A Study on Distance-based Outlier Detection on Uncertain Data

Salman Ahmed Shaikh
Graduate School of Systems and Information Engineering, University of Tsukuba, Japan
Email: salman@kde.cs.tsukuba.ac.jp

Uncertain data management, querying and mining have become important because the majority of real world data is accompanied with uncertainty these days. Uncertainty in data is often caused by the deficiency in underlying data collecting equipments or sometimes manually introduced to preserve data privacy. The uncertainty information in the data is useful and can be used to improve the quality of the underlying results. Therefore in this dissertation, three problems are being solved related to outlier detection on uncertain data. 1) Distance-based outlier detection on uncertain data: In this research, we give a novel definition of distance-based outliers on uncertain data. Since the distance probability computation is expensive, a cell-based approach is proposed to index the dataset objects and to speed up the outlier detection process. The cell-based approach identifies and prunes the cells containing only inliers based on its bounds on outlier score (#*D*-neighbors). Similarly it can also detect the cells containing only outliers. 2) Top-*k* outlier detection on uncertain data: In this work, a top-*k* distance-based outlier detection approach is presented. In order to detect top-*k* outliers from uncertain data efficiently, we propose a data structure, populated-cells list (PC-list). Using the PC-list, the top-*k* outlier detection algorithm needs to consider only a fraction of the dataset objects and hence quickly identifies candidate objects for the top-*k* outliers. 3) Continuous outlier detection on uncertain data streams: In this part of the dissertation, a distance-based approach is proposed to detect outliers continuously from a set of uncertain objects' states that are originated synchronously from a group of data sources (e.g., sensors in WSN). A set of objects' states at a timestamp is called a state set. Usually, the duration between two consecutive timestamps is very short and the state of all the objects may not change much in this duration. Therefore, to eliminate the unnecessary computation at every timestamp, an incremental approach of outlier detection is proposed which makes use of outlier detection results obtained from the previous timestamp to detect outliers in the current timestamp. Finally, extensive experimental evaluations on real and synthetic datasets are presented for each of the proposed outlier detection approaches, to prove their accuracy, efficiency and scalability.

I. Introduction

Outlier detection is a fundamental problem in data mining. It has applications in many domains including credit card fraud detection, network intrusion detection, environment monitoring, medical sciences etc. Several definitions of outlier have been given in past, but there exists no universally agreed upon definition. Hawkins [1] defines an outlier as an observation that deviates so much from other observations as to arouse suspicion that it was generated by a different mechanism.

In statistics, one can find over 100 outlier detection techniques. These have been developed for different data distributions, parameters, desired number of outliers and type of expected outliers [2]. However, most of the statistical techniques are not useful due to several reasons. For example, the majority of the statistical techniques are univariate, in some techniques parameters are difficult to determine, and in other techniques outliers cannot be obtained until the underlying data distribution is known. In order to overcome these problems, several outlier detection techniques have been proposed in the data mining including nearest-neighbour based, density-based, clustering-based and distance-based [3], [5]. In this dissertation, our focus is the distance-based approach because the distance-based approach is the simplest and the most commonly used. It can be used as preprocessing before applying more sophisticated application dependent outlier detection techniques. Moreover, it coincides well with other data-mining techniques, e.g., *k*-NN, clustering, etc.

The very first definition of distance-based outlier was given by Knorr, et al. [3]. They defined an object o to be an outlier if at most $M = N(1 - p)$ objects are within D-distance of o, where N is the number of objects in the dataset and p is the fraction of objects that lie farther than the D-distance of o. They also presented a cell-based approach to efficiently compute the distance-based outliers. S. Ramaswamy, et al. [5] formulated distance-based outliers as the top-t data objects whose distances to their k^{th} nearest neighbour is largest. Angiulli et al. [6] gave a slightly different definition of outliers than S. Ramaswamy, et al. [5] by considering the average distance to their k nearest neighbours. Beside these, there are some works on the detection of the distance-based outliers over stream data [7], [8]. These works are based on the definition of distance-based outliers by Knorr, et al. [3]. Furthermore, F. Angiulli, et al. [7] gave an approximate algorithm to reduce the memory space required by its exact counterpart. Later on M. Kontaki, et al. [8] extended F. Angiulli, et al. work [7] by adding the concepts of multi-query and micro-cluster based distance-based outlier detection. However all these approaches were given for the deterministic data and cannot handle the uncertain data.

