Uncertain Data Mining: A New Research Direction

Michael Chau[1], Reynold Cheng[2], and Ben Kao[3]

1: School of Business, The University of Hong Kong, Pokfulam, Hong Kong
2: Department of Computing, Hong Kong Polytechnic University Kowloon, Hong Kong
3: Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong
Emails: mchau@business.hku.hk, csckcheng@comp.polyu.edu.hk, kao@cs.hku.hk

Abstract

Data uncertainty is often found in real-world applications due to reasons such as imprecise measurement, outdated sources, or sampling errors. Recently, much research has been published in the area of managing data uncertainty in databases. We propose that when data mining is performed on uncertain data, data uncertainty has to be considered in order to obtain high quality data mining results. We call this the "Uncertain Data Mining" problem. In this paper, we present a framework for possible research directions in this area. We also present the UK-means clustering algorithm as an example to illustrate how the traditional K-means algorithm can be modified to handle data uncertainty in data mining.

1. Introduction

Data is often associated with uncertainty because of measurement inaccuracy, sampling discrepancy, outdated data sources, or other errors. This is especially true for applications that require interaction with the physical world, such as location-based services [15] and sensor monitoring [3]. 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 [4]. These various sources of uncertainty have to be considered in order to produce accurate query and mining results.

In recent years, there has been much research on the management of uncertain data in databases, such as the representation of uncertainty in databases and querying data with uncertainty. However, little research work has addressed the issue of mining uncertain data. We note that with uncertainty, data values are no longer atomic. To apply traditional data mining techniques, uncertain data has to be *summarized* into atomic values. Taking moving-object applications as an example again, the location of an object can be summarized either by its last recorded location, or by an expected location (if the probability distribution of an object's location is taken into account). Unfortunately, discrepancy in the summarized recorded values and the actual values could seriously affect the quality of the mining results. Figure 1 illustrates this problem when a clustering algorithm is applied to moving objects with location uncertainty. Figure 1(a) shows the actual locations of a set of objects, and Figure 1(b) shows the recorded location of these objects, which are already outdated. The clusters obtained from these outdated values could be significantly different from those obtained as if the actual locations were available (Figure 1(b)). If we solely rely on the recorded values, many objects could possibly be put into wrong clusters. Even worse, each member of a cluster would change the cluster centroids, thus resulting in more errors.

Chau, M., Cheng, R., and Kao, B., "Uncertain Data Mining: A New Research Direction," in *Proceedings of the Workshop on the Sciences of the Artificial*, Hualien, Taiwan, December 7-8, 2005.

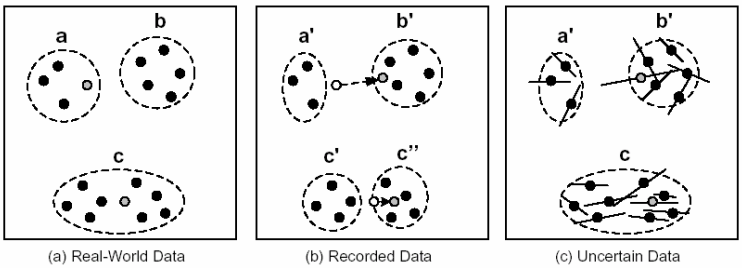

Figure 1. (a) The real-world data are partitioned into three clusters (a, b, c). (b) The recorded locations of some objects (shaded) are not the same as their true location, thus creating clusters a', b', c' and c''. Note that a' has one fewer object than a, and b' has one more object than b. Also, c is mistakenly split into c' and c''. (c) Line uncertainty is considered to produce clusters a', b' and c. The clustering result is closer to that of (a) than (b).

We suggest incorporating uncertainty information, such as the probability density functions (pdf) of uncertain data, into existing data mining methods so that the mining results could resemble closer to the results obtained as if actual data were available and used in the mining process (Figure 2(c)).

In this paper we study how uncertainty can be incorporated in data mining by using data clustering as a motivating example. We call this the *Uncertain Data Mining* problem. In this paper, we present a framework for possible research directions in this area.

