Volume 3, No. 8, August 2012

Journal of Global Research in Computer Science

ISSN-2229-371X

RESEARCH PAPER

Available Online at www.jgrcs.info

IMPROVEMENT OF CLUSTERING ON HIGH DIMENSIONAL DATASETS USING PCA

Dr.Dharmender Kumar[1], Parveen Kumar[2]

Guru Jambheshwar University of Science and Technology, Hisar
dharmindia24@gmail.com,
sisodia.parveen@gmail.com

Abstract: The last decade has seen an explosive growth in the generation and collection of data. In the field of data mining there are various techniques are used to extract useful information from the data set. There is various estimation techniques are used in clustering methods out of these Euclidean distance and density is used for estimation. Out of these estimation techniques one another technique mass is also used. Mass based clustering gives better performance when apply with PCA technique in multidimensional and high dimensional data set. On the basis of run time behaviour the DEMassDBSCAN algorithm is better than DBSCAN clustering algorithm is discussed.

Keywords: KDD, PCA, CLI.

INTRODUCTION

In the emerging field of knowledge discovery in databases (KDD) and data mining there are various techniques are use to find useful information from large amount of data. The term "KDD" is an overall process of discovering useful knowledge from data. A new generation of computational techniques and tools is required to support the extraction of useful knowledge from the rapidly growing volumes of data [1]. In KDD there are various steps are use for extraction useful knowledge of large amount of data are selection, preprocessing, transformation, data mining and evaluation. Data mining is a step in KDD [2]. In the data mining there are various techniques are used out of which clustering is mostly used. In clustering there are methods like partitioning, density based, hierarchical, grid and model based clustering. These methods are used in clustering but a new technique is introduced here is mass based clustering.

CLUSTERING

In Field of data mining there are various techniques are used like association, information retrieval, classification, clustering, regression, prediction and frequent pattern mining. Clustering is an important unsupervised learning technique. In partitioning method the popular algorithm k-means is used and in the density based algorithm the DBSCAN algorithm is used [3]. In clustering there are various clusters are formed based on the data sets i.e. distance, density and mass. Now a days in clustering mass estimation technique is used it is alternate of density based clustering. The distance estimation is applied on partitioning method of clustering the algorithm which is used to implement is k-means clustering algorithm. The DBSCAN clustering algorithm is applied on arbitrary shaped spatial data with noise. In the DBSCAN algorithm the points are grouped are dense and forms cluster. In DBSCAN algorithm is based on various parameters like core points, min pts, border points and noise. The points are not belonging to cluster are called noise [4]. In the DEMassDBSCAN algorithm there are also core regions and noise points are used as a parameter.

RELATED WORK

Mass based clustering:

In the clustering approach there is various estimation techniques are used to form clusters. The mass estimation is another technique to find clusters in arbitrary shape data. In the clustering the mass estimation is unique because in this estimation there is no use of distance or density [6]. In the DBSCAN clustering algorithm hyper spheres are used to show points but in DEMassDBSCAN clustering algorithm rectangular regions are used for showing the points [7].
What is Mass?

Number of points in a given region is called mass of that region. Each region in a given data space having rectangular shape and for estimation of mass a function that called rectangular function [5].

One- Dimensional Mass Estimation: The estimation of mass in data space is depends on levels (h). If value of level h=1 the shape of function is concave. For multidimensional mass estimation the value of h is >1.

$$mass\,(x,h)\begin{cases}\sum_{i=1}^{n-1} mass_i(x,h-1)p(s_i), h>1\\ \sum_{i=1}^{n-1} m_i(x)p(s_i), \qquad h=1\end{cases}$$

Eq-1 [6]

In the one dimensional mass estimation the value of p (s_i) is equals to

$$p(s_i)=\frac{x_{i+1}-x_i}{x_n-x_i} \qquad [5]$$

For one-dimensional mass estimation the value of h is equals to 1.

$$\sum_{i=1}^{n-1} m_i(x)p(s_i), \qquad h=1$$

In the above equation the mi(x) mass base function.

$$\overline{mass}(x,h) = \frac{1}{t}\sum_{k=1}^{t} mass(x,h/D_k) \quad \text{Eq-2}$$

Combining these two equations-

$$\sum_{i=1}^{n-1} m_i(x)p(s_i) \approx \frac{1}{t}\sum_{i=1}^{t} m(T_i(x))$$

$$mass(x,h) \approx \frac{1}{t}\sum_{i=1}^{t} m(T_i^h(x)$$

$$\overline{mass}(x,h) \approx \frac{1}{t}\sum_{i=1}^{t} m(T_i^h(x/Di)) \quad [6]$$

For large data bases the values of t is 1000 and Ψ 256. For small data sets the value of h and Ψ is varies.

