International Journal of Enterprise Computing and Business Systems (Online)

http://www.ijecbs.com

Vol. 1 Issue 1 January 2011

A SURVEY ON PARTITION CLUSTERING ALGORITHMS

S. Anitha Elavarasi

Lecturer,

Department of CSE,

Sona College of Technology,

Salem-636 005, India

E-mail : anishaer@gmail.com

Dr. J. Akilandeswari

Professor and Head

Department of IT

Sona College of Technology

Salem-636 005, India

E-mail : akila_rangabashyam@yahoo.co

Dr. B. Sathiyabhama

Professor and Head,

Department of CSE,

Sona College of Technology,

Salem-636 005, India

E-mail : sathya674@yahoo.co.in

Abstract

Learning is the process of generating useful information from a huge volume of data. Learning can be classified as supervised learning and unsupervised learning. Clustering is a kind of unsupervised

learning. A pattern representing a common behavior or characteristics that exist among each item can be generated. This paper gives an overview of different partition clustering algorithm. It describes about the general working behavior, the methodologies followed on these approaches and the parameters which affects the performance of these algorithms.

Keywords: Clustering, Supervised Learning, Unsupervised Learning

1. Introduction

Clustering is a process of grouping objects with similar properties. Any cluster should exhibit two main properties; low inter-class similarity and high intra-class similarity. Clustering is an unsupervised learning i.e. it learns by observation rather than examples. There are no predefined class label exists for the data points. Cluster analysis is used in a number of applications such as data analysis, image processing, market analysis etc. Clustering helps in gaining, overall distribution of patterns and correlation among data objects [**1**]. This paper describes about the general working behavior, the methodologies to be followed and the parameters which affects the performance of the partition clustering algorithms.

This paper is organized as follows; section 2 gives an overview of different clustering algorithms. In section 3 various partition clustering algorithms, the methodology applied on these algorithms and the parameter which has the impact on the efficiency of these algorithms are described. Finally in section 4 the conclusions are provided.

2. Clustering Overview

Clustering is a division of data into groups of similar objects. [3] Clustering algorithm can be divided into the following categories:

1. Hierarchical clustering algorithm
2. Partition clustering algorithm
3. Spectral clustering algorithm
4. Grid based clustering algorithm
5. Density based clustering algorithm

2.1 Hierarchical Clustering Algorithm

Hierarchical clustering algorithm groups data objects to form a tree shaped structure. It can be broadly classified into agglomerative hierarchical clustering and divisive hierarchical clustering. In agglomerative approach which is also called as bottom up approach, each data points are considered to be a separate cluster and on each iteration clusters are merged based on a criteria. The merging can be done by using single link, complete link, centroid or wards method. In divisive approach all data points are considered as a single cluster and they are splited into number of clusters based on certain criteria, and this is called as top down approach. Examples for this algorithms are LEGCLUST [23], BRICH [20] (Balance Iterative Reducing and Clustering using Hierarchies), CURE (Cluster Using REpresentatives) [21], and Chemeleon [1].

2.2 Spectral Clustering Algorithm

Spectral clustering refers to a class of techniques which relies on the Eigen structure of a similarity matrix. Clusters are formed by partition data points using the similarity matrix. Any spectral clustering algorithm will have three main stages [24]. They are

1. Preprocessing: Deals with the construction of similarity matrix.
2. Spectral Mapping: Deals with the construction of eigen vectors for the similarity matrix
3. Post Processing: Deals with the grouping data points

The following are advantages of Spectral clustering algorithm:

1. Strong assumptions on cluster shape are not made.
2. Simple to implement.
3. Objective does not consider local optima.
4. Statistically consistent.
5. Works faster.

The major drawback of this approach is that it exhibits high computational complexity. For the larger dataset it requires $O(n^3)$ where n is the number of data points [17]. Examples for this algorithms are SM (Shi and Malik) algorithm, KVV (Kannan,Vempala andVetta) algorithm, NJW (Ng, Jordan and Weiss) algorithm [23].

2.3 Grid based Clustering Algorithm

Grid based algorithm quantize the object space into a finite number of cells that forms a grid structure [1].Operations are done on these grids. The advantage of this method is lower processing time. Clustering complexity is based on the number of populated grid cells and does not depend on the number of objects in the dataset. The major features of this algorithm are:

1. No distance computations.
2. Clustering is performed on summarized data points.
3. Shapes are limited to union of grid-cells.
4. The complexity of the algorithm is usually O(Number of populated grid-cells)

STING [1] is an example for this algorithm.