The rest of the paper is structured as follows. Related work is reviewed in Section 2. In Section 3 we define the problem of clustering on data with uncertainty and present our proposed algorithm. Section 4 presents the application of our algorithm to a moving-object database. Detailed experiment results are shown in Section 5. We conclude our paper and suggest possible research directions in Section 6.

2. Research Background

In recent years, there is significant research interest in data uncertainty management. Data uncertainty can be categorized into two types, namely *existential uncertainty* and *value uncertainty*. In the first type it is uncertain whether the object or data tuple exists or not. For example, a tuple in a relational database could be associated with a probability value that indicates the confidence of its presence [1,11]. In value uncertainty, a data item is modelled as a closed region which bounds its possible values, together with a probability density function (pdf) of its value [3,4,12,15]. This model can be used to quantify the imprecision of location and sensor data in a constantly-evolving environment. Most works in this area have been devoted to "imprecise queries", which provide probabilistic guarantees over correctness of answers. For example, in [5], indexing solutions for range queries over uncertain data have been proposed. The same authors also proposed solutions for aggregate queries such as nearest-neighbor queries in [4]. Notice that all these works have applied the study of uncertain data management to simple database queries, instead of to the relatively more complicated data analysis and mining problems.

The clustering problem has been well studied in data mining research. A standard clustering process consists of five major steps: pattern representation, definition of a pattern similarity metric, clustering or grouping, data abstraction, and output assessment [10]. Only a few studies on data mining or data clustering for uncertain data have been reported. Hamdan and Govaert have addressed the problem of fitting mixture densities to uncertain data for clustering using the EM algorithm [8]. However, the model cannot be readily applied to other clustering algorithms and is rather customized for EM. Clustering on interval data also has been studied. Different distance measures, like city-block distance or Minkowski distance, have been used in measuring the similarity between two intervals [6,9]. The pdf of the interval is not taken into account in most of these metrics. Another related area of research is fuzzy clustering. Fuzzy clustering has been long studied in fuzzy logic [13]. In fuzzy clustering, a cluster is represented by a fuzzy subset of a set of objects. Each object has a "degree of belongingness" for each cluster. In other words, an object can belong to more than one cluster, each

with a different degree. The fuzzy *c*-means algorithm was one of the most widely used fuzzy clustering method [2,7]. Different fuzzy clustering methods have been applied on normal data or fuzzy data to produce fuzzy clusters [14]. While their work is based on a fuzzy data model, our work is developed based on the uncertainty model of moving objects.

3. Taxonomy of Uncertain Data Mining

In Figure 2, we propose a taxonomy to illustrate how data mining methods can be classified based on whether data imprecision is considered. There are a number of common data mining techniques, e.g., association rule mining, data classification, data clustering, that need to be modified in order to handle uncertain data. Moreover, we distinguish two types of data clustering: hard clustering and fuzzy clustering. Hard clustering aims at improving the accuracy of clustering by considering expected data values after data uncertainty is considered. On the other hand, fuzzy clustering presents the clustering result in a "fuzzy" form. An example of a fuzzy clustering result is that each data item is given a probability of being assigned to each member in a set of clusters [14].

