DEIM Forum 2012 D3-1

Outlier Detection on Uncertain Data of Gaussian Distribution

シエイクサルマンアーメット† 北川 博之†

† 筑波大学大学院システム情報工学研究科 〒 305-8573 つくば市天王台 1-1-1
E-mail: †salman@kde.cs.tsukuba.ac.jp, ††kitagawa@cs.tsukuba.ac.jp

Outlier Detection on Uncertain Data of Gaussian Distribution

Salman Ahmed SHAIKH† and Hiroyuki KITAGAWA†

† Graduate School of Systems and Information Engineering, University of Tsukuba
Tennodai, Tsukuba, Ibaraki 305-8573, Japan
E-mail: †salman@kde.cs.tsukuba.ac.jp, ††kitagawa@cs.tsukuba.ac.jp

Abstract Managing, querying and mining in uncertain data is becoming important because majority of real world data is accompanied with uncertainty these days. Uncertainty in data is often caused by the deficiency in underlying data collecting equipment or sometimes manually introduced to preserve the data privacy. In this work, we propose a notion of distance-based outlier detection on uncertain data of Gaussian distribution. In order to reduce the cost of complex distance function computation, we propose a cell based approach. We also propose use of 3σ filtering to further reduce the cost of computation. An empirical study on both real and synthetic data verifies the effectiveness of our proposed approach.

Key words Outlier Detection, Uncertain Data, Cell Based Approach

1. Introduction

Outlier detection is one of the most important data mining technique with a vital importance in many application domains including credit card fraud detection [16], network intrusion detection [14], environment monitoring [17], medical sciences [15] etc. Although there exists no any universal agreed upon definition of outliers, yet some definitions are general enough to give a basic idea of outliers. Hawkins [6] 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 [7], Barnet and Lewis mentioned that an outlying observation, or outlier, is one that appears to deviate markedly from other members of the sample in which it occurs.

Most of the earliest outlier detection techniques were given by statistics. In statistics over 100 outlier detection techniques have been developed for different circumstances, depending on the data distribution, whether or not the distribution parameters are known, the number of expected outliers and the type of expected outliers [7], [18] but most statistical techniques are univariate and in majority of techniques, the parameter of distribution may be difficult to determine. In order to overcome problems in statistical techniques several Distance-based approaches for Outlier detection are proposed in computer science [4], [13], [8], [9].

Uncertainty. Due to the incremental usage of sensors, RFIDs and similar devices for data collection these days, data contains certain degree of inherent uncertainty. The causes of uncertainty may include limitation of equipments, absence of data and delay or loss of data in transfer. In order to get reliable results from such a data, uncertainty needs to be considered in calculation. In this work we propose a notion of distance-based outliers on uncertain data.

In the following, uncertainty of data is modelled by the most commonly used PDF i.e., Gaussian distribution. We derived a distance function using Gaussian difference distribution to compute the distance-based outliers on uncertain data. Our distance function includes the integral of irreducible function, which makes the distance function computation very costly. Therefore we also propose a cell-based algorithm of outlier detection to efficiently compute the distance-based outliers on uncertain data. The cell-based algorithm prunes objects by identifying outliers or pruning non-outliers without the need to compute costly distance function, hence reducing the number of distance function evaluation required. This work is an extension of our work published in [1]. The major contribution of this work compared to our work in [1] is the use of 3σ-filtering to improve the efficiency of our cell-based algorithm of distance-based outlier detection on uncertain data. Finally grid structure is used to further reduce the computation time required for distance-based outlier detection.

The rest of the paper is organized as follows. Section 2. surveys the previous work related to ours. Section 3. gives the derivation of distance function. Section 4. discusses the naive algorithm of distance-based outlier detection on uncertain data. The cell-based algorithm is given in section 5. Section 6. is dedicated to empirical study and Section 7. concludes our paper.

2. Related Work

Outlier detection is a well studied area of data mining. Different authors have classified this area differently. The problem of outlier detection has been classified into statistical approaches, depth-based approaches, deviation-based approaches, distance-based approaches, density-based approaches and high-dimensional approaches by [10].

