Density-Based Clustering Based on Probability Distribution for Uncertain Data

Pramod Patil, Ashish Patel, Parag Kulkarni

Abstract: ***Today we have seen so much digital uncertain data produced. Handling of this uncertain data is very difficult. Commonly, the distance between these uncertain object descriptions are expressed by one numerical distance value. Clustering on uncertain data is one of the essential and challenging tasks in mining uncertain data. The previous methods extend partitioning clustering methods like k-means and density-based clustering methods like DBSCAN on uncertain data based on geometric distances between objects. Such method facing the problems with the data that they cannot handle uncertain objects that are geometrically indistinguishable (such as weather data across the world at same time). In this paper, we model uncertain objects in both continuous and discrete domains with the help of probability distribution. We use Kullback-Leibler divergence to measure similarity between uncertain objects in both the continuous and discrete Values, and integrate that into partitioning and density-based clustering methods to cluster uncertain objects. We first find out uncertain objects and then we cluster uncertain data according to partitioning based clustering. Then remaining data we clustered by using any traditional method of clustering.***

Index Terms—Clustering, Uncertain Data, Probabilistic Mass Function, Probabilistic Density Estimation, Fast Gaussian Transform.

I. INTRODUCTION

Data is mostly associated with uncertain values because of measurement inaccuracy, sampling discrepancy, outdated data sources, or other errors. This is true for applications that require interaction with the physical world, such as sensor monitoring and location-based services. For example, in the scenario of moving objects (such as vehicles or people), it is impossible for the database to track the exact locations of all objects at all-time instants. Therefore, the location of each object is associated with uncertainty between updates. These various sources of uncertainty have to be considered in order to produce accurate query and mining results. Clustering uncertain data has been well recognized as an important issue. Uncertain data object can be represented by a probability distribution. The problem of clustering uncertain objects according to their probability distribution happen in many cases [1]. For Example, a weather station monitors weather conditions including various measurements like temperature, precipitation amount, humidity, wind speed, and direction [1]. The daily weather record varies from day to day, which can be modeled as an uncertain object represented by a distribution over the space formed by several measurements. Can we group the weather conditions during the last month for stations. Essentially, we need to cluster the uncertain objects according to their distributions [1].

Manuscript Received on June 2014.

Prof. Pramod Patil, Research Scholar, University of Pune, India.
Mr. Ashish Patel, Student, PDDYPIET, University of Pune, India.
Dr. ParagKulkarni, College of Engineering, Pune, India.

II. LITERATURE SURVEY

The previous studies on clustering uncertain data are largely various extensions of the traditional clustering algorithms designed for certain data. As an object in a certain data set is a single point, the distribution regarding the object itself is not considered in traditional clustering algorithms.Thus, the studies that extended traditional algorithms to cluster uncertain data are limited to using geometric distance-based similarity measures, and cannot capture the difference between uncertain objects with different distributions. Specifically, two principal categories exist in literature, namely partitioning clustering approaches [4], [8] and density-based clustering approaches [2], [5]. As these approaches only explore the geometric properties of data objects and focus on instances of uncertain objects, they do not consider the similarity between uncertain objects in terms of distributions. Let us examine this problem in the existing categories of approaches in detail. Suppose we have two sets A and B of uncertain objects. The objects in A follow uniform distribution, and those in B follow Gaussian distribution. Suppose all objects in both sets have the same mean value (i.e., the same center). Consequently, their geometric locations (i.e., areas that they occupied) heavily overlap. Clearly, the two sets of objects form two clusters due to their different distributions. Partitioning clustering approaches: Extend the k-means method with the use of the expected distance to measure the similarity between two uncertain objects. The expected distance between an object P and a cluster center c (which is a certain point) is$ED(P.c) = \int_P f_P(x)dist(x.c)dx$ where f_p denotes the probability density function of P and the distance measure dist is the square of Euclidean distance. In it is proved that ED(P.c) is equal to the dist between the center (i.e., the mean) P.c of P and c plus the variance of P. That is, ED(P.c)= dist(P.c.c) + Var (P). Accordingly, P can be assigned to the cluster center $argmin_c\{ED(P.c)\} = argmin_c\{dist(P.c.c)\}$ Thus, only the centers of objects are taken into account in these uncertain versions of the k-means method. In our case, as every object has the same center, the expected distance-based approaches cannot distinguish the two sets of objects having different distributions[4], [8]. Density-based clustering approaches: Extend the DBSCAN method in a probabilistic way. The basic idea behind the algorithms does not change—objects in geometrically dense regions are grouped together as clusters and clusters are separated by sparse regions. However, in our case, objects heavily overlap. There are no clear sparse regions to separate objects into clusters. Therefore, the density-based approaches cannot work well [2], [5], [8].