Due to the increasing usage of sensors, RFIDs, GPS and similar devices for data collection these days, data contains certain degree of inherent uncertainty [4]. The causes of uncertainty may include but are not limited to limitation of equipments, absence of data, inconsistent supply voltage and delay or loss of data in transfer [4]. In order to get reliable results from such data, uncertainty needs to be considered in calculation.

Driven by emerging requirements, recently a lot of research has focused on managing, querying and mining of the uncertain data. The problem of outlier detection on the uncertain data was first studied by Aggarwal, et al. [9]. According to them, an uncertain object o is a density-based (δ, η) outlier, if the probability of existence of the o in some subspace of a region with density at least η is less than δ. In order to compute (δ, η) outliers, firstly density of all the subspaces needs to be computed and then the η-probability of each o in the dataset is computed to tell if o is an outlier. Since this computation is very expensive, a sampling procedure is used to approximate the η-probability. In contrast to the Aggarwal, et al. work [9], this work addresses the detection of the distance-based outliers in full space, where the distance probability between two uncertain objects is computed by the Gaussian difference distribution [10]. Therefore, the problem definition is quite different from Aggarwal, et al. [9].

In this dissertation, our focus is the distance-based outlier detection from the uncertain data of the Gaussian distribution. We mainly used a cell-based technique to speed-up the outlier detection process. Our main contributions are summarized as follows.

- **UDB-Outlier Detection:** A cell-based approach of the distance-based outlier detection on uncertain static data, summarized in Section II.
- ***k*UDB Outlier Detection:** A top-k outlier detection approach on uncertain data, summarized in Section III.
- **CUDB-Outlier Detection:** A continuous outlier detection approach on uncertain data streams, presented in Section IV.

II. Distance-based Outlier Detection on Uncertain Data (UDB Outlier Detection)

In this section, a distance-based outlier detection approach on uncertain static data is presented. To obtain distance-based outliers, distance probability needs to be computed between uncertain data objects. However this computation is very costly. Therefore a cell-grid structure is used to index dataset objects and prune inliers and identify outliers.

In this work, d-dimensional uncertain objects o_i are considered, with attribute $\overrightarrow{\mathcal{A}_i} = (x_{i,1}, ..., x_{i,d})$ following the Gaussian PDF with mean $\overrightarrow{\mu_i} = (\mu_{i,1}, ..., \mu_{i,d})$ and co-variance matrix $\Sigma_i = diag(\sigma^2_{i,1}, ..., \sigma^2_{i,d})$, respectively. Namely, the vector $\overrightarrow{\mathcal{A}_i}$ is a random variable that follows the Gaussian distribution $\overrightarrow{\mathcal{A}_i} \sim \mathcal{N}(\overrightarrow{\mu_i}, \Sigma_i)$. Note that $\overrightarrow{\mu_i}$ denotes the observed coordinates (attribute values) of object o_i. The complete database consists of a set of such objects, $\mathcal{GDB} = \{o_1, ..., o_N\}$, where $N = |\mathcal{GDB}|$. Hence a distance-based outlier on the uncertain dataset is defined as follows.

Definition 1: An uncertain object o in a database $\mathcal{GDB}$ is a distance-based outlier, if the expected number of objects $o_i \in \mathcal{GDB}$ (including o itself) lying within D-distance of o is less than or equal to threshold $\theta = N(1-p)$, where N is the number of uncertain objects in database $\mathcal{GDB}$, and p is the fraction of objects in $\mathcal{GDB}$ that lies farther than D-distance of o.