Distance-based outliers detection approach on deterministic data was introduced by Knorr et al. in [4]. In this work, the authors defined a point p to be an outlier if at most M points are within d distance of the point. They also presented a cell-based algorithm to efficiently compute the distance-based outliers. [11] formulated distance-based outliers based on the distance of a point from its kth nearest neighbour. The points were ranked on the basis of its distance to its kth nearest neighbour and the top n points were declared outliers in this ranking. H.V.Nguyen et al. in [19] proposed subspace outlier detection method for high dimensional data. In order to detect outliers, subspace outlier score function $FS_{out}(p, S)$ of a point p with respect to its k nearest neighbors in a subspace S is evaluated. Top n points are then selected as outliers in any subspace whose outlier scores are largest. Recently in [8], the authors assessed and evaluated several distance-based outlier detection approaches and highlighted a family of state of the art distance-based outlier detection algorithms. A cell-based approach of outlier detection for very large dataset which cannot be loaded into memory at once was given by [20]. In their approach, a large dataset is loaded into memory by blocks and the data are placed into appropriate cells based on their values. Cell's density is calculated and the data located in high density cells are filtered from outlier calculation. Cells' densities are recorded for the next block of data. The final calculation for outliers is done on the data in low denstiy cells.

Recently a lot of research has focused on managing, querying and mining of uncertain data [12], [9], [21] due to the use of sensors in many applications. The problem of outlier detection on uncertain data was first studied by Aggarwal et.al. in [12]. They represented an uncertain object by a PDF. They defined an uncertain object O to be a density-based (δ, η) outlier, if the probability of O existing in some subspace of a region with density at least η is less than δ. In [9], the authors proposed the distance-based outlier detection on uncertain data. In their approach, each tuple in the uncertain table is associated with an existential probability. Moreover in their work, possible world semantic was used to mine the outliers. B. Jiang et al. in [21] gave an outlier detection model considering both uncertain objects and their instances. According to their model an uncertain object is outlier if majority of its instances are outliers and an object's instance is outlier if its normality (opposite of outlierness) is less than some threshold. In their work, Baye's theorem is used to calculate the normality of object's instance. In our work, objects' uncertainty is modelled by Gaussian distribution and we utilize Gaussian difference distribution to calculate the outlier probability.

3. Distance-based Outlier on Uncertain Data

Several definitions of distance-based outliers have been proposed in past. In this paper, we extend the notion of distance-based outliers given by E.M.Knorr et.al. in [4] for uncertain data of Gaussian distribution.

In statistics, the Gaussian distribution *(or normal distribution)* is the most important and the most commonly used distribution. In the following, we consider k-dimensional uncertain objects O_i, each given by a Gaussian PDF with attribute $\overrightarrow{\mathcal{A}_i} = (x_{i,1}, ..., x_{i,k})^T$, mean $\overrightarrow{\mu_i} = (\mu_{i,1}, ..., \mu_{i,k})^T$ and variance $\Sigma_i = diag(\sigma^2_{i,1}, ..., \sigma^2_{i,k})$ respectively. The complete database consists of a set of such objects, $\mathcal{GDB} = \{O_1, ..., O_N\}$ where $N = |\mathcal{GDB}|$ is the number of uncertain objects in $\mathcal{GDB}$. The vector $\overrightarrow{\mathcal{A}_i}$ is a random variable of the corresponding uncertain objects that follows Gaussian distribution $\overrightarrow{\mathcal{A}_i} \sim \mathcal{N}(\overrightarrow{\mu_i}, \Sigma_i)$.

We assume that the observed coordinates are $\overrightarrow{\mu_i}$ vectors of the objects which follow Gaussian distribution. Based on this assumption, in the rest of the paper we will use $\overrightarrow{\mu_i}$ to denote the real observed coordinates of object O_i. We can now define the distance based outliers on uncertain data of Gaussian distribution as follows.