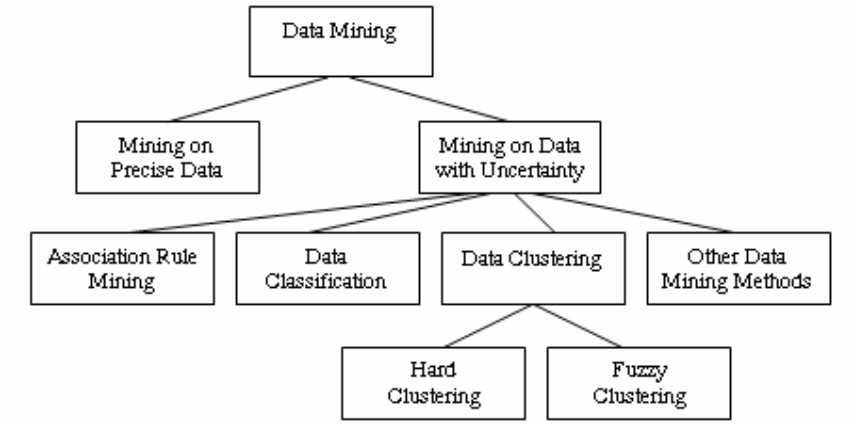

Figure 2. A taxonomy of data mining on data with uncertainty

For example, when uncertainty is considered, there is an interesting problem on how each tuple and the uncertainty associated should be represented in the dataset. Moreover, the notion of *support* and other metrics would need to be redefined. Well-known association rule mining algorithm (such as Apriori) has to be revised in order to take this into account. Similarly, in data classification and data clustering, traditional algorithms may not work any more because uncertainty was not taken into account. Important metrics, like cluster centroids, distance between two objects, or distance between an object and a centroid, have to be redefined and further studied.

4. Example on Uncertain Data Clustering

In this section, we present our work on uncertain data clustering as an example of uncertain data mining. This illustrates our idea of adapting traditional data mining algorithm for uncertain data.

4.1 Problem Definition

Let S be a set of V-dimensional vectors $\mathbf{x}_i$, where $i = 1$ to n, representing the attribute values of all the records to be considered in the clustering application. Each record o_i is associated with a probability density function (pdf), $f_i(\mathbf{x})$, which is the probability density function of o_i's attribute values $\mathbf{x}$ at time t. We do not limit how the uncertainty function evolves over time, or what the probability density function of a record is. An example pdf is the uniform density function, which depicts the worst-case or "most uncertain" scenario [3]. Another pdf commonly used is the Gaussian distribution, which can be used to describe measurement errors [12,15].

The clustering problem is to find a set C of clusters C_j, where j = 1 to K, with cluster means $\mathbf{c}_j$ based on similarity. Different clustering algorithms have different objective functions, but the general idea is to minimize the distance between objects in the same cluster while maximizing the distance between objects in different clusters. Minimization of intra-cluster distance can also be viewed as the minimization of the distance between each data point $\mathbf{x}_i$ and the cluster means $\mathbf{c}_j$ of the cluster C_j that $\mathbf{x}_i$ is assigned to. In this paper, we only consider hard clustering, i.e., every object is assigned to one and only one cluster.

4.2 K-means Clustering for Precise Data

The classical K-means clustering algorithm which aims at finding a set C of K clusters C_j with cluster mean $\mathbf{c}_j$ to minimize the sum of squared errors (SSE). The SSE is usually calculated as follows:

$$\sum_{j=1}^{K} \sum_{x_i \in C_j} \left\| \mathbf{c}_j - \mathbf{x}_i \right\|^2 \tag{1}$$

where || . || is a distance metric between a data point $\mathbf{x}_i$ and a cluster mean $\mathbf{c}_j$. For example, the Euclidean distance is defined as:

$$\left\| \mathbf{x} - \mathbf{y} \right\| = \sqrt{\sum_{i=1}^{V} \left| x_i - y_i \right|^2} \tag{2}$$

The mean (centroid) of a cluster C_i is defined by the following vector:

$$\mathbf{c}_j = \frac{1}{\left| C_j \right|} \sum_{i \in C_j} \mathbf{x}_i \tag{3}$$

The K-means algorithm is as follows:

1. Assign initial values for cluster means $\mathbf{c}_1$ to $\mathbf{c}_K$
2. **repeat**
3. **for** i = 1 to n **do**
4. Assign each data point $\mathbf{x}_i$ to cluster C_j where || $\boldsymbol{c}_j$ - $\mathbf{x}_i$ || is the minimum.
5. **end for**
6. **for** j = 1 to K **do**
7. Recalculate cluster mean $\boldsymbol{c}_j$ of cluster C_j
8. **end for**
9. **until** convergence
10. **return** C

Convergence can be defined based on different criteria. Some example convergence criteria include: (1) when the change in the sum of squared errors is smaller than a certain user-specified threshold, (2) when no objects are reassigned to a different cluster in an iteration and (3) when the number of iterations has reached a pre-defined maximum number.

