Engineering Sciences International Research Journal
ISSN (Print) : 2330 – 4338 ; Volume 1 Issue 1 ǁ February. 2013 ǁ

A Study of Partitioning Clustering Algorithms for Spatial Data Mining

N. Santhosh Kumar[1], K Nageswara Rao[2], A. Govardhan[3], V. Sitha Ramulu[4]

Abstract: Spatial data mining is the extraction of implicit knowledge, spatial relation and the discovery of interesting characteristics and patterns that are not explicitly represented in the databases. Spatial data mining has wide range of applications in many fields, including GIS system, image data base exploration, medical imaging etc., Clustering is an important concept in spatial data mining. Clustering methods can be divided into two types-partitioning and hierarchical approaches. The partitioning approach aims to divide the data set into several clusters, which may not overlap with each other but together cover the entire data space. A data item is assigned to the "closest" cluster based on the proximity or dissimilarity measure. Hierarchical clustering approaches decompose the data set with a sequence of the nested partitions, from fine to coarse resolution. In this paper we present an overview of partitional clustering algorithms for spatial data mining. Partitional clustering algorithms are divided in to three types: distance- based, model- based and density- based algorithms. Distance based partitioning clustering rely on the distance or dissimilarity measure and an optimization criterion to group those most similar objects in to clusters. Model-based partitioning clustering methods assume that the data of each cluster conform to a specific statistical distribution (e.g., Gaussian distribution) and the whole dataset is a mixture of several distribution models. Density-based approaches regard the cluster as a dense region (relative to sparse regions) of data objects.

Keywords: CLARANS, Clustering, DBSCAN, K- means, Spatial data mining.

1. INTRODUCTION

Spatial data mining[1] is the application of data mining techniques to the spatial data. Because of the huge amounts (usually, terabytes) of spatial data that may be obtained from the satellite images, medical equipments, video cameras, etc. It is costly and often unrealistic for the users to examine spatial data in detail. Spatial data mining aims to automate such a knowledge discovery process.

Clustering is an important concept in the spatial data mining. Cluster analysis[2] divides data into meaningful or useful groups (clusters). Cluster analysis has long been used in a wide variety of fields- such as psychology and other social sciences, biology, statistics, pattern recognition, information retrieval, machine learning, and data mining.

Clustering methods can be divided into two types-partitioning and hierarchical approaches . The partitioning approach aims to divide the data set into several clusters, which may not overlap with each other but together cover the entire data space. A data item is assigned to the "closest" cluster based on the proximity or dissimilarity measure. Hierarchical clustering approaches decompose the data set with a sequence of the nested partitions, from fine to coarse resolution. In this paper we present an overview of partitional clustering algorithms for spatial data mining.

2. DISTANCE- BASED PARTITIONING CLUSTERING

Distance based partitioning clustering rely on the distance or dissimilarity measure and an optimization criterion to group those most similar objects in to clusters. In this paper we give an overview of the two distance based partitioning clustering algorithms- CLARANS and k- means.

A. CLARANS Algorithm

CLARANS[3] algorithm mix both PAM(Partitioning Around Medoids) and CLARA(Clustering LARge Applications) by searching only the subset of the dataset and it does not confine itself to any sample at any given time. One key difference between the CLARANS and PAM [3] is that the former only checks a sample of the neighbors of a node. But, unlike the CLARA, each sample is drawn dynamically in the sense that no nodes corresponding to particular objects are eliminated outright. In other words, while the CLARA draws a sample of nodes at the beginning of a search, CLARANS draws a sample of neighbors in each step of a search. This has the benefit of not confining a search to a localized area.

Algorithm CLARANS

1. Input parameters numlocal and maxneighbor. Initialize i to 1, and mincost to a large number.

2. Set current to an arbitrary node in Gn;k .
3. Set j to 1.
4. Consider a random neighbor S of current, and based on 5, calculate the cost differential of the two nodes.

5. If S has a lower cost, set current to S, and go to Step 3.

6. Otherwise, increment j by 1. If j maxneighbor, go to Step 4.

7. Otherwise, when j > maxneighbor, compare the cost of current with mincost. If the former is less than mincost, set mincost to the cost of current and set bestnode to current.

8. Increment i by 1. If i > numlocal, output bestnode and halt. Otherwise, go to Step 2.

Steps 3 to 6 above search for nodes with progressively lower costs. But, if the current node has already been compared with the maximum number of the neighbors of the node (specified by maxneighbor) and is still of the lowest cost, the current node is declared to be a "local" minimum. Then, in Step 7, the cost of this local minimum is compared with the lowest cost obtained so far. The lower of the two costs above is stored in mincost. Algorithm CLARANS then repeats to search for other local minima, until numlocal of them have been found.