Multi-dimensional mass estimation:

In multidimensional mass estimation there is value of h >1. In multidimensional mass estimation there are two functions mass and random spaces generator in data space.

$$\overline{gmass}(x) \approx \frac{1}{t}\sum_{i=1}^{t} m(T_i^h(X/D_i)) \quad [6]$$

Mass Based Clustering:

$$h:d-Tree(x) = m_j = m(T^h(x/D)) \quad [6]$$

Multidimensional mass estimation using h: d Trees:

$$\overline{gmass}(x) \approx \frac{1}{t}\sum_{i=1}^{t} h:d-(x)Tree_i(X) \quad [6]$$

The multi dimensional data set the value of h (level) is greater than 1 and is is similar to one dimensional mass estimation. In this equation the value of *x* is replaced by X.

MassTER Algorithm: In the masTER algorithm the mass will estimate using h:d trees. In the h:d trees the h stands for no of times attribute appears and d stands for no. of dimensions. The height of tree is calculated as l=h×d .

```
Algorithm
Step 1 Build trees $t_i$.
Step 2 Assign a cluster seed for every instance.
Step 3 Join two seed based clusters.
Step 4 Cluster having instances less than ƞ are called noise instances.
Step 5: return clusters $c_j$,j=1,2,3,-------k. And E.(noise instances)

Data input:
D= Input Data.
T= No of trees.
Ψ= Sampling size.
h= how many times attribute come in a path.
ƞ= minimum instances in a cluster.
```

In this algorithm firstly h:d trees are made of instance in a region. The cluster seed is assigned for each region in data space. The cluster seed are joined and make pair with neighbour cluster seed. The instances having value less than ƞ are defined noise [6].

Proposed Work: in the proposed work the principle component analysis is applied on the pre-processed data sets then clustering is applied on numeric data sets.

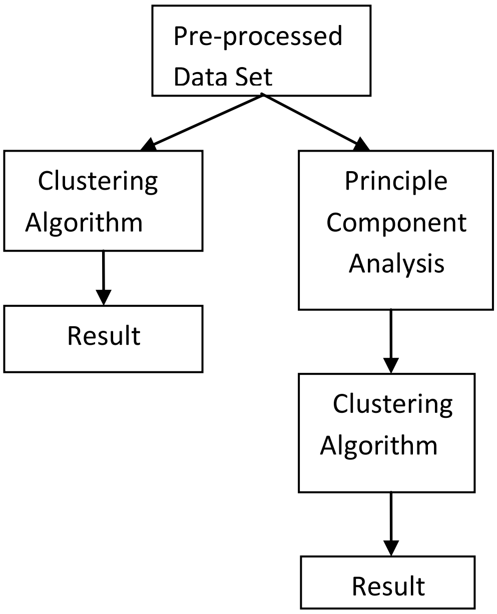

Figure: 1

PCA (principle component analysis):

PCA is a powerful tool for analyzing data and patterns are identified in data. For high dimensional data sets the reduction in the dimensions without loses of information. In the PCA first of all take data set and then subtract the means from each of dimensions. In next step covariance matrix is computed from mean subtracted data. For information about data the eigenvectors and eigen values are calculated from covariance matrix. The eigenvector having high eigen value is called principle component of the dataset. In the final step we will take transpose of vector and multiply from original data set [10]. In the generalized PCA there is constructive algebra-geometric solution of subspace if no. of subspaces is known. In the generalized PCA algebraic and geometric interpretations are done in probabilistic manner [11].

TOOL

In the implementation of data mining algorithms mostly weka toolkit is used. Weka is developed by University of Waikato in New Zealand weka . Weka is popular open source machine learning software and it provides Graphical user interface. Weka is mainly consists of four interfaces like Explorar, Experimenter, Knowledge flow and Simple CLI [8]. In the implementation of any algorithm .arff file format data is used. Weka's main user interface is the explorer, but essentially the same functionality can be accessed through the component based Knowledge Flow interface and from the command line. By using weka we can implement all standard algorithms [9].

EXPERIMENTAL RESULT

In this paper there is K-means and DBSCAN clustering algorithm is used for benchmark. The clustering result is analyzed in terms of CPU run time (in nano seconds), clustered instances, unclustered instances. In the implementation of masTER the two data sets are used. The experimental result will described in following subsections. In the DEMassDBSCAN algorithm the values of ψ and t is taken 256 and 1000 by default. In the implementation of DEMassDBSCAN and DBSCAN algorithm there are only one parameter is used at one time h for DEMassDBSCAN and ε for DBSCAN. For implementation of DEMassDBSCAN and DBSCAN algorithm we take following data sets that are shown below:

Pollution Data set: The Iris data set contains 60 instances and 16 attributes. In the clustering result the run time is better than DBSCAN algorithm that is shown in table 1.

Table 1

Pollution Data set		
	DEMassDBSCAN	DBSCAN
Run Time	0.03	0.05
Cluster	1	1
Unassigned	3	6

Segment Data set: The Segment Data set contains2310 instances and 20attributes. In the clustering result the run time is better than DBSCAN algorithm that is shown in table2 .