III. OUR IDEAS AND CONTRIBUTIONS

Similarity measurement between two probability

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

distributions is not a new problem at all. In information theory, the similarity between two distributions can be measured by the Kullback-Leibler divergence (KL divergence) [3].The distribution difference cannot be captured by geometric distances. For example the two objects (each one is represented by a set of sampled points) have different geometric locations. Their probability density Functions over the entire data space are different and the difference can be captured by KL divergence, although the geometric locations of the two objects are heavily overlapping, they have different distributions (one is uniform and the other is Gaussian). The difference between their distributions can also be discovered by KL divergence, but cannot be captured by the existing methods. We consider uncertain objects as random variables with certain distributions. We consider both the discrete case and the continuous case. Directly computing KL divergence between probability distributions can be very costly or even infeasible if the distributions are complex. Although KL divergence is meaningful, a significant challenge of clustering using KL divergence is how to evaluate KL divergence efficiently on many uncertain objects. We develop a general framework of clustering uncertain objects considering the distribution as the first class citizen in both discrete and continuous cases. Uncertain objects can have any discrete or continuous distribution. We show that distribution differences cannot be captured by the previous methods based on geometric distances. We use KL divergence to measure the similarity between distributions, and demonstrate the effectiveness of KL divergence in both partitioning and density-based clustering methods.

IV. MATHEMATICAL MODEL

Uncertain Object and Probability Distributions:

We consider the dataset in two domains (i.e. discrete and continuous). If the object is discrete random variable then its probability distribution is represented by a probability mass function. The probability mass function of an uncertain objects is directly estimated by the sampling the number of observation.

The pmf of object P is,

$$P(x) = \frac{|p \epsilon P \mid p = x|}{|P|}$$

Where p□ P is an observation value of P and |.| is the cardinality of a set. Otherwise, if the domain is continuous then the object is a continuous random variable and its probability distribution is calculated probability density function For example, cameras rating is a discrete set {1,2,3,4,5} and the domain of temperature is continuous real numbers. For continuous domains, we estimate the probability density function of an uncertain object by kernel density estimation. Kernel density estimation is a nonparametric way of estimating the probability density function of a continuous random variable. Given a sample of a continuous random variable P, the kernel density estimate of the probability density function is the sum of |P|kernel functions. Each Gaussian kernel function is centered at a sample point p □ P with variance h. h is called the bandwidth, and is used to control the level of smoothing. A popular choice of the bandwidth is the Sliverman approximation rule for which $h = 1.06 * \delta|P|^{-\frac{1}{5}}$, where □ is the standard deviation of the sample points of P. The density estimator is

$$P(x) = \frac{1}{|P|\sqrt{2\pi h}} \sum_{p \epsilon P} e^{-\frac{(x-p)^2}{2h^2}}$$

KL Divergence:

KL divergence between two probability distributions is defined as follows.