B. K- means Algorithm

The k-means algorithm [4] is a fast method to perform clustering. The algorithm consists of a simple re-estimation procedure as shown below .

Algorithm K- means

Input: a set of *n* data points, and the number of clusters (*K*)

Output: centroids of the *K* clusters

1. Initialize the *K* cluster centers

2. Repeat Assign each data point to its nearest cluster center

3. Re compute the cluster centers using the current cluster memberships

4. Until there is no further change in the assignment of the data points to new cluster centers The original *n* data points that to be clustered are contained in the dataset $X = \{x1, ..., xn\}$. The k-means algorithm partitions *n* data points into *K* different sets. The assignment of a data point *xi* to its nearest cluster center *cj* (step 2) is decided on the basis of the function called membership function, $m(cj|xi)$. The function returns either one of the {0,1} values: $m(cj|xi) = 1$ if $j = argmin_k \| xi - ck \|^2$; it is zero, otherwise. In step 3, the new centroids of clusters can be computed from all data points *xi* in the cluster. The objective function *J* of the algorithm is to minimize the sum of error squared, $J = \sum_{i=1:n} min_{j\{1..k\}} \| xi - ck \|^2$.

3. MODEL BASED PARTITIONING CLUSTERING

Model-based partitioning clustering methods assume the data of each cluster conform to a specific statistical distribution (e.g., Gaussian distribution) and the whole dataset is a mixture of several distribution models. In this paper we give an overview of the model based partitioning clustering algorithms- expectation-maximization (EM).

A. Expectation Maximization (EM)

Expectation maximization (EM)[5] is a clustering algorithm that works based on the partitioning methods. The EM is a memory efficient and also easy to implement algorithm, with a profound probabilistic background. This method estimates the missing parameters of probabilistic models. Generally, this is an optimization approach, which had given some initial approximation of the cluster parameters, iteratively performs the two steps: first, the expectation step computes the values expected for the cluster probabilities, and second, the maximization step computes the distribution parameters and their likelihood given the data. It iterates until the parameters being optimized reach a fix- point or until the log-likelihood function, which measures the quality of clustering, reaches its maximum. The algorithm is similar to the K-means procedure in that a set of parameters are re-computed until a desired convergence value has been achieved. The finite mixtures model assumes that all attributes to be independent random variables.

A mixture is a set of *N* probability distributions where each of the distribution represents a cluster. An individual instance is assigned a probability that it would have a certain set of attribute values given it was the member of a specific cluster. In the simplest case *N=2*, the probability distributes are assumed to be normal and the data instances consist of a single real-valued attribute. Using the scenario, the job of the algorithm is to determine the values of five parameters, specifically:

1. The mean and standard deviation for cluster 1

2. The mean and standard deviation for cluster 2

3. The sampling probability *P* for cluster 1 (the probability for cluster 2 is *1-P*)

The procedure for expectation maximization is as follows:

ISBN: 978-93-81583-90-6

1. Guess initial values for the five parameters.

2. Use the probability density function for a normal distribution to compute the cluster probability for each instance. In the case of a single independent variable with mean μ and standard deviation σ, the formula is:

$$f(x) = \frac{1}{(\sqrt{2\pi}\sigma)e^{\frac{-(x-\mu)^2}{2\sigma^2}}}$$

3. Use the probability scores to re-estimate the five parameters.

4. Return to Step 2.

4. DENSITY BASED PARTITIONING CLUSTERING

Density-based approaches regard a cluster as a dense region, that is relative to sparse regions, of data objects. Density-based clustering can adopt two different strategies:

A. Grid based approach

A grid-based approach divides the data space into a number of finite set of multi-dimensional grid cells, calculates the density of each grid cell, and then groups those neighboring dense cells into a cluster.

CLIQUE Algorithm: CLIQUE, named for Clustering In QUEst, the data mining research project at IBM Almaden, [6] is a density and grid-based approach for high dimensional data sets that provides "automatic sub-space clustering of high dimensional data." The algorithm CLIQUE has three phases:

- Identification of subspaces that contain clusters
- Identification of clusters
- Generation of minimal description for the clusters

The first phase of the algorithm involves a bottom-up algorithm to find dense units. It first makes a pass over the data to identify 1-dimensional dense units. Dense units in parents are determined based on the information available in the children. The second step of the algorithm is a matter of finding the connected components in a graph using the dense units as vertices, and having an edge present if and only if two dense units share a common face. A depth-first search algorithm is used to find the connected components. The identification of clusters is dependent on the number of dense units, *n,* and these have been previously limited by the threshold parameter. The final step takes the connected components identified in step 2 and generates a concise description of the cluster.