2.4 Density based Clustering Algorithm

Density based algorithm continue to grow the given cluster as long as the density in the neighborhood exceeds certain threshold [1]. This algorithm is suitable for handling noise in the dataset. The following points are enumerated as the features of this algorithm.

1. Handles clusters of arbitrary shape
2. Handle noise
3. Needs only one scan of the input dataset.
4. Needs density parameters to be initialized.

DBSCAN, DENCLUE and OPTICS [1] are examples for this algorithm.

3. Partition Clustering Algorithm

Partition clustering algorithm splits the data points into k partition, where each partition represents a cluster. The partition is done based on certain objective function. One such criterion functions is minimizing square error criterion which is computed as,

$$E = \sum \sum || p - m_i ||^2 \quad (1)$$

where p is the point in a cluster and m_i is the mean of the cluster. The cluster should exhibit two properties, they are (1) each group must contain at least one object (2) each object must belong to exactly one group. The main draw back of this algorithm [3] is whenever a point is close to the center of another cluster, it gives poor result due to overlapping of data points.

3.1 Genetic K-Means Clustering Algorithm for Mixed Numeric and Categorical Dataset

Genetic K-means algorithm (GKA) is the combination of genetic algorithm and K-means algorithm. GKA is suitable for numeric data set which can be overcome by Genetic K-means clustering algorithm for mixed dataset [4]. It uses an enhanced cost function to handle the categorical data.

3.1.1 Methodology

1. Objective function: Defines the objective function (φ) for mixed data type

$$\varphi = \sum_{i=1}^{n} V(d_i, c_j) \quad (2)$$

where $V(d_i, c_j)$ distance of data d_i from the closest center c_j

2. Selection: Calculates the probability distribution function (P_z) using the fitness value $F(S_z)$.

$$Pz = \frac{F(Sz)}{\sum_{z=1}^{Z} F(Sz)} \quad (3)$$

where S_z denotes the solution.

3. Mutation: Mutation is performed to achieve global optimum using the probability distribution function P_k.

$$Pk = \frac{1.5 * d_{max}(X_n) - d(X_n, C_k) + 0.5}{\sum_{k=1}^{K}(1.5 * d_{max}(X_n) - d(X_n, C_k) + 0.5} \quad (4)$$

where $d(X_n, C_k)$ is the Euclidean distance between pattern X_n and the centroid C_k , $d_{max}(X_n)$ represents the maximum distance for the pattern X_n

4. Finally convergence is obtained by applying K-means operator.

The author tested the algorithm using the Iris, Vote and Heart Diseases dataset taken from the UCI repository and judge the quality of the clusters obtained.

3.2 SCALE Algorithm

SCALE [5] is a framework designed for the transactional dataset. The similarity measures used for the categorical data items for transaction dataset are Weighted Coverage Density (WCD). It is the improvised measure of coverage density (CD). WCD preserves as many frequent items as possible within the cluster. It controls the overlapping of items among clusters. Data items are filled in a 2D grid. The problem of

clustering is to minimize the unfilled number of cells with appropriate number of partition. Once the clusters are formed they are evaluated using two measures (i)LISR (Large Item Size ratio) and (ii) AMI (Average pair cluster Merging Index). LISR makes use of the percentage of large items in the clustering results to evaluate the clustering quality. AMI applies coverage density to indicate the structural difference between clusters.

3.2.1 Methodology

1. Sampling: Huge volume of data in the transactional database are sampled into smaller groups
2. Cluster structure assessment: Generate the candidate cluster based on the sample data set.
3. Clustering: Done in three phases: cluster structure assessment phase, WCD based initial cluster assessment phase and iterative clustering refinement phase. Weighted Coverage Density is define as:

$$WCD = \frac{\sum_{j=1}^{N_k} Occur(I_{jk}) \geq}{S_k \times N_k} \quad (5)$$

where Occur(I_{jk}) represents the occurrence of the item I_{jk}, N_k represents the number of transaction, S_k represents the sum of occurrence of all items in cluster C_k.

4. Evaluation: The algorithm is evaluated by two measures. LISR (Large Item Size ratio) and AMI (Average pair cluster Merging Index)

$$LISR = \sum_{k=1}^{K} \frac{N^k}{N} \times \frac{\sum_{j=1}^{M^k} Occur(I_{kj}) \times IN(Occur(I_{kj})) \geq \tau \times N_k}{S_k} \quad (6)$$

$$AMI = \frac{1}{K} \sum_{i=1}^{K} D_i \quad (7)$$

Where $IN(Occur(I_{kj}))$ represents indicator function, τ represents minimum support, M^k represents number of items, and D_i represents the dissimilarity measures.

