outliers. [2,3,4] found that K-means can be easily to be trapped in a local minimum and time consuming when applied to large volume data [5]. To improve the performance of K-means, [3, 4, 5, 6] have been proposed a new algorithm, Incremental K-means (IKM).

Thus, in this study we are combining the idea of IKM with a Genetic Algorithms (GA) to improve the performance of the IKM to cluster the large data sets known as GAIKM [15]. However, we are focusing on clustering the binary data only to validate this new proposed algorithm. Unlike the other quantitative, binary data does not have a noise problem and can be represent as a categorical data set. Furthermore, the development of clustering algorithm that focusing on binary data is not grown as tremendously like numerical data. Clustering for numerical data always be more challenging compared to categorical data as it has many problems to discover such as high dimensionality [9], sparsity [10] and noise [11]. However, the problem for categorical data is not clear as the problem for
numerical data sets [8]. Even though the challenges of dealing with binary data is not same as numerical, this data sets are interesting and useful for some variety reasons.

2. Clustering Algorithms
Clustering or unsupervised classification is a technique to group a data points into $K$ clusters which each of the characteristics within the clusters will have a high similarity and different as possible from the other characteristics in the other clusters.

Given a set of $n$ variables, $Y = y_1, y_2, ..., y_n$ to be clustered with each of these $Y_i \in R^P$ is an attribute vector used to describe the variables. These variables will be clustered into a set of clusters, $K = k_1, k_2, ..., k_K$. The clusters are disjoint $K_i \cap K_j = \emptyset$ for $i \neq j$. The numbers of $K$ may be known a priori or not. Let $\overline{y}_i$ be the centre of the cluster $K_i$. The main goal of clustering is to find the minimum distance between the $y_i$ to the closest centre $\overline{y}_k$ as follows:

$$\sum_{k=1}^{K} \sum_{y_k \in K_k} ||y_i - \overline{y}_k||^2$$

Due the weaknesses of K-means to cluster large data sets, several researchers have proposed an Incremental K-means (IKM). This method is more efficient and can handle a large volume of data. However, IKM still facing some problem when dealing with larger $K$, where the process of clustering can be slowed down. It is because IKM needs to find all possible places to insert the new clusters. In some cases, Scalable K-means (SKM) tend to be more efficient compared to IKM when applied to data that have a high dimensional and very sparse matrix.

Another promising algorithm to handle a large data set is Genetic Algorithms (GA). This technique was proposed by John Holland and his colleagues in the early of 1970’s. GA was inspired by the process of biological evolution; selection, crossover and mutation process [7]. The advantages of GA are the ability to search the best population (solution) and escaping from trapping at the local optima. This makes GA more efficient compared to the other optimization methods.

Thus, in this study, GA will be used to provide a good population/solution and IKM will be acting as a clustering tool to cluster a data provided from GA.

3. Genetic Algorithms with Incremental K-means (GAIKM)
Descriptions about Incremental K-means and Genetic Algorithm is presented in this section.

3.1 Incremental K-means (IKM)
In this study, we are combining the IKM that was proposed by [3] that specifically deals with large binary data sets. This IKM have some differences between the online-K-means and standard K-means where it will not iterate until convergence. Only the values of $N$ and $M$ will be updated at each step which makes this algorithm is computationally efficient. Let $n$ be the number of cases, and $K$ is the number of clusters. Then, $N_j, j = 1, ..., k$ is the number of cases that are assigned to cluster $j$ while $M_{ij}$ is the sum of the instances equal to 1 recorded for the factor $i$. Where $i$ is for all the cases that are assigned to cluster $j$. While the centers, $C = \{C_j, j = 1, ..., k\}$ the variance $R = \{R_j, j = 1, ..., k\} of each factor in each cluster, where $R_j = (v_{1j}, ..., v_{pj})'$, and the center weight $W = \{W_j, j = 1, ..., k\}$ need to be updated only every $n/m$ steps if only if $n/m$ is an integer. The parameter that controls the updating frequency is namely as $m$. [3] was suggested that to ensure the best results the value of $m$ should be calculated as $m = \sqrt{n}$. Such choice, however is viable only if a batch computation where the number of records in the database is known in advance to be equal to $n$. For the complexity computation, the variants of K-means designed for clustering binary data require a computation time at least of order approximately $O(Tkn)$, where $T$ is the average over all cases of how many factors out of d have recorded value 1. The space required for storing the matrices $M$ and $C$ is of order $O(dk)$.
while it is $O(k)$ for the vector $N$ and $W$. The space for $R$ can be saved because it can be straightforwardly computed from $C$. However, an additional space is required to hold temporarily a certain case in a buffer.