The objects that lie within the D-distance of o_i are called its *D-neighbors*, and the set of the D-neighbors of o_i and the number of D-neighbours are denoted by $DN(o_i)$ and #D-neighbors(o_i), respectively. The distance probability between two uncertain objects (o_i and o_j) following the Gaussian distribution is computed using the difference between two Gaussian distributions, which is given by another distribution known as the Gaussian difference distribution [10], $|\overrightarrow{\mathcal{A}_i} - \overrightarrow{\mathcal{A}_j}| = \mathcal{N}(\overrightarrow{\mu_i} - \overrightarrow{\mu_j}, \Sigma_i + \Sigma_j)$ [10]. Let $Pr(o_i, o_j, D)$ denotes the probability that the $o_j \in DN(o_i)$ and is defined as follows.

$$Pr(o_i, o_j, D) = \int_R \mathcal{N}(\overrightarrow{\mu_i} - \overrightarrow{\mu_j}, \Sigma_i + \Sigma_j) \mathrm{d}\overrightarrow{\mathcal{A}}, \qquad (1)$$

where R is a sphere with centre $(\overrightarrow{\mu_i} - \overrightarrow{\mu_j})$ and radius D.

Cell-based Approach: Since the naive approach of the distance-based outlier detection makes use of nested loop and is very expensive, a cell-based approach of outlier detection is proposed in this work. To identify the distance-based outliers using the cell-based approach, each object in $\mathcal{GDB}$ is quantized to a 2-dimensional grid $\mathcal{G}$ that is partitioned into cells of length l (l is user-defined parameter). Let $C_{x,y}$ be a cell in $\mathcal{G}$, where positive integers x and y denote the cell indices. The layers $(L_1, ..., L_n)$ of $C_{x,y} \in \mathcal{G}$ are the neighbouring cells of $C_{x,y}$ and are defined as follows.

$$L_1(C_{x,y}) = \{C_{u,v} | u = x \pm 1, v = y \pm 1, C_{u,v} \neq C_{x,y}\}.$$
$$L_2(C_{x,y}) = \{C_{u,v} | u = x \pm 2, v = y \pm 2, C_{u,v} \notin L_1(C_{x,y}), C_{u,v} \neq C_{x,y}\}.$$

$L_3(C_{x,y}), ..., L_n(C_{x,y})$ are defined in a similar way. In the sequel, we will use C to denote $C_{x,y}$ when there is no confusion. To identify the cells $C \in \mathcal{G}$, containing only inliers or outliers, their bounds on the #D-neighbours are used. The upper bound of a cell C, $UB(C)$, binds the maximum #D-neighbors for any object in C, while the lower bound, $LB(C)$, binds the minimum #D-neighbours for any object in C. To compute a cell bounds, its and its layers object count, minimum and maximum ordinary Euclidean distances between C and its layers and pre-computed $Pr(\alpha, D)$ values are used. Interested readers may refer to our work [11] for the details on bounds calculation.

Based on the bounds, a cell $C \in \mathcal{G}$ can be pruned as inlier if $LB(C) > \theta$. On the other hand, if $UB(C) \leq \theta$, the C is identified as an outlier cell. There may be some un-pruned objects in the un-pruned cells even after the cell-based pruning, for all such objects, the naive computation is used.

III. Top-k Outlier Detection on Uncertain Data (kUDB Outlier Detection)

In the approach presented in Section II, an object can be either classified as an outlier or an inlier. Different parameter combinations return varying number of outliers and no outlier ranking is available. Therefore in this section, a top-k approach of the distance-based outliers on uncertain data is presented and is defined as follows.

Definition 2: The top-k distance-based outliers are the k uncertain objects in the dataset $\mathcal{G}DB$ for which the expected number of objects $o_i \in \mathcal{G}DB$ lying within the D-distance is smallest.

To identify the top-k outliers efficiently, a populated-cells list (PC-list) based approach is presented. A PC-list is an array of non-empty cells of the grid $\mathcal{G}$, where grid is used to index dataset objects. The cell-grid used in this section is similar to the one defined in Section II except the regions $R_{D-t\sigma}$ and $R_{D+t\sigma}$.