4.3 K-means Clustering for Uncertain Data

In order to take into account data uncertainty in the clustering process, we propose a clustering algorithm with the goal of minimizing the *expected* sum of squared errors E(SSE). Notice that a data object $\mathbf{x}_i$ is specified by an uncertainty region with an uncertainty pdf $f(\mathbf{x}_i)$. Given a set of clusters, C_j's the expected SSE can be calculated as follow:

$$\begin{aligned} & E\left(\sum_{j=1}^{k} \sum_{i \in C_j} \left\| \mathbf{c}_j - \mathbf{x}_i \right\|^2 \right) \\ &= \sum_{j=1}^{k} \sum_{i \in C_j} E\left(\left\| \mathbf{c}_j - \mathbf{x}_i \right\|^2 \right) \\ &= \sum_{j=1}^{k} \sum_{i \in C_j} \int \left\| \mathbf{c}_j - \mathbf{x}_i \right\|^2 f(\mathbf{x}_i) d\mathbf{x}_i \end{aligned} \tag{4}$$

Cluster means are given by:

$$\mathbf{c}_j = E\left(\frac{1}{|C_j|}\sum_{i\in C_j}\mathbf{x}_i\right)$$
$$= \frac{1}{|C_j|}\sum_{i\in C_j}E(\mathbf{x}_i) \quad (5)$$
$$= \frac{1}{|C_j|}\sum_{i\in C_j}\int \mathbf{x}_i f(\mathbf{x}_i)d\mathbf{x}_i$$

We now propose a new K-means algorithm, called UK-means, for clustering uncertain data.

1. Assign initial values for cluster means $\mathbf{c}_1$ to $\mathbf{c}_K$
2. **repeat**
3. **for** i = 1 to n **do**
4. Assign each data point $\mathbf{x}_i$ to cluster C_j where E(|| $\mathbf{c}_j$ - $\mathbf{x}_i$ ||) is the minimum.
5. **end for**
6. **for** j = 1 to K **do**
7. Recalculate cluster mean $\mathbf{c}_j$ of cluster C_j
8. **end for**
9. **until** convergence
10. **return** C

The main difference between UK-mean clustering and K-means clustering lies in the computation of distance and clusters. In particular, UK-means compute the *expected* distance and cluster centroids based on the data uncertainty model. Again, convergence can be defined based on different criteria. Note that if the convergence is based on squared error, E(SSE) as in Equation (4) should be used instead of SSE.

In Step 4, it is often difficult to determine $E(\| \mathbf{c}_j - \mathbf{x}_i \|)$ algebraically. In particular, the variety of geometric shapes of uncertainty regions (e.g., line, circle) and different uncertainty pdf imply that numerical integration methods are necessary. In view of this, $E(\| \mathbf{c}_j - \mathbf{x}_i \|^2)$, which is easier to obtain, is used instead. This allows us to determine the cluster assignment (i.e., Step 4) using a simple algebraic expression.

5. A Case Study and Evaluation

5.1 Clustering Data with Line-moving Uncertainty

The UK-means algorithm presented in the last section is applicable to any uncertainty region and pdf. To demonstrate the feasibility of the approach, we describe how the proposed algorithm can be applied to uncertainty models specific to moving objects that are moving in a two-dimensional space. We also present the evaluation results of the algorithm. The algorithm was applied to a model with the *unidirectional line-moving uncertainty*, which requires that each object's location is uniformly distributed in a line segment along the line of movement in one direction.

Suppose we have a centroid c = (p, q) and a data object x specified by a line uncertainty region with a uniform distribution. Let the end points of the line segment uncertainty be (a,b) and (c,d). The line equation can be parametrized by (a + t (c - a), b + t (d - b)), where t is between [0,1]. Let the uncertainty pdf be $f(t)$. Also, let the distance of the line segment uncertainty be $D = \sqrt{(c-a)^2 + (d-b)^2}$. We have:

$$E(\| \mathbf{c} - \mathbf{x} \|^2) = \int_0^1 f(t)(D^2t^2 + Bt + C)dt \quad (6)$$

where $B = 2[(c - a)(a - p) + (d - b)(b - q)]$
$C = (p - a)^2 + (q - b)^2$

If *f(t)* is uniform, then *f(t) = 1*, and the above becomes:

$$E(\text{distance of line uncertainty from centroid}^2) = \frac{D^2}{3} + \frac{B}{2} + C \quad (7)$$

We are thus able to compute the expected squared distance easily for line-moving uncertainty for uniform distribution. These formulae can be readily used by the UK-means algorithm to decide the assignment of clusters. Nonetheless, the use of uniform distribution is only a specific example here. When the pdf's are not uniform (e.g., Gaussian), sampling techniques can be used to estimate $E(\| \mathbf{c}_j - \mathbf{x}_i \|)$.