Definition. *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 $\mathcal{GDB}$, uncertain objects in $\mathcal{GDB}$ follow Gaussian distribution and p is the fraction of objects in $\mathcal{GDB}$ that lies farther than d-distance of O.*

According to the definition above, the set of uncertain distance-based outliers in $\mathcal{GDB}$ is defined as follows,

$$\mathcal{UDBO}utliers = \{O_i \in \mathcal{GDB} | \sum_{j=1}^{|\mathcal{GDB}|} Pr(|\overrightarrow{\mathcal{A}_i} - \overrightarrow{\mathcal{A}_j}| \leqq d) \leq \theta\} \ . \tag{1}$$

In order to find distance-based outliers in $\mathcal{GDB}$, the distance between Gaussian distributed objects need to be calculated. In the following we define and derive the expressions for difference between Gaussian distributed objects.

3. 1 Gaussian Difference Distribution

The distribution of the difference of two Gaussian distributed variates O_i and O_j with means and variances (μ_i, σ_i^2) and (μ_j, σ_j^2) respectively, is given by another Gaussian distribution with mean $\mu_{i-j} = \mu_i - \mu_j$ and variance $\sigma_{i-j}^2 = \sigma_i^2 + \sigma_j^2$ [2]. Hence we can write $\mathcal{A}_i - \mathcal{A}_j \sim \mathcal{N}(\mu_{i-j}, \sigma_{i-j}^2)$.

1-Dimensional Gaussian Difference Distribution within Distance d

Using Gaussian difference distribution, the probability that the uncertain object O_i lies within d-distance of uncertain object O_j is given by,

$$Pr(|\mathcal{A}_i - \mathcal{A}_j| \leqq d) = \int_{-d}^{d} \mathcal{N}(\mu_{i-j}, \sigma_{i-j}^2)\mathrm{d}x \ , \tag{2}$$

where $\mathcal{A}_i \sim \mathcal{N}(\mu_i, \sigma_i^2)$ and $\mathcal{A}_j \sim \mathcal{N}(\mu_j, \sigma_j^2)$.

2-Dimensional Gaussian Difference Distribution within Distance d

The expression for the 2-dimensional Gaussian difference distribution is defined in Lemma 1 below.

Lemma 1.(2D Gaussian Difference Distribution within Distance d**)** *let $\overrightarrow{\mathcal{A}_i} \sim \mathcal{N}(\overrightarrow{\mu_i}, \Sigma_i)$ and $\overrightarrow{\mathcal{A}_j} \sim \mathcal{N}(\overrightarrow{\mu_j}, \Sigma_j)$ be two 2-dimensional Gaussian distributed variates, where $\overrightarrow{\mu_i} = (\mu_{i,1}, \mu_{i,2})^T$, $\overrightarrow{\mu_j} = (\mu_{j,1}, \mu_{j,2})^T$, $\Sigma_i = diag(\sigma_{i,1}^2, \sigma_{i,2}^2)$ and $\Sigma_j = diag(\sigma_{j,1}^2, \sigma_{j,2}^2)$. The probability that O_i lies within d-distance of O_j is given by,*

$$Pr(|\overrightarrow{\mathcal{A}_i} - \overrightarrow{\mathcal{A}_j}| \leqq d) = \frac{1}{2\pi\sqrt{(\sigma_{i,1}^2 + \sigma_{j,1}^2)(\sigma_{i,2}^2 + \sigma_{j,2}^2)}} \int_0^d \int_0^{2\pi} \exp\left\{ -\left(\frac{(r\cos\theta - \alpha_1)^2}{2(\sigma_{i,1}^2 + \sigma_{j,1}^2)} + \frac{(r\sin\theta - \alpha_2)^2}{2(\sigma_{i,2}^2 + \sigma_{j,2}^2)} \right) \right\} r \, d\theta \, dr \ , \tag{3}$$

where $\alpha_1 = \mu_{i,1} - \mu_{j,1}$ and $\alpha_2 = \mu_{i,2} - \mu_{j,2}$ are the differences between the means of objects O_i and O_j.

Proof. See appendix.