Kullback-LeiblerDivergence.In the discrete case, let f and g be two probability mass function in a discrete domain ID with a finite or countably infinite number of values. The Kullback-Leibler diverge between f and g is

$$D(f \| g) = \sum_{x \, \varepsilon \, ID} \left(f(x) \log \frac{f(x)}{g(x)} \right)$$

In continuous case, let f and g be two probability density functions in a continuous domain ID with a continuous range of values. The Kullback0Leibler divergence between f and g is

$$D(f \| g) = \int_{ID} \left(f(x) \log \frac{f(x)}{g(x)} \right)$$

In both discrete and continuous cases, KL divergence is defined only in the case where for any x in domain ID if f(x) > 0 then g(x) > 0.

V. IMPLEMENTATION DETAILS

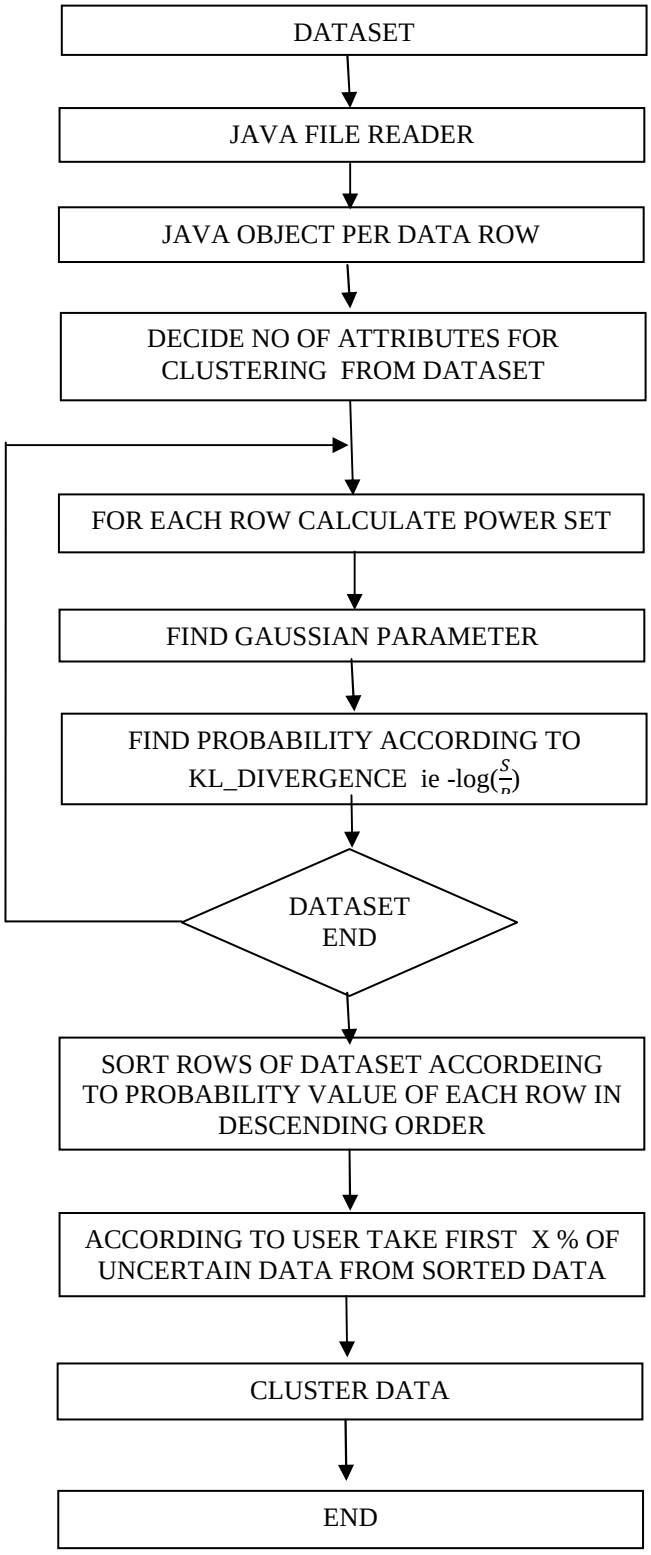
Algorithms

KL-Divergence algorithm:

Input dataset D = {$D_1, D_2, D_3, \ldots, D_N$}
Output dataset C = {$C_1, C_2, C_3, \ldots, C_N$}
Step 1 : Get the dataset
Step 2 : Read dataset into 2-D vector
Step 3 : For each row
  Step 3.1 : Calculate power set
  Step 3.2 : Calculate Count / N
    Where count = no of match
    N = total no of elements in row
Step 4 : END for
Step 5 : Set variance h (Silverman bandwidth)
Step 6: Identify sample point p
Step 7 : For each h α p
  Step 7.1 : Take variable M
  Step 7.2 : M = $e^{-\frac{(x-p)*(x-p)}{2*h*h}}$
Step 8 : END for
Step 9 : Gaussian parameter P = $\frac{M}{\sqrt{2*\pi*h}}$
Step 10 : For each row
  Step 10.1 : -log($\frac{S}{P}$)
Step 11 :END for
Step 12 : Identify elements near to infinity
Step 13 : Add into similar C_i
Step 14 : Display all vectors according to cluster C_i={$C_1, C_2, C_3, \ldots, C_N$}
Step 15 : END

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

Flowchart



Density Based Clustering Algorithm

CreateClusters (D, eps, MinPts)

C=0

For each unvisited point P in dataset D

Mark P as visited

NeighborPts = regionQuery (P, eps)

If sizeof(NeighborPts) <MinPts

Mark P as NOISE

Else

C= next cluster

ExpandCluster(P,NeighborPts,C,eps, MinPts)

Input:

```
790121,1001,7093,867,9,-10, 1,-20, 0,5011, 8,5
790122,1001,7093,867,9,-110,0,-1101,0,310, 8,2
790123,1001,7093,867,9,-503,0,-1301,0,999918,7,1
790124,1001,7093,867,9,-10, 1,-140, 1,1010, 7,2
790125,1001,7093,867,9,-110,0,-120, 0,1010, 8,1
790126,1001,7093,867,9,-110,0,-200, 0,4011, 8,2
790127,1001,7093,867,9,-90, 0,-120, 0,8011, 8,8
790128,1001,7093,867,9,-100,0,-100, 0,110, 8,4
790129,1001,7093,867,9,-110,1,-130, 1,999918,8,1
790130,1001,7093,867,9, 1, 0,-170, 0,6011, 8,8
790131,1001,7093,867,9,-60, 0,-60, 0,6110, 8,7
790201,1001,7093,867,9,-50, 0,-90, 0,211, 8,4
790202,1001,7093,867,9, 1, 0,-60, 0,16011, 8,7
790203,1001,7093,867,9,-101,0,-10, 0,999918,8,4
790204,1001,7093,867,9,-60, 0,-80, 0,2611, 8,8
790205,1001,7093,867,9,-60, 0,-701, 0,3010, 8,1
790206,1001,7093,867,9,-80, 1,-90, 1,5011, 8,6
790207,1001,7093,867,9,-90, 0,-130, 0,5011, 8,8
790208,1001,7093,867,9,-160,0,-170, 0,5010, 8,6
790209,1001,7093,867,9,-160,0,-190, 0,0, 8,0
790210,1001,7093,867,9,-150,0,-190, 0,111, 8,4
790211,1001,7093,867,9,-120,0,-160, 0,2010, 8,7
790212,1001,7093,867,9,-200,1,-1902,1,999918,8,1
790213,1001,7093,867,9,-30, 0,-200, 0,1010, 8,4
790214,1001,7093,867,9,-40, 0,-50, 0,210, 8,3
790215,1001,7093,867,9, 20, 1,-90, 0,999918,7,3
790216,1001,7093,867,9, 40, 0,-10, 0,01000, 8,1
```