The author used two synthetic dataset and three real dataset (Zoo, mushroom and retail). The results of SCALE are compared with CLOP algorithm and found to be better with respect to the following features (1) time spent on computing values for tuning parameter is reduced, (2) handling larger dataset (3) producing good quality cluster for domain specific measure.

3.3 Harmony K-Means algorithm

Harmony K-means [6] algorithm is based on the Harmony search optimization method which is proven using the finite Markov chain theory. It is a meta heuristic algorithm and it achieves global optimum. The advantages of this algorithm are,

(i) Needs only few mathematical prerequisite,

(ii) Uses stochastic random searches,

(iii) Treats continuous variable without any loss of precision,

(iv) Does not demand for initializing the decision variable

(v) Encoding or decoding is not performed for the decision variable.

3.3.1 Methodology

1. Initialize the parameters Harmony Memory Size (HMS), Pitch Adjusting Rate (PAR), Number of Improvisation (NI), Harmony Memory Considering Rate (HMCR).
2. Generate a new harmony vector
3. Update the harmony memory
4. Check for the stopping criterion.
5. Terminate the process when maximum numbers of improvisations are reached.

The author experiment the algorithm using five dataset (three from TRECO, one from DMOZ and one from newsgroup) and compared the result with the K-means using the Euclidean and cosine correlation measures. He also compared with the GA, PSO, Mises-Fisher Generative Model based algorithm (GM) The main focus on the quality and speed of convergence.

3.4 Initializing K–Means using Genetic Algorithm

Initializing K-means using Genetic algorithm [8] overcomes the initializing problem of the K-means algorithm by using the genetic algorithm. This approach solves the blind search problem of the K-means algorithm.

3.4.1 Methodology

1. Initialization of initial population P_0

2. Repeat the step 3 through step 6 until termination condition is met
3. Crossover or recombination is done
4. Mutation is performed
5. Evaluation is performed
6. Apply K-means algorithm.

The author tested the algorithm using four different dataset chosen from MATLAB. The algorithms used for comparison were K-means, Genetic algorithm and Genetic algorithm initializing K-means (GAIK). The trade off between average error rate and average time between these algorithms were listed by the author.

3.5 Cluster Ensemble approach for mixed data

Dataset with mixed data type are common in real life. Cluster Ensemble [9] is a method to combine several runs of different clustering algorithm to get a common partition of the original dataset. In the paper divide and conquers technique is formulated. Existing algorithm use similarity measures like Euclidean distance which gives good result for the numeric attribute. This will not work well for categorical attribute. In the cluster ensemble approach numeric data are handled separately and categorical data are handled separately. Then both the results are then treated in a categorical manner. Different types of algorithm used for categorical data are K-Modes, K-Prototype, ROCK [22] and squeezer algorithm. In K-Mode the total mismatch of categorical attributes of two data record is projected. The Squeezer algorithm yields good clustering result, good scalability and it handles high dimensional data set efficiently.

3.5.1 Methodology

1. Splitting of the given data set into two parts. One for numerical data and another for categorical data
2. Applying any one of the existing clustering algorithms for numerical data set
3. Applying any one of the existing clustering algorithms for categorical data set
4. Combining the output of step 2 and step 3
5. Clustering the results using squeezer algorithm

The credit approval and cleve (heart diseases) dataset are used and measures the cluster accuracy and cluster error rate. The cluster accuracy ‘r‘is defined by

$$r = \frac{\sum_{i=1}^{K} a_i}{n} \quad (8)$$

where K represents number of clusters, a_i represents number of instance occurring in both the cluster i and its class and n represents number of instance in the dataset . Finally cluster error rate 'e' defined by

$$e = 1 - r \quad (9)$$

where r represents cluster accuracy. The algorithm is compared with k-prototype algorithm.

3.6 Hybrid Evolutionary algorithm

The Hybrid Evolutionary algorithm [13] is based on the combination of Ant colony Optimization and Simulated Annealing (ACO-SA). The one of the major draw backs of K-means algorithm is, the result depends on the initial choice of the cluster center. Inorder to achieve global optima the initial choice of cluster center can be chosen with the help of ACO and SA.

Ants are insects which find the shortest path between its nest and food with the help of Pheromone. Pheromone is a chemical substance deposited by ant used for the communication between them. The two most important factors used to find the shortest path are (i) the intensity of Pheromone and (ii) the path length. Annealing is a process by which the liquid freeze or metal recrystalize. Ant colony Optimization is used for finding the colony between the data points. Simulated Annealing is used as a good local search algorithm for finding the best global position by using the Cumulative Probability.