B. Neighborhood based approach

The key idea of neighborhood-based approaches is that, given a radius e or a side length w, the neighborhood (either a hyper-sphere of radius e or a hyper-cube of side length w) of the object has to contain at least a minimum number of objects (MinFts) to form a cluster around this object.

DBSCAN algorithm: DBSCAN (Density Based Spatial Clustering of Applications with Noise) algorithm[7] is based on center-based approach. In the center-based approach, density is estimated for a particular point in the dataset by counting the number of points within a specified radius, *Eps*, of that point. This includes the point itself. The center-based approach to density allows us to classify a point as a core point, a border point, a noise or background point. A point is core point if the number of points within *Eps*, a user-specified parameter, exceeds a certain threshold, *MinPts*, which is also a user- specified parameter.

The DBSCAN algorithm is given as follows:

1. Label all points as core, border, or noise points.

2. Eliminate noise points.

3. Put an edge between all core points that are within *Eps* of each other.

4. Make each group of connected core points into a separate cluster.

5. Assign each border point to one of the clusters with its associated core points.

Any two core points that are close enough within a distance *Eps* of one another are put in the same cluster. It is also applicable for any border point which is close enough to a core point is put in the same cluster as the core point. Noise points are disposed. The basic approach of how to determine the parameters *Eps* and *MinPts* is to look at the behavior of the distance from a point to its **k**th nearest neighbor, which is called **k**-dist. The **k**-dists are computed for all the data points for some k.

5. CONCLUSION

Clustering is an important concept in spatial data mining. Clustering methods can be divided into two types-partitioning and hierarchical approaches . The partitioning approach aims to divide the data set into several clusters, which may not overlap with each other but together cover the entire data

space. Partitioning clustering is divided in to 3 types. The first type is distance- based methods. CLARANS and k- means algorithms are distance based methods and they methods concentrate on how well points fit into their clusters and tend to build clusters of proper convex shapes.The second type of partitioning clustering is density- based algorithms. They try to discover dense connected components of data, which are flexible in terms of their shape. Density-based connectivity is used in the algorithms like DBSCAN.

The third type of partitioning clustering is model- based algorithms. Model-based partitioning clustering methods assume the data of each cluster conform to a specific statistical distribution (e.g., Gaussian distribution) and the whole dataset is a mixture of several distribution models. An example is Expectation maximization (EM). This method estimates missing parameters of probabilistic models. Generally, this is an optimization approach, which had given some initial approximation of the cluster parameters, iteratively performs two steps: first, the expectation step computes the values expected for the cluster probabilities, and second, the maximization step computes the distribution parameters and their likelihood given the data.

6. REFERENCES

[1] W. Lu, J. Han, and B. C. Ooi. Discovery of General Knowledge in Large Spatial Databases. In Proc. Far East Workshop on Geographic Information Systems pp. 275-289, Singapore, June 1993.

[2] Richard C. Dubes and Anil K. Jain, (1988), Algorithms for Clustering Data, Prentice Hall.

[3] Krzysztof Koperski.; Junas Adhikary.; and Jiawei Han. Spatial Data Mining: Progress and Challenges Survey Paper, School of Computer Science Simon Fraser University Burnaby, B.C.Canada V5A IS6.

[4] Hamerly, G., Elkan, C.: Alternatives to the k-means algorithm that find better clusterings. In Proceedings of 11th ACM CIKM International Conference on Information and Knowledge Management (2002) 600-607

[5] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," Journal of the Royal Statistical Society, vol. 39, pp. 1-38, 1977.

[6] AGRAWAL, R., GEHRKE, J., GUNOPULOS, D., and RAGHAVAN, P. 1998. Automatic subspace clustering of high dimensional data for data mining applications. In Proceedings of the ACM SIGMOD Conference, 94-105, Seattle, WA.

[7] Borah, B., Bhattacharyya, D. K. “An Improved Sampling-Based DBSCAN for Large Spatial Databases,” In: Proceedings of International Conference on Intelligent Sensing and Information, pp. 92-96, 2004.

* * *

[1]Research Scholar, Dept. of CSE, JNTU- Hyderabad, India.
[2]Professor & Head, Computer Science & Engineering,PVPSIT – Vijayawada, A.P., India.
[3]Professor in CSE & Director of Evaluation, JNTU Hyderabad, A.P., India.
[4]Research Scholar, Dept. of CSE, Acharya Nagarjuna University, A.P., India.