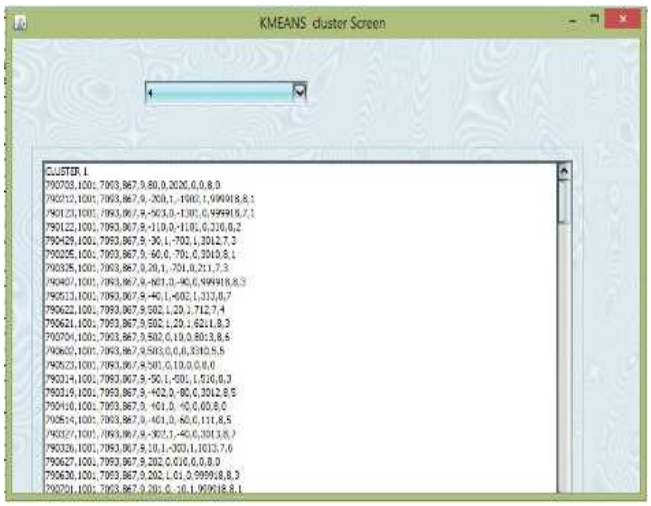
Above dataset having so many attributes out of that Date, StationId, Temperature, precipitation, pressure, wind child and sunshine are used. Output: We are getting the better performance by using probability mass function (pmf) and probability density function (pdf) for discrete random domain variables and continuous random variable respectively. We are also trying to tune the performance of algorithm. When you are using K-Means or DB-Scan algorithm directly on uncertain data to cluster the data they shows poor results. But when you are using KL-Divergence algorithm with K-Means or DB-Scan algorithm to cluster data then it shows good performance. KL-Divergence firstly cluster all infinite distance data into separate cluster. So you can avoids data loss.

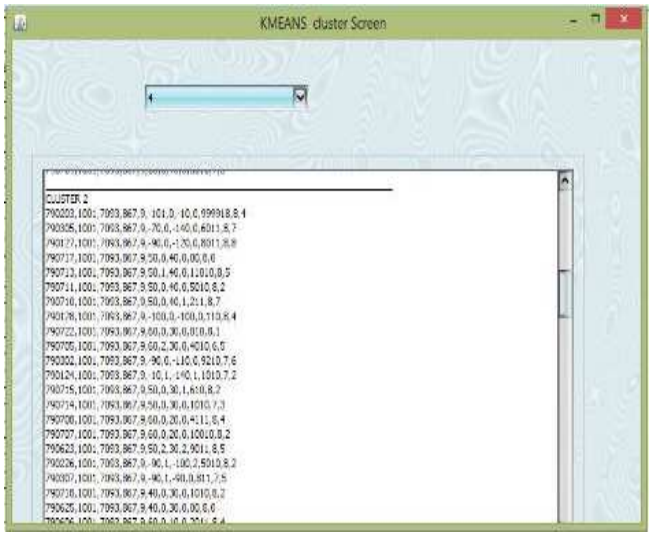
K-Means :

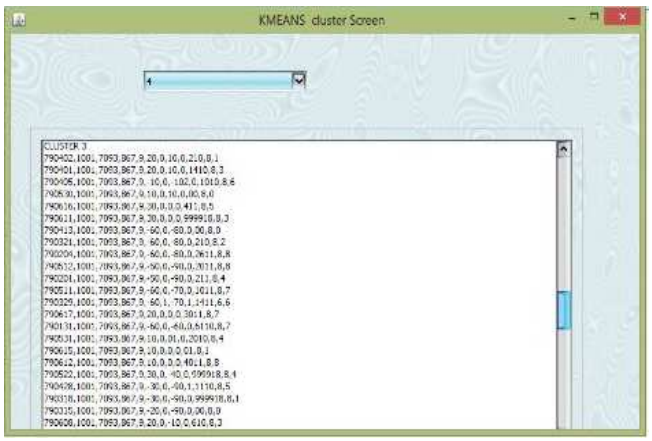
Following snapshots of K-Means algorithm. When you are applying k-means algorithm directly on uncertain data it shows poor results. We applying K-Means algorithm directly on uncertain data which snapshot is attached in input. Here we consider two attributes of dataset ie. Temperature(6th column) and wind child(12th column).But due to uncertainty in input dataset it shows poor results. In

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

following snapshot we taken user input for 4 clusters. It shows 4 clusters according to K-Means algorithm.