Multidimensional Gaussian Difference Distribution within Distance d

Our distance function can easily be extended to multi-dimension case. Let $\overrightarrow{\mathcal{A}_i}$ and $\overrightarrow{\mathcal{A}_j}$ be two k-dimensional normal random vectors with means $\overrightarrow{\mu_i} = (\mu_{i,1}, ..., \mu_{i,k})^T$ and $\overrightarrow{\mu_j} = (\mu_{j,1}, ..., \mu_{j,k})^T$ and diagonal covariance matrices $\Sigma_i = diag(\sigma_{i,1}^2, ..., \sigma_{i,k}^2)$ and $\Sigma_j = diag(\sigma_{j,1}^2, ..., \sigma_{j,k}^2)$ respectively. The probability that the uncertain object O_i lies within d distance of uncertain object O_j is given by,

$$Pr(|\overrightarrow{\mathcal{A}_i} - \overrightarrow{\mathcal{A}_j}| \leqq d) = \int_R \mathcal{N}(\overrightarrow{\mu_{i-j}}, \Sigma_{i-j})\mathrm{d}x \ , \tag{4}$$

where $\overrightarrow{\mu_{i-j}} = \overrightarrow{\mu_i} - \overrightarrow{\mu_j}$, $\Sigma_{i-j} = \Sigma_i + \Sigma_j$ and R is a sphere with centre $\overrightarrow{\mu_{i-j}}$ and radius d.

Algorithm 1 Distance-based Outlier on Uncertain Data: The NL Approach

```
Input: database $\mathcal{GDB}$, distance $d$, percentage $p$ , standard deviation $\sigma$
Output: Uncertain Distance Based Outliers
  $N \leftarrow$ number of objects in $\mathcal{GDB}$;
  $\theta \leftarrow N(1-p)$;            /*calculating the threshold value*/
  for each $O$ in $\mathcal{GDB}$ do
    $EV_O \leftarrow 0$; /*$EV_O$ denotes the expected value of object $O$*/
    for each $O_i$ in $\mathcal{GDB}$ do
      $EV_O = EV_O + Pr(|\overrightarrow{\mathcal{A}} - \overrightarrow{\mathcal{A}_i}| \leqq d)$;
      if $EV_O > \theta$ then
        mark $O$ as non-outlier, GOTO next $O$;
      end if
    end for
    mark $O$ as outlier;
  end for
```

4. Naive Approach

The Naive approach of distance-based outlier detection on uncertain data is the use of Nested-loop. The approach includes the evaluation of distance function between each object $O_i \in \mathcal{GDB}$ and every other object in the $\mathcal{GDB}$ until O_i may be decided as outlier or non-outlier. In the worst case this approach requires the evaluation of $O(N^2)$ distance functions. The algorithm 1 gives the naive approach of distance based outliers.

5. Cell-based Approach

The naive approach of distance-based outlier detection on uncertain data requires a lot of computational time to detect outliers even from small dataset. In the following we propose a cell-based approach of distance-based outlier detection on uncertain data, which can reduce significantly the number of distance functions evaluations. The proposed approach first map database objects to a cell-grid structure and then prunes majority of objects by identifying the cells containing only outliers or non-outliers. We only considered cells within $d + 3\sigma$ distance of the target cell as shown in Fig. 1, for the reason discussed in section 5. 4. For un-pruned objects, Grid File indexing is utilized to further reduce the number of distance function computations.

5. 1 Grid Structure

We assume that our data objects are 2 dimensional. In order to find distance-based outliers on uncertain data, we quantize each object $O_i \in \mathcal{G}DB$, in 2 dimensional space that has been partitioned into cells of length l (cell length is discussed in section 5. 7). Let $C_{x,y}$ be any cell of the Grid, then the neighbouring cells of $C_{x,y}$ form layers around it as shown in Fig. 1. Layers of any cell $C_{x,y}$ in the Grid are defined as follows.

- Layer 1 cells of $C_{x,y}$ are given by

$$L_1(C_{x,y}) = \{C_{u,v} | u = x \pm 1, v = y \pm 1, C_{u,v} \not= C_{x,y}\} \ .$$

- Layer 2 cells of $C_{x,y}$, and are given by

$$L_2(C_{x,y}) = \{C_{u,v} | u = x \pm 2, v = y \pm 2, \\ C_{u,v} \notin L_1(C_{x,y}), C_{u,v} \not= C_{x,y}\} \ .$$

$L_3(C_{x,y}), ..., L_n(C_{x,y})$ are defined in a similar way. Where n denotes the number of cell layers and is discussed in section 5. 4.