The $R_{D-t\sigma}(C)$ denotes a region formed by $\left\lfloor \frac{D-t\sigma}{l\sqrt{d}} - 1 \right\rfloor$ neighbouring layers of $C \in \mathcal{G}$ as shown in Fig.2. The region $R_{D-t\sigma}(C)$ is chosen in such a way that for each $o_i \in C$ and $o_j \in R_{D-t\sigma}(C)$, $Pr(o_i, o_j, D) \approx 1$. Similarly, the $R_{D+t\sigma}(C)$ denotes a region formed by $\left\lceil \frac{D+t\sigma}{l} \right\rceil$ neighbouring layers of cell $C \in \mathcal{G}$ as shown in Fig.1. Region $R_{D+t\sigma}(C)$ is chosen in such a way that for each $o_i \in C$ and $o_j \notin R_{D+t\sigma}(C)$, the $Pr(o_i, o_j, D)$ approaches zero.

Let $N(C)$ be the number of objects in C, and $N_{D-t\sigma}(C)$ be the number of objects within cells in the region $R_{D-t\sigma}(C)$ (including C itself). Then the PC-list (PC) is a sorted list containing $N(C)$ and $N_{D-t\sigma}(C)$ for each non-empty cell $C \in \mathcal{G}$ as shown in Fig.2. The tuples in the PC-list are sorted in an ascending order of $N_{D-t\sigma}(C)$ column. The idea behind sorting is that outliers tend to exist in sparse regions. Sorting tuples in the PC-list, lets us identify cells with few number of neighbouring objects or cells in sparse regions.

A cell C can be pruned as the inlier cell or identified as the cell containing the top-k outlier candidates using its bounds on the #D-neighbors. To compute the cell bounds, the minimum and the maximum ordinary Euclidean distances between cells are required. Besides this, the object count of each $C \in PC$ and the $Pr(\alpha, D)$ values for α ranging from the minimum to the maximum ordinary Euclidean distances between cells in the $\mathcal{G}$ are also required. The $Pr(\alpha, D)$ values are precomputed and stored in a look-up table to be used by the top-k outlier detection algorithm.

Let $\mathbb{C}_{cell}$ is a list for holding the candidate outlier cells from the PC-list, sorted in an ascending order of the $UB(C)$. Let $C^k \in \mathbb{C}_{cell}$ is a cell with the minimum upper bound containing the k^{th} object. A $C \in PC$ is a candidate outlier cell whenever $\sum_{C' \in \mathbb{C}_{cell}} N(C') < k$ or $LB(C) \leq \theta$, where $\theta = UB(C^k)$ denotes the threshold.

For a $C \in PC$, if the $LB(C) > \theta$, the C cannot contain any of the top-k outliers and can be pruned. On the other hand, if the $LB(C) \leq \theta$, the C may contain top-k outliers. The C is added to the $\mathbb{C}_{cell}$, such that the $\mathbb{C}_{cell}$ remain sorted of its $UB(C)$ attribute. Set the $\theta = UB(C^k)$ and remove the C' from $\mathbb{C}_{cell}$, such that the $LB(C') > \theta$, as they cannot contain the top-k outliers.

The PC-list is scanned from top to bottom for the candidate outlier cells. During the scanning, if a $C' \in PC$ is found such that $Pr(D - t\sigma, D) * N_{D-t\sigma}(C') > \theta$, which is a lower bound on #D-neighbours of C', the C' cannot contain the top-k outliers and can be pruned. Since the PC-list is sorted of the $N_{D-t\sigma}(C)$, any cell after the C' must have $N_{D-t\sigma}(C) \geq N_{D-t\sigma}(C')$. Hence the lower bound of $C \in PC$ after C' must be greater than or equal to the lower bound of $C' \in PC$ and cannot contain the top-k outliers. Hence the PC-list scanning can be stopped at this point.

Finally, exact #D-neighbors is computed for each object in the candidate outlier cells to find the top-k outliers and their ranking. Two approximate top-k outlier detection algorithms are also presented in this work to increase the efficiency of the outlier detection. The first approximate algorithm only approximates the candidate objects' #D-neighbors, while the second approximate algorithm makes use of the bounded Gaussian uncertainty to increase the efficiency of the top-k outlier detection algorithm.