### 3.2 Data Representation

Let $n \times d$ is a matrix of binary represents by $D$, and $n$ is the number of cases and $d$ is the number of variables. If the data points, $t_i; i = 1, \ldots, n$ is assumed as d-dimensional column vector whose elements are the factor values recorded at the occurrence of the case $i$, then

$$
D = \begin{pmatrix}
  t_1 \\
  t_2 \\
  \vdots \\
  t_n
\end{pmatrix},
$$

$t_i = (a_{i1}, a_{i2}, \ldots, a_{id})'$, $a_{ij} \in \{0,1\}$

'$'$ means a transpose of a vector or matrix. An example of such a matrix is a follow:

$$
D = \begin{pmatrix}
  0 & 1 & 1 & \ldots & 1 & 1 & 1 \\
  1 & 0 & 0 & \ldots & 1 & 1 & 0 \\
  \vdots \\
  0 & 1 & 0 & \ldots & 1 & 0 & 1
\end{pmatrix}
$$

The number at the intersection of row (case) $i$ and column (factor) $l$ is 1, if the factor $l$ has been recorded concomitant to this factor $i$ and 0 otherwise.

### 3.3 GAIKM

#### 3.3.1 Fitness Function

The critical thing in GA is the choices of fitness function where it can influence the results of the population. For GAIKM, the fitness function is to minimize the sum of intra-cluster variances, and needed only the sufficient statistics $N_j = \sum_{i \in t_j} 1$, the number of cases in cluster $j, j = 1, \ldots, k$ and $M_j$, is the vector whose entries are the sums of 1 for each factor of the data points in cluster $C_j$. The variance of the proportion $C_j = M_j / N_j$ is equal to $R_j = C_j (1 - C_j)$, and the intra-cluster variance can be written as follows

$$
q(R,W) = \sum_{j=1}^{k} W_j \sum_{i=1}^{d} R_{ij}
$$

Where $W_j = N_j / n$. The fitness function for GAIKM in this study is in Equation 1, where to indicate the better results is the minimum value of $qRW$.

#### 3.3.2 Processes in GAIKM

There are three processes in GA, selection, crossover and mutation. Each for this process, has their selection which method is more suitable for the data used.

In GAIKM, for the selection process, we chose the Stochastic Universal Sampling (SUS) as this method is known to be unbiased compared to Roulette Wheel Selection [12]. Then, the one-point crossover was applied to the data set. In crossover process, two chromosomes (parent) were chosen randomly and one single line was draw between the data set to separate into two sections (head) and tail. This process is used to exchange parts of these parents to produce another two new chromosomes. The new chromosomes will possibly obtain a better and fitter chromosome compared to their parent.
The last process is mutation where it will prevent the new population from trapping at the local optima. This process also keeps the diversity in the new population. In this study, the uniform random mutation was used with the value of the probability will be range between 0.7, 50, 0.9.

The procedure of the GAIKM was simple. First, \( s \) sets of cluster centres were generated at random and were encoded as a population of \( s \) chromosomes. Then, the stream of the cases \( t_1, t_2, \ldots, t_n \) was inserted to the algorithm in a group of \( h \) cases at a time. If \( h = 1 \) was chosen, then each case was input to the algorithm one at a time in the sequence. If \( h = n \), then a batch execution was performed. These \( h \) cases were assigned to the closest centre using the distance \( \delta(t_i, C_j) \). In GAIKM, the distance of the data points, \( t_i \) to the closest centre was calculated by using the Hamming distance.

However, unlike the IKM, in the GAIKM each case had a chance to be assigned in \( s \) different cluster systems. This makes GA differ from the other clustering methods that consider only a single one as IKM. As the new cases were input into the algorithm, the steps of the GA were developed according to the three operators of selection, crossover and mutation processes.

Then a new population of \( s \) chromosomes, each one corresponding to a partition of the data points in the current database, replaced the old population. At the end, \( s \) different partitions of all \( n \) data points in the population were available.