3.6.1 Methodology

1. Generation of initial population and trial intensity for the given dataset
2. Selection of best local position using simulated annealing algorithm for the i^{th} colony
3. Selection of best global position for the i^{th} colony
4. Repeat step 2 and step 3 for all the colonies.
5. Check for the convergence condition C, if convergence is satisfied, stop the process. Otherwise repeat the process from step 2. The formula for convergence is

$$C = \sqrt{\sum_{i=1}^{N} \left| X_i^{k+1} - X_i^{k} \right|^2} \quad (10)$$

where X_i represents position of i^{th} individual, k represents iteration and N represents number of objects.

The author compares ACO-SA with ACO, SA and K-means algorithm using six real life data sets (Iris, Wine, Vowel, CMC, Wisconsin breast cancer and Ripley's glass). The algorithm provides global optimum with smaller standard deviation and few functional evaluations.

3.7 Modified global K-means algorithm

The goal of the modified global k-means MGKM) [10] algorithm is to obtain global minima and to work well with the larger gene expression dataset. The author proposed modified global K-means algorithm. The paper focuses on the hard unconstrained partition clustering problem and to improve global search properties. The Global K-means computes the cluster successfully. During its first iteration the centroid for a set is computed. To compute k-partition at the k-th iteration it uses the centers of the k-1 cluster from the previous iteration. Global K-means can be applied for a smaller dataset but it is not suitable for average or larger dataset because of high computational time. The problem can be avoided by using the squared distance concept to find the closest cluster center among k-1 clusters.

3.7.1 Methodology

1. Initialization is done by selecting a tolerance value greater than zero and computing the center x^1
2. Computation of cluster center for k partition is done
3. Refinement of cluster center by using the objective function

$$f(x^1,\ldots x^k) = \frac{1}{m}\sum_{j=1}^{m} \min_{j=1,\ldots,k} \left\| x^j - a^i \right\|^2 \quad (11)$$

where x^j represents the data point, a^i represents cluster centroid of i^{th} cluster.

4. Check for stopping criteria, if not repeat the process from step 2.

$$\text{Stopping criteria} = \frac{f^{k-1} - f^{k}}{f^{k-1}} < \varepsilon \quad (12)$$

The author compares the MGKM algorithm with the multi-start K-means (MSKM) and global K-means (GKM) on six different gene expression datasets. The parameters for comparison are, number of cluster (N), CPU time (t) and cluster function (f). The MGKM outperforms the other algorithm as the number of cluster increases.

3.8 FGKA

Fast Genetic K-means algorithm (FGKA) [10] is inspired from the Genetic K-means Algorithm (GKA). Both GKA and FGKA achieve global optimum, still FGKA runs much faster than the GKA. The algorithm starts with the initialization phase with an initial population P_0. Evolution of next population will be done with the selection, mutation and K-means operators which are done in sequence until a termination condition is met. The objective of selection operation is to find the population having greatest fitness value and assign smaller fitness value for illegal strings. The objective of mutation operation is to achieve global optimum. It generates positive probability and makes the pattern to move closer to the cluster and provides a legal solution.

3.8.1 Methodology

1. Generation of Initial population
2. Check for termination condition; if the condition is reached go to step 6, else repeat step 3 through step 6.
3. Apply objective function to minimize the Total Within-Cluster Variation (TWCV) and is defined by

$$TWCV = \sum_{n=1}^{N} \sum_{d=1}^{D} X_{nd}^2 - \sum_{k=1}^{K} \frac{1}{zK} \sum_{d=1}^{D} SF\ kd^2 \quad (13)$$

where X_{nd} represent the d^{th} feature of pattern X_n, SF_{kd} is the sum of the d^{th} feature of all the patterns in G_k, D denotes the dimension and N denotes the pattern

4. Apply proportional selection operator P_z defined as,

$$P_z = \frac{F(Sz)}{\sum_{z=1}^{Z} F(Sz)} \quad (14)$$

where S_z denoted the solution, $F(Sz)$ denotes the fitness value.

5. Selection of probability distribution (P_k) is done on Mutation operator.

$$P_k = \frac{1.5 * d\max(Xn) - d(Xn,Ck) + 0.5}{\sum_{k=1}^{K}(1.5 * d\max(Xn) - d(Xn,Ck) + 0.5} \quad (15)$$

where $d(X_n, C_k)$ is the Euclidean distance between pattern X_n and the centroid C_k , $d_{max}(X_n)$ represents the maximum distance for the pattern X_n

6. K-means operator is added as the last step to speed up the convergence process.
7. Final Cluster formation.

The advantage of FGKA over GKM is, it provides efficient TWCV calculation, illegal strings are eliminated and the mutation operation is simplified. The author compares FGKA with GKA and K-means using two dataset fig2data and chodata. FGKA runs 20 times faster than GKA

4. Conclusion

The paper describes different methodologies and parameters associated with partition clustering algorithms. The drawback of k-means algorithm is to find the optimal k value and initial centroid for each cluster. This is overcome by applying the concepts such as genetic algorithm, simulated annealing, harmony search techniques and ant colony optimization.