5.2 Experiments

Experiments were conducted to evaluate the performance of UK-means. The goal is to study whether the inclusion of data uncertainty improves clustering quality. We simulate the following scenario: a system that tracks the locations of a set of moving objects has taken a snapshot of the whereabouts of the objects. This location data is stored in a set called **recorded**. Each object assumes an uncertainty model. Let **uncertainty** captures such uncertainty information. We compare two clustering approaches: (1) apply K-means to **recorded** and (2) apply UK-means to **recorded** + **uncertainty**. More specifically, we first generated a set of random data points in a 100 x 100 2D space as **recorded**. For each data point, we then randomly generated its uncertainty according to the unidirectional line-uncertainty model. The uncertainty specification (**uncertainty**) of an object contains the type of the uncertainty (bidirectional line), the maximum distance *d* that the object can move, and the direction that the object can move.

The *actual locations* of the objects were then generated based on **recorded** and **uncertainty**, simulating the scenario that the objects have moved away from their original locations as registered in **recorded**. Specifically, for each data point, we took its position in **recorded** and then generated a random number to decide the distance that the object should have moved. If it is free-moving (circle) uncertainty or bidirectional uncertainty, we generated another random number to see which direction the object should move. We use **actual** to denote the set of actual locations of the objects.

Ideally, a system should know **actual** and apply K-means on the actual locations. Although not practical, such clustering result serves as a good reference for the quality of clustering result. Hence, we compute and compare the cluster output of the following data sets:

(1) **recorded** (using classical K-means)
(2) **recorded + uncertainty** (using UK-means)
(3) **actual** (using classical K-means)

In order to verify the ability of the UK-means algorithm in generating a set of clusters close to the ones generated from **actual**, we apply a widely-used metric known as the Adjusted Rand Index (ARI), which measures the similarity between clustering results [16]. A higher ARI value indicates a higher degree of similarity between two clusters. We will compare the ARI between the sets of clusters created in (2) and (3) and the ARI between those created in (1) and (3).

Three parameters, namely number of objects (*n*), number of clusters (*K*), and the maximum distance an object can move (*d*), were varied during the experiment. Table 1 shows the different experiment results by varying *d* while keeping $n = 1000$ and $K = 20$. Under each set of different parameter settings, 500 rounds were run. In each round, the sets of **recorded**, **uncertainty**, and **actual** were first generated and the same set of data was used for the three clustering processes. The same set of initial centroids were also used in each of the three processes in order to avoid any bias resulting from the initial conditions of the K-means and UK-means algorithms. In each round, both K-means (in (1) and

(3)) and UK-means (in (2)) were allowed to run until there was no change in cluster membership for all objects in two consecutive iterations, or when the number of iterations reached 10000. The ARI values and time elapsed were averaged across the 500 runs for UK-means and K-means, respectively.

As can be seen from Table 1, the UK-means algorithm consistently showed a higher ARI than the traditional K-means algorithm applied on the recorded data. Pairwise *t*-tests were conducted and the results showed that the difference in the ARI values of the two methods was significant for all setting ($p < 0.000001$ for every case). The results demonstrated that the clusters produced by the UK-means algorithm were more similar to those clusters obtained from the real-world data. In other words, the UK-means algorithm can give a set of clusters that could be a better prediction of the clusters that would be produced if the real-world data were available.

Table 1. Experiment results

D	2.5	5	7.5	10	20	50
ARI (UK-means)	0.733	0.689	0.652	0.632	0.506	0.311
ARI (K-means)	0.700	0.626	0.573	0.523	0.351	0.121
Improvement	0.033	0.063	0.079	0.109	0.155	0.189
% of improvement	4.77%	10.03%	13.84%	20.82%	44.34%	155.75%

In terms of efficiency, we found that the UK-means algorithm requires more computational time than K-means, but it often only required a reasonable amount of extra time, which is well justified since the clustering quality is higher when uncertainty is considered.