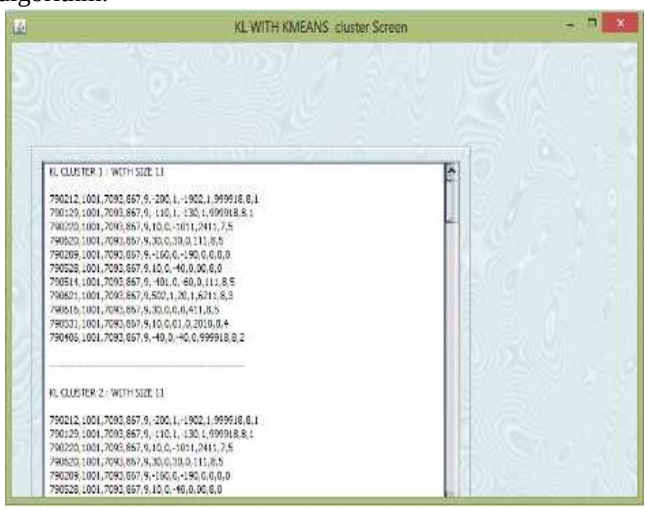


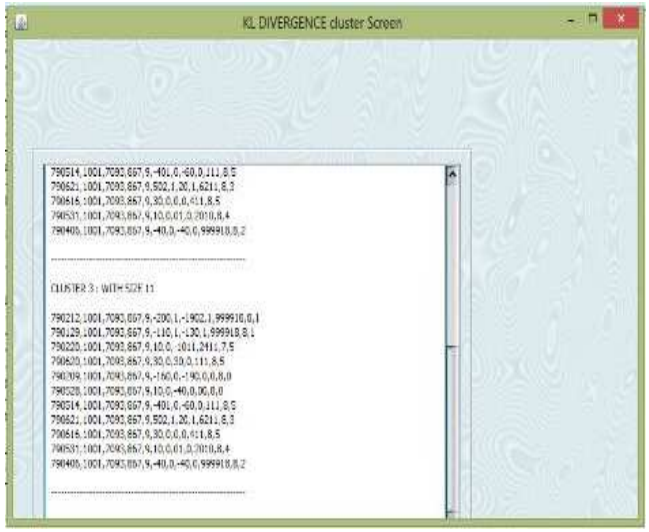


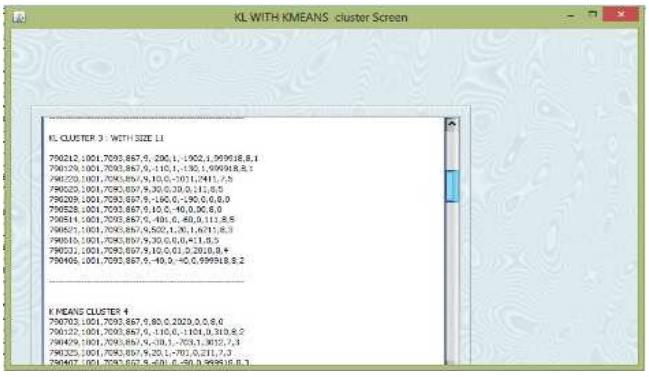


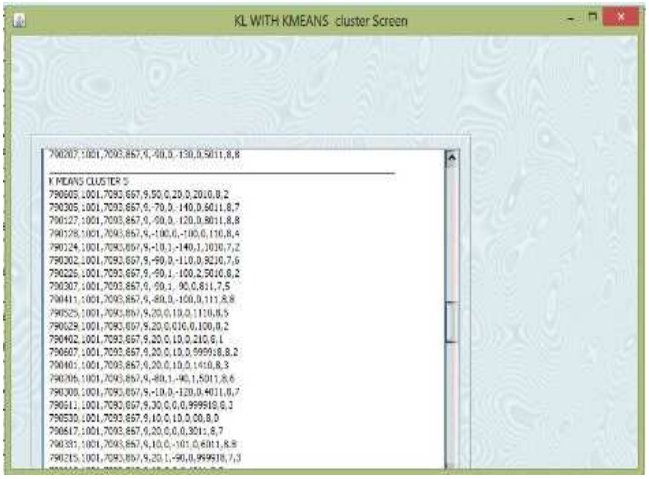
KL-Divergence with K-Means

Following snapshots of KL-Divergence with K-Means algorithm. Here we applied first KL-Devergence algorithm to cluster uncertain data which shows high uncertainty. We sort all data rows according to probabilistic similarities between all data in descending order. Here we taken first 20% data as uncertain data. That data