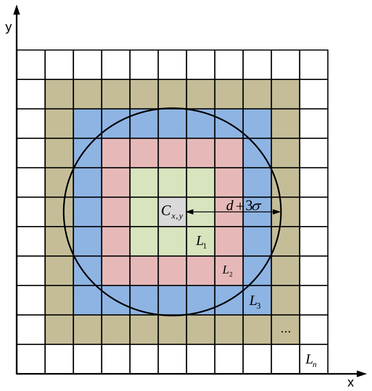


図 1: Cell Layers

5. 2 Cell Layers Bounds and Lookup Table

The bounds of a cell or cell layers are defined for pruning outliers and non-outliers without evaluating the distance functions for the objects in the cell. The upper and lower bounds of any cell or cell layers are shown in Fig.2 and are defined as follows.

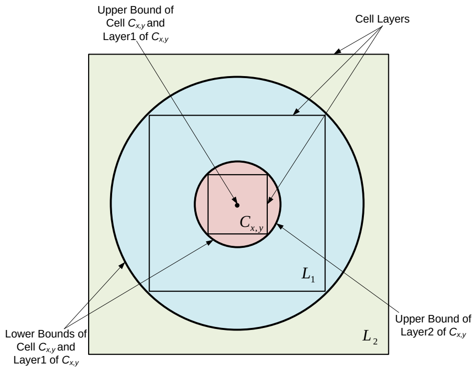

Upper Bound of Cell $C_{x,y}$ and Layer1 of $C_{x,y}$
Cell Layers
$C_{x,y}$
L_1
Lower Bounds of Cell $C_{x,y}$ and Layer1 of $C_{x,y}$
L_2
Upper Bound of Layer2 of $C_{x,y}$

図 2: Cell Bounds

Upper Bound. By an upper bound of a cell (cell layers) we mean the maximum contribution by any of the objects in this cell (cell layers) to the target cell. According to our distance function an object in cell $C_{x,y}$ can contribute at its maximum to object O in cell $C_{x,y}$ when $\alpha_1 = \alpha_2 = 0$ in Eq.3. Similarly the upper bound contributions of objects in $L_i(C_{x,y})$ layers (i.e., $L_1(C_{x,y}), ..., L_n(C_{x,y})$) to objects in $C_{x,y}$ are obtained by setting $\alpha_1 = \alpha_2 = (i-1)\sqrt{2}l$ in Eq.3.

Lower Bound. By a lower bound of a cell (cell layers) we mean the minimum contribution by any of the objects in this cell (cell layers) to the target cell. According to our distance function an object in cell $C_{x,y}$ contributes at its minimum to object O in cell $C_{x,y}$ when $\alpha_1 = \alpha_2 = \sqrt{2}l$ in Eq.3. Similarly, the lower bound contributions of objects in $L_i(C_{x,y})$ layers (i.e., $L_1(C_{x,y}), ..., L_n(C_{x,y})$) to objects in $C_{x,y}$ are obtained by setting $\alpha_1 = \alpha_2 = (i+1)\sqrt{2}l$ in Eq.3.

Lookup Table. The above upper bound and lower bound contributions of objects in $L_i(C_{x,y})$ to $C_{x,y}$ are decided only by the i-value and independent from the locations of $C_{x,y}$. Hence, we compute the bounds and store them in a lookup table to be used in the cell-based algorithm.