We further conducted experiments by varying *n*, *K*, and *d* for different values, while keeping the other variables constant. In all cases, we found that the UK-means algorithm showed improvement over the traditional K-means algorithm, and the differences were all statistically significant (as shown by the *t*-test result in each case). Our preliminary results showed that the improvement of the UK-means algorithm is higher when the degree of uncertainty (as represented by *d*) increases. On the other hand, the number of objects and number of clusters do not have a large effect on the performance of the UK-means algorithm, except when the number of clusters is very small.

6. Conclusion and Future Work

Traditional data mining algorithms do not consider uncertainty inherent in a data item and can produce incorrect mining results that do not correspond to the real-world data. In this paper we propose a taxonomy of research in the area of uncertain data mining. We also present the UK-means algorithm as a case study and illustrate how the proposed algorithm can be applied. With the increasing complexity of real-world data brought by advanced sensor devices, we believe that uncertain data mining is an important and significant research area.

Acknowledgement

We would like to thank Jackey Ng (University of Hong Kong), David Cheung (University of Hong Kong), Edward Hung (Hong Kong Polytechnic University), and Kevin Yip (Yale University) for their valuable suggestions.

References

1. Barbara, D., Garcia-Molina, H. and Porter, D. "The Management of Probabilistic Data," *IEEE Transactions on Knowledge and Data Engineering*, 4(5), 1992.
2. Bezdek, J. C. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York (1981).
3. Cheng, R., Kalashnikov, D., and Prabhakar, S. "Evaluating Probabilistic Queries over Imprecise Data," *Proceedings of the ACM SIGMOD International Conference on Management of Data*, June 2003.
4. Cheng, R., Kalashnikov, D., and Prabhakar, S. "Querying Imprecise Data in Moving Object Environments," *IEEE Transactions on Knowledge and Data Engineering*, 16(9) (2004) 1112-1127.
5. Cheng, R., Xia, X., Prabhakar, S., Shah, R. and Vitter, J. "Efficient Indexing Methods for Probabilistic Threshold Queries over Uncertain Data," *Proceedings of VLDB*, 2004.
6. de Souza, R. M. C. R. and de Carvalho, F. de A. T. "Clustering of Interval Data Based on City–Block Distances," *Pattern Recognition Letters*, 25 (2004) 353–365.
7. Dunn, J. C. "A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact Well-Separated Clusters," *Journal of Cybernetics*, 3 (1973) 32-57.
8. Hamdan, H. and Govaert, G. "Mixture Model Clustering of Uncertain Data," *IEEE International Conference on Fuzzy Systems* (2005) 879-884.
9. Ichino, M., Yaguchi, H. "Generalized Minkowski Metrics for Mixed Feature Type Data Analysis," *IEEE Transactions on Systems, Man and Cybernetics*, 24(4) (1994) 698–708.
10. Jain, A. and Dubes, R. *Algorithms for Clustering Data*. Prentice Hall, New Jersey (1988).
11. Nilesh N. D. and Suciu, D. "Efficient Query Evaluation on Probabilistic Databases," *VLDB* (2004) 864-875.
12. Pfoser D. and Jensen, C. "Capturing the Uncertainty of Moving-objects Representations," *Proceedings of the SSDBM Conference*, 123–132, 1999.
13. Ruspini, E. H. "A New Approach to Clustering," *Information Control*, 15(1) (1969) 22-32.
14. Sato, M., Sato, Y., and Jain, L. *Fuzzy Clustering Models and Applications*. Physica-Verlag, Heidelberg (1997).
15. Wolfson, O., Sistla, P., Chamberlain, S. and Yesha, Y. "Updating and Querying Databases that Track Mobile Units," *Distributed and Parallel Databases*, 7(3), 1999.
16. Yeung, K. and Ruzzo, W. "An Empirical Study on Principal Component Analysis for Clustering Gene Expression Data," *Bioinformatics*, 17(9) (2001) 763-774.