IV. Continuous Outlier Detection on Uncertain Data Streams (CUDB Outlier Detection)

Since the data obtained from sensors, RFIDs and similar devices is continuous and contain certain degree of uncertainty, this section presents a continuous outliers detection approach on uncertain data streams. Here streams are the sequences of objects' states generated over time. Like the previous works, this work assumes that the object's uncertainty follows the Gaussian distribution. Assuming that there are N objects whose states may change over time and let $S^j = \{\overrightarrow{\mathcal{A}_1^j}, ..., \overrightarrow{\mathcal{A}_N^j}\}$ denotes a set of such objects at time t_j, then the distance-based outlier in uncertain data streams is defined as follows.

Definition 3: An uncertain object o_i is a distance-based outlier at time t_j, if the expected number of objects in S^j lying within the D-distance of o_i are less than or equal threshold to $\theta = N(1 - p)$, where N is the number of objects, and p is the fraction of objects that lie farther than the D-distance of $o_i \in S^j$.

A straightforward approach to identify the distance-based outliers from the set of objects S^j at every time t_j, is to execute the cell-based algorithm of Section II on S^j at every time instance. However according to our assumption, states remain unchanged between two time instances for most of the objects. Hence, processing all the objects in all the state sets is meaningless. Therefore, we propose an incremental outlier detection approach based on change between S^{j-1} and S^j using the cell-based algorithm.

Let *SC-Object* denotes an object whose state have changed from t_{j-1} to t_j and SC^j denotes a set of such objects at time t_j. A differential processing is used to detect outliers from SC^j. The idea is to process only the objects which are either themselves SC-Objects or are affected by the SC-Objects. In this work, the cell-based approach of Section II is utilized to process the SC-Objects.

To simplify the problem, consider the case with one SC-Object, o_p. Let $o_i \in C^j_{x,y}$ represent a cell $C_{x,y}$ containing o_i at time t_j. As a result of state change, $o_p \in \mathcal{G}$ can move in the following two ways.

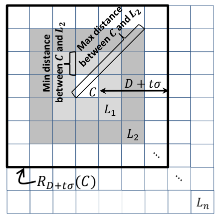
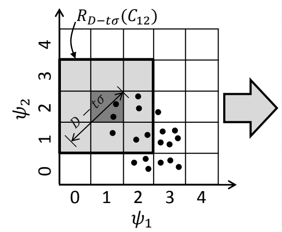


Fig. 1: Cell Layers and Bounds



Cell	$N(C)$	$N_{D-t\sigma}(C)$
C_{12}	2	7
C_{11}	1	10
C_{32}	1	10
C_{22}	2	13
C_{30}	3	13
C_{20}	3	14
C_{31}	5	16
C_{21}	2	19

Fig. 2: PC-list building

[Case 1] o_p moved to a different cell:

$$o_p \in C^{j-1}_{x,y}, o_p \in C^{j}_{x,y}, C^{j-1}_{x,y} \neq C^{j}_{x,y}.$$

In this case, o_p affects cells $C^{j-1}_{x,y}$, $C^{j}_{x,y}$, their $R_{D-t\sigma}$ and $R_{D+t\sigma}$ neighboring regions and all the objects in the un-pruned cells.

[Case 2] o_p moved within a cell:

$$o_p \in C^{j-1}_{x,y}, o_p \in C^{j}_{x,y}, C^{j-1}_{x,y} = C^{j}_{x,y}.$$

Since the number of objects in the $C^{j}_{x,y}$ remain same as a result of the o_p movement, the cell-based pruned cells are not affected. However all the objects in the un-pruned cells are affected, because the objects in the un-pruned cells require the computation of their #D-neighbors. Since the #D-neighbors computation of an object o_p require the $Pr(o_p, o_q, D)$ computation of o_p and each $o_q \in S^j$, even a slight movement in any dataset object can influence its #D-neighbors.

In practise, there are more than one SC-Objects from time t_{j-1} to t_j. Therefore, we expand the above idea to more than one SC-Objects. For the incremental outlier processing, the cells requiring re-processing as a result of state change are identified. Such cells are called the *Target cells*. Hence there are following three types of target cells.