References

1. Jiawei Han, Micheline Kamber, "Data Mining Concepts and Techniques" Elsevier Publication.
2. P.J. Flynn, A.K. Jain, M.N. Murty,1999. Data Clustering: A Review. ACM Computing Surveys, vol. 31, no. 3: 264-323.
3. P. Berkhin,2002. Survey of Clustering Data Mining Techniques. Ttechnical report, Accrue Software, San Jose, Cailf.
4. Dharmendra K Roy and Lokesh K Sharma, 2010. Genetic K-Means clustering Algorithmfor mixed numeric and categorical data. International journal of Artificial Intelligence& ApplicationsVol1 No2.
5. K. Chen, L. Liu, H. Yan, Z. Yi, 2010. SCALE: a scalable framework for efficiently clustering transactional data. Data Mining Knowledge Discovery. 20:1–27
6. H. Abolhassani, M. Mahdavi, 2009. Harmony K-means algorithm for document clustering. Data Mining Knowledge Discovery.18:370–391.
7. BB Firouzi, T Niknam, M Nayeripour, 2008. An efficient hybrid evolutionary algorithm for cluster analysis. World Applied Sciences Journal:300-30.
8. B Al-Shboul, SH Myaeng, 2009. Initializing K-Means using Genetic Algorithms. World Academy of Science, Engineering and Technology.
9. S Deng, Z He, X Xu, 2005. Clustering mixed numeric and categorical data: A cluster ensemble approach. Arxiv preprint cs/0509011.
10. AM Bagirov, K Mardaneh, 2006. Modified global k-means algorithm for clustering in gene expression data sets. ACM International Conference Proceeding on Intelligent systems for bioinformatics - Volume 73**:** 23-27

11. Y Deng, Lu, F Fotouhi, S Lu, 2004. FGKA: a Fast Genetic K-means Clustering Algorithm. Proceedings of the ACM symposium on Applied computing: 622 - 623
12. C Ding, X He, 2004. K-means clustering via principal component analysis. ACM twenty-first International Conference Proceeding on Machine learning : 29
13. B.Bahmani Firouzi, T.Niknam, M.Nayeripour, 2008. A New Evolutionary Algorithm for Cluster Analysis. International Journal of Computer Science.
14. Michael Laszlo Sumitra Mukherjee**,** 2007. A genetic algorithm that exchanges neighboring centers for k-means clustering. Pattern Recognition Letters**,** Volume 28: 2359-2366
15. K. Krishna and M. Murty, 1999. Genetic K-Means Algorithm. IEEE Transactions on Systems, Man, and Cybernetics vol. 29, NO. 3: 433-439.
16. Von Luxburg U, 2007. A Tutorial on Spectral Clustering, Statistics and Computing, 17(4):395-416.
17. Cai X. Y. et al, 2008. Survey on Spectral Clustering Algorithms. Computer Science:14-18
18. F. R. Bach and M. I. Jordan, 2006. Learning spectral clustering, with application to speech separation . Journal of Machine Learning Research : 1963–2001.
19. L. Huang, M. I. Jordan, D. Yan, 2009. Fast approximate spectral clustering. Technical report, Department of Statistics, UC Berkeley.
20. M. Livny, R.Ramakrishnan, T. Zhang, 1996. BIRCH: An Efficient Clustering Method for Very Large Databases. Proceeding ACMSIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery :103-114.
21. S. Guha, R. Rastogi, and K. Shim, 1998. CURE: An Efficient Clustering Algorithm for Large Databases. Proc. ACM Int'l Conf. Management of Data : 73-84.
22. S. Guha, R. Rastogi, and K. Shim, 2000. ROCK: A Robust Clustering Algorithm for Categorical Attributes. Information Systems, vol. 25, no. 5 : 345-366.
23. Santos, J.M, de Sa, J.M, Alexandre, L.A , 2008. LEGClust- A Clustering Algorithm based on Layered Entropic subgraph. Pattern Analysis and Machine Intelligence, IEEE Transactions : 62-75.
24. M Meila, D Verma,2001. Comparison of spectral clustering algorithm. University of Washington, Technical report

International Journal of Enterprise Computing and Business Systems (Online)

http://www.ijecbs.com

Vol. 1 Issue 1 January 2011