Table 2

Segment Data set		
	DEMassDBSCAN	DBSCAN
Run Time	0.48	7
Cluster	3	7
Unassigned	1	1

Waveform 5000 Data set: The Pollution Data set contains 5000 instances and 41 attributes. In the clustering result the run time is better than DBSCAN algorithm that is shown in table 3.

Table3.

Waveform 5000 Data Set		
	DEMassDBSCAN	DBSCAN
Run Time	0.01	61.08
Cluster	13	3
Unassigned	46	53

Mfeat-factors: The Diabetes data set contains 2000 instances and 217 attributes. In the clustering result the runtime is better than DBSCAN algorithm that is shown in table 4.

Table 4

Mfeat-factorsData set		
	DEMassDBSCAN	DBSCAN
Run Time	0.6	8.57
Cluster	1	10
Unassigned	36	19

On the basis of run time behaviour the DEMass DBSCAN clustering algorithm is better than DBSCAN clustering algorithm. It gives better result when applied on large datasets. The result is shown in Table 5.

Table 5

Run time behaviour				
	Pollution	Segment	Waveform 5000	Mfeat-factors
DEMassDB SCAN	0.03	0.48	0.01	0.6
DEMass DBSCAN without PCA	0.03	0.67	2.8	6.69
DBSCAN	0.05	7	61.8	8.57
DBSCAN without PCA	0.05	8.78	74.85	60.22

The comparison between DEMassDBSCAN And DBSCAN Clustering algorithm is based on their run time is shown below.

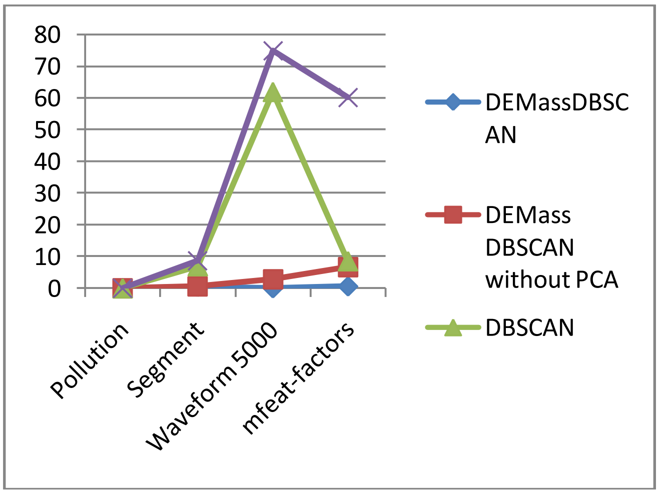

Figure: 2

CONCLUSION

In this paper the runtime behaviour of the DEMassDBSCAN algorithm is better than DBSCAN clustering algorithm. It will give better results on the large data sets. The un-assigned clusters are less than the DBSAN clustering algorithm. The result of the DEMassDBSCAN is efficient and having less noise points out of whole data set. In future mass estimation is applied on other tasks.

REFERENCES

[1]. Usama Fayyad, Gregory Piatetsky-Shapiro and padharaic Smyth, "The KDD Process for Extracting Useful Knowledge from Volumes of Data" ACM Computing Surveys, Vol. 39, 1996.

[2]. Qi Luo "Knowledge Discovery and Data Mining" Work shop on Knowledge Discovery and Data Mining , 2008 pp.3-5.

[3]. Pang NingTan, Michael Steinbach, Vipin Kumar, "Introduction to Data mining" Addison Wesley, 2004.

[4]. Martin Ester, Hans-Peter Kriegel,Torg Sander,Xiaowei Xu KDD “A Density Based algorithm for discovering clusters in large spatial data bases with noise”, KDD Proceedings.1998.

[5]. J.Han and M. Kamber. “Data mining: Concepts and techniques” Second edition, Elsevier, 2006.

[6]. K.M Ting, G-T Zhou, F. T Liu and J.S.C Tan, “Mass Estimation and its Applications”, Proceedings of KDD,2010

[7]. K.M. Ting, Jonathan R Wells, **“**Multi-Dimensional Mass Estimation and Mass-based Clustering”, IEEE Proceedings 2010.

[8]. Kai Ming Ting, Takashi Washioy, Jonathan R. Wells and Fei Tony Liu, **“**Density Estimation based on Mass,” international conference on data mining, 2011.

[9]. Z. Markov, and I.Russell “An introduction to WEKA data mining system” tutorial. http://www.cs.ccsu.edu

[10]. Sree Ram Nimmagadda, Phaneendra Kanakamedala and Vijay Bashkarreddy Yaramala, “Implementation of Clustering Through Machine Learning Tool” IJCSI Vol. 8, 2011

[11]. Lindsay I Smith, A tutorial on Principal Components Analysis, 2002.

[12]. Rene´ Vidal, Member, IEEE, Yi Ma, Member, IEEE, and Shankar Sastry, Fellow, IEEE “Generalized Principal Component Analysis (GPCA)” VOL. 27, NO. 12, DECEMBER 2005.