Type A: Cells containing SC-Objects which have moved to or from another cell at time t_j.

Type B: $R_{D-t\sigma}$ and $R_{D+t\sigma}$ neighbouring cells of Type A cells, except those classified as Type A cells.

Type C: Un-pruned cells of the grid $\mathcal{G}$. Type C cells may include Type A and B cells.

All three cell types, i.e., A, B and C require re-outlier detection at time instance t_j.

The main cost of the proposed approach lies in the processing of un-pruned objects in the Type C cells. The un-pruned objects processing require the computation of their #D-neighbors. This is very expensive because the #D-neighbors computation of an object o_p require the $Pr(o_p, o_q, D)$ computation of o_p and each $o_q \in S$. In the incremental algorithm, this cost can be reduced by utilizing the #D-neighbors computed in the previous state. Namely, a Hash table is used to store the $Pr(o_p, o_q, D)$ values computed at time t_{j-1}. At time t_j, these values can be retrieved from the Hash table in $O(1)$ time. Hence at time t_j, the $Pr(o_p, o_q, D)$ values need to be computed only in two cases; 1) States of o_p, o_q or both have changed, 2) $Pr(o_p, o_q, D)$ is not available in the Hash table. Since the un-pruned objects form a fraction of the complete dataset, the memory required to hold the Hash table is not significant. However it saves a lot of computation time. An approximate approach using the bounded Gaussian uncertainty is also proposed to increase the outlier detection efficiency.

V. Conclusions and Future Work

In this work, we addressed the problem of distance-based outlier detection on uncertain static data and uncertain data streams and proposed three approaches. The UDB-Outlier and the kUDB-Outlier are proposed for uncertain static data while the CUDB-Outlier for uncertain data streams. The accuracy, efficiency and scalability of the proposed approaches are proved by extensive experiments on real and synthetic datasets (Please refer to the thesis for the detailed experiments).

In the future, one of the natural extension of this work is the detection of the top-k outliers from uncertain data streams. Since in the current work, object's uncertainty is given by the Gaussian distribution, its extension to other uncertainty models could be another interesting future direction. Moreover, extension of this work to handle very high dimensional data is another useful future direction.

References

[1] Hawkins, D.: Identification of Outliers. Chapman and Hall, London (1980)
[2] Maimon, O., Rockach, L.: Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers. Kluwer Academic, Norwell (2005)
[3] Knorr, E.M., Ng, R.T., Tucakov, V.: Distance-based outliers: algorithms and applications. VLDBJ. 8(3-4), 237-253 (2000)
[4] Sharma, A.B., Golubchik, L., Govindan, R.: Sensor faults: detection methods and prevalence in real-world datasets. ACM Trans. Sens. Netw. 6(3), 23:1-39 (2010)
[5] Ramaswamy, S., Rastogi, R., Shim, K.: Efficient algorithms for mining outliers from large datasets. In: ACM SIGMOD, pp. 427-438 (2000)
[6] Angiulli, F., Pizzuti, C.: Fast outlier detection in high dim. spaces. In: PKDD, pp. 15-26 (2002)
[7] Angiulli, F., Fassetti, F.: Detecting distance-based outliers in streams of data. In: CIKM, pp. 811-820 (2007)
[8] Kontaki, M., Gounaris, A., Papadopoulos, A.N., Tsichlas, K., Manolopoulos, Y.: Continuous monitoring of distance-based outliers over data streams. In: ICDE, pp. 135-146 (2011)
[9] Aggarwal, C.C., Yu, P.S.: Outlier detection with uncertain data. In: SIAM International Conference on Data Mining, pp. 483-493 (2008)
[10] Weisstein, E.W.: Normal Difference Distribution. From MathWorldA Wolfram Web Resource. http://mathworld.wolfram.com/NormalDifferenceDistribution. Accessed 27 Jan 2012
[11] Shaikh, S.A., Kitagawa, H.: Efficient Distance-based Outlier Detection on Uncertain Datasets of Gaussian Distribution. World Wide Web, pp. 1–28 (2013)