we clustered into 3 cluster. Then after that we removed that data and weagain apply K-Means algorithm for remaining 80% data. We have made 4 clusters of that data. Due to removed uncertain data K-Means shows good performance. Also we avoid data loss. Following four snapshots are showing clusters of KL-Divergence as well as clusters of clusters of K-Means algorithm.

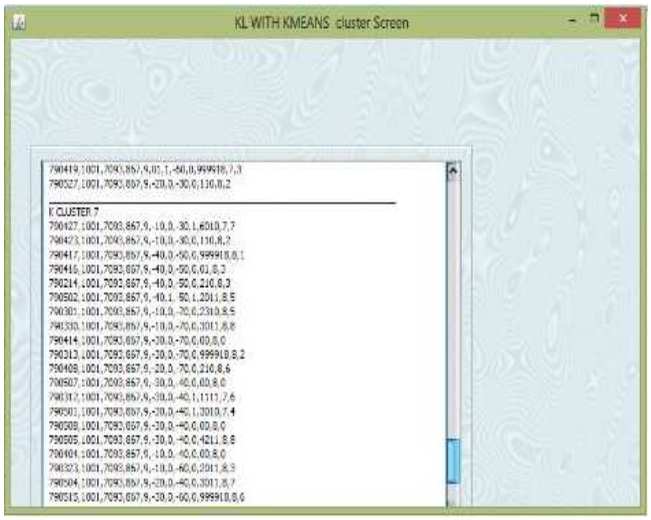








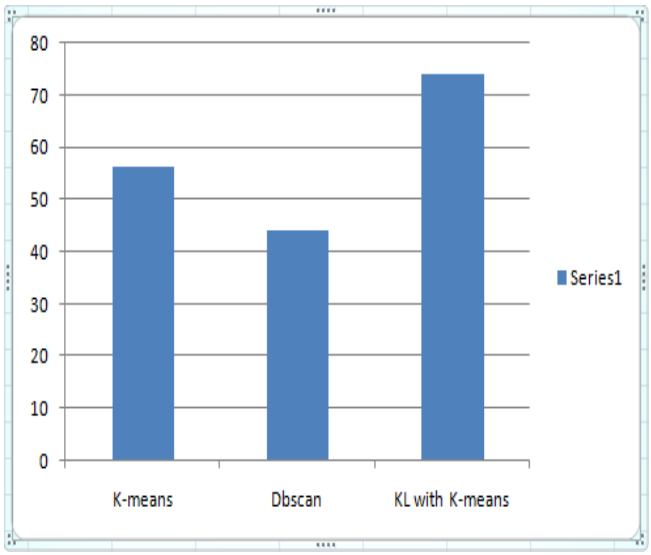
Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.