### 4. Result and findings

Four binary data sets were tested in this study. To compare the performance of GAIKM, the other clustering methods; Genetic Clustering for Unknown K (GCUK), Incremental K-means (IKM), Scalable K-means (SKM) and standard K-means algorithm were used with the same value of K.

The original data sets are from the categorical data which have been changed into the binary form. Three different real life binary data sets were used to show the effectiveness of the GAIKM. These data were the Nursery [13], Car Evaluation [13] and Bank Marketing [14] data sets. Christchurch Road Traffic Accidents (CRTA) data set was taken from the New Zealand Transport Agency (NZTA).

| Data set   | GAIKM     | IKM     | SKM     | K-means | GCUK  |
|------------|-----------|---------|---------|---------|-------|
| Nursery    | 2166.76   | 2256.1  | 2258.6  | 2209.4  | 2328.4|
| Car        | 305.95    | 312.49  | 320.91  | 322.89  | 311.64|
| Bank       | 4638.09   | 5475.9  | 4884.8  | 5130.7  | 5118.7|
| CRTA       | 562.89    | 1040.1  | 981.42  | 719.2   | 630.09|

From the results in Table 1, all the lowest values of the objective function were attained by the proposed method, GAIKM. These values are far lower than the IKM and SKM, showing that the GAIKM is an efficient clustering algorithm especially to cluster the binary data sets.

### 5. Conclusion

In this study, GA was chosen to combine with IKM to enhance the performance of IKM. The implementation was proven to be efficient in the previous chapter. The GA was acted as a tool to provide a group of the best populations (solutions) in the entire population. On the other hand, IKM acted as tools for the clustering process.

However, unlike the IKM, in the GAIKM, each case has a chance to be assign in a different cluster system. This process is different than IKM which only gave a single one solution. Elitism used in GA also enhance the performance of GAIKM as it ensures that the best solutions are not eliminated during the process of iteration.
As for the algorithmic performance, the results showed that GAIKM algorithm greatly improved the IKM itself. The comparison of GAIKM with SKM, IKM, GCUK and standard K-means showed that GAIKM attained the lowest of the fitness value.

Acknowledgement
This research is supported by Short Term Grant Vot. U369 from University of Tun Hussein Onn Malaysia.

References
[1] Hruschka, E. R., Campello, R., Freitas, A. A., and de Carvalho, A 2009. Applications and Reviews, IEEE Transportations, 39(2) pp 133-155
[2] Lu, Yu., S., Fotouhi, F., Deng, Y., and Brown, S.J. (2004). In Proceedings of the 2005 ACM symposium on Applied computing. SAC’04, pages 622-623
[3] Ordonez, C. 2003. In Proceedings of the 8th ACM SIGMOD workshop on Research issue in data mining and knowledge discovery, DMKD ’03, pages 12-19
[4] Pham, D. T., Dimov, S. S., and Nguyen, C. D 2004. An incremental K-means algorithm
[5] Sanjay Chakraborty, N. N. 2011. In High Performance Architecture and Grid Computing Communications in Computer and Information Science, pages 338-341
[6] Sanjay, C., N.K., N., and Lopamudra, D 2011. International Journal of Computer Applications, 27(11) pp 14-18
[7] Goldberg, D. E. 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA
[8] Duda, R. O. and Hart, P. 1974. IEEE Transaction on Automatic Control, 19(4) pp 462-463
[9] Ordonez, C and Omiecinski, E. 2002. In Proceedings of the eleventh international conference on Information and Knowledge Management, CIKM ’02, pages 590-599
[10] Guha, S., Rastogi, R., and Shim, K. 1998. ACM SIGMOD Record, 27(2) pp 73-84
[11] Bradley, P. S., Fayyad, U., Reina, C., and Paul S. Bradley, Usama M. Fayyad, C. A. R 1998. American Association for Artificial Intelligence, pages 9-15.
[12] Baker, J. E. 1987. In Proc of the Second International Conference of Genetic Algorithms and Their Application. Pages 14-21
[13] Lichman, M. 2013. UCI Machine Learning Repository
[14] Sergio Moro, Paulo Cortez and Paulo Rita 2014. A Data-Driven Approach to Predict the Success of Bank Telemarketing. Decision Support Systems. 62 pp 22-31
[15] S Saharan, R Baragona 2014. Far East Journal of Theoretical Statistics 45(1) pp 67-89