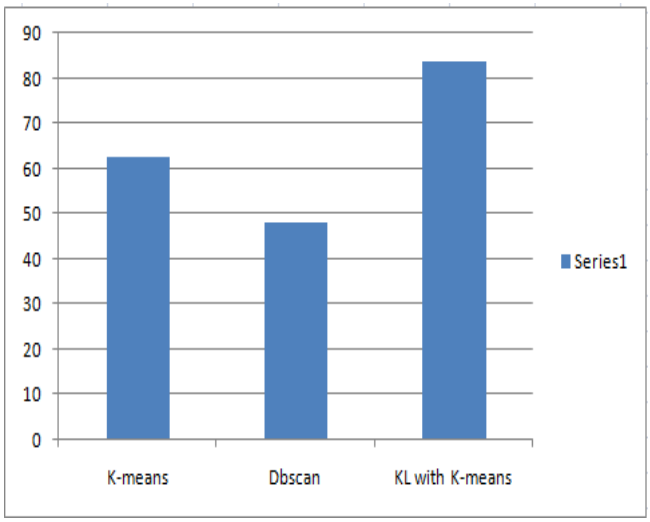
VI. ANALYSIS

In analysis graphs we are showing precision and recall with respected to K-means , DB-Scan and KL with K-means algorithm.

Precision Graph:



Recall Graph:



VII. CONCLUSION

We are using KL divergence as the similarity measurement for clustering uncertain data based on the similarity between their distributions of objects in the continuous and discrete cases. We used KL divergence into the partitioning and density- based clustering methods and got the quality results. The uncertain DBSCAN method all require evaluation of KL divergences of many attributes of the objects. As the number of uncertain objects and the sample size of each object increase, it is costly to evaluate a large amount of KL divergences expressions. The work is required to save the computation.

VIII. ACKNOWLEDGMENT

I wish to thank all the people who has directly or indirectly helped me in completing Paper work successfully.I express my gratitude towards my project guide and also towards Head of Computer Engineering Department for their valuable suggestions and constant guideline during this paper work also acknowledge the research work done by all researchers in this field across the Internet for maintaining valuable document and resource on the Internet.

REFERENCES

[1] Jiang, Jian Pei, Yufei Tao, Member and XueminLin,"Clustering Uncertain Data Based on Probability Distribution Similarity,"IEEE TRANSACTIONS ON KDE, VOL. 25, NO. 4, APRIL 2013.
[2] H. P. Kriegel and M. Pfeifle," Hierarchical Density-Based Clustering of Uncertain Data," Proc. IEEE Int'l Conf. Data Mining(ICDM).
[3] S. Kullback and R.A. Leibler, " On Information and Suffficiency," The Annals of Math Statistics.
[4] J. Han and M. Kambler Data Mining: Concept and Techniques.
[5] M. Ester, H-P. Kriegel, J. Sander and X.Xu,"A Density-Based Algorithm for Discovering Clusters in Large Spatial databases with Noise," Published in Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96).
[6] T. Imielinski and W. L. Lipski Jr.," Incomplete Information in relational Databases," J. ACM , vol. 31,pp. 761-791,1984.
[7] J.B. MacQueen," Some Methods for Classification and Analysis of Statistics and Probability, 1967.
[8] W.K.Ngai,B. Kao,C.K. Chui,R,Cheng,M,Chau and K. Y. Yip,"Efficient Clustering of Uncertain Data," Proc. Sixth Int'l Conf. ICDM,2006.
[9] J.M. Ponte and W. B. Croft,"A Language Modeling Approach to Information Retrival," Proc. 21st Ann. Nt'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR),1998.
[10] D.W. Scott, Multivariate Density Estimation: Theory, Practical and Visualization,Wiley,1992.
[11] B.W. Silverman, Density Estimation for Statistics and Data Analysis. Chapman and Hall,1986
[12] J. Xu and W.B. Croft," Cluster-Based Language Models for Distributed Retrival," Proc. 22nd Ann, Int'l ACM SIGIR,1999.
[13] C. Yang R. Duraiswami NA. Gumerov and L.S. Davis," Improved Fast Gauss Transform and Efficient Kernel Density Estimation," Proc. IEEE Int'l Conf. Computer Vision (ICCV) 2003.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.