5. 3 Pruning of Outliers and Non-outliers Cells

Having defined cell bounds and cell layers bounds, a cell can be pruned as an outlier or non-outlier cell. If the minimum contribution to cell $C_{x,y}$, obtained by the product of cell objects count and cell lower bound is greater than threshold θ, then none of the objects in $C_{x,y}$ could be outliers and we can prune it as non-outliers cell.

$$MinContribution(C_{x,y}) = 1 + (Count(C_{x,y}) - 1) * LowerBound(C_{x,y}) \ .$$

On the other hand if the maximum contribution to cell $C_{x,y}$, obtained by the product of cell objects count and cell upper bound plus the expected contribution by rest of the objects in the database $\mathcal{G}DB$ is less than or equal to θ, then all the objects in $C_{x,y}$ are outliers and we can prune it as outliers cell.

$$MaxContribution(C_{x,y}) = Count(C_{x,y}) + (N - Count(C_{x,y})) * UpperBound(L_1(C_{x,y})) \ .$$

If none of the above conditions hold, then we need to check the contribution of higher cell layers i.e., contributions of $L_1(C_{x,y})$,...,$L_n(C_{x,y})$, until we may either decide the cell $C_{x,y}$ as containing only outliers or only non-outliers or left the cell undecided for the post-pruning evaluation.

5.4 3σ-Filtering

In statistics, 3σ rule states that nearly all(99.73%) of the values in a Gaussian distribution lie within three standard deviation or 3σ of the mean. Therefore according to our definition of distance function, the probability contribution of an object O_j lying greater than or equal to $d + 3\sigma$ distance from object O_i is negligibly small. Hence we can safely restrict the evaluation of distance function to $d + 3\sigma$ distance from object or cell under consideration as shown in Fig.3 and Fig.1 respectively. Hence number of cell layers n to be considered for any cell $C_{x,y}$ can be safely restricted to $n = \lceil \frac{d+3\sigma}{l} \rceil$. This filtering can help reduce the number of distance function evaluation required for an object.

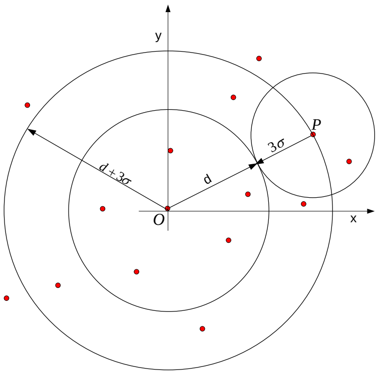


図 3: Object P at $d + 3\sigma$ distance from object O

5.5 Grid File Index

Cell-based pruning may leave some of the cells undecided, i.e., they are neither pruned as non-outliers cells nor as outliers cells. For all the uncertain objects in such cells, we need to follow Nested-loop approach. Our distance function of outlier detection requires a lot of computation time and may reduce the efficiency of our cell-based algorithm even after initial pruning. As we know from our distance function, that it produces higher probability for the nearer objects than the farther objects. We can utilize our Grid structure as Grid-file index [3] with no additional indexing cost to retrieve the nearer objects before the farther objects for the computation of expected value of all un-pruned objects. This will further reduce the number of evaluations required for distance function, hence reducing the overall cost of computation.

5.6 Cell-based Algorithm of Outlier Detection

In order to reduce the costly computation of distance function, we propose cell-based algorithm. The main idea of this algorithm is to prune the cells containing only outliers or non-outliers. Algorithm 2 starts by first calculating the bounds of cell layers and storing them in a look-up table. In line 2, we calculate 3σ layers, that is the number of layers that lie within $d+3\sigma$ distance of any cell $C_{x,y}$. The database objects are then mapped to appropriate cells of the Grid. For each cell, $C_{x,y}$ in Grid, $MinContribution$ and $MaxContribution$ i.e., minimum and maximum contributions are maintained which are used for effectively pruning the cells as outliers or non-outliers. If a cell $C_{x,y}$ can not be pruned, the objects of such cells are checked individually for outliers using Grid-file index.

Although the number of distance function evaluations required in worst case for the cell-based algorithm is same as that of naive approach, i.e., $O(N^2)$ but the experimental results on both synthetic and real datasets show that our proposed approach is very efficient.

5.7 Cell Length *l*

Due to the complexity of our distance function, it is not possible to derive a single cell length l suitable for all the combinations of d and variances. Therefore we conducted several experiments to come up with a cell length which may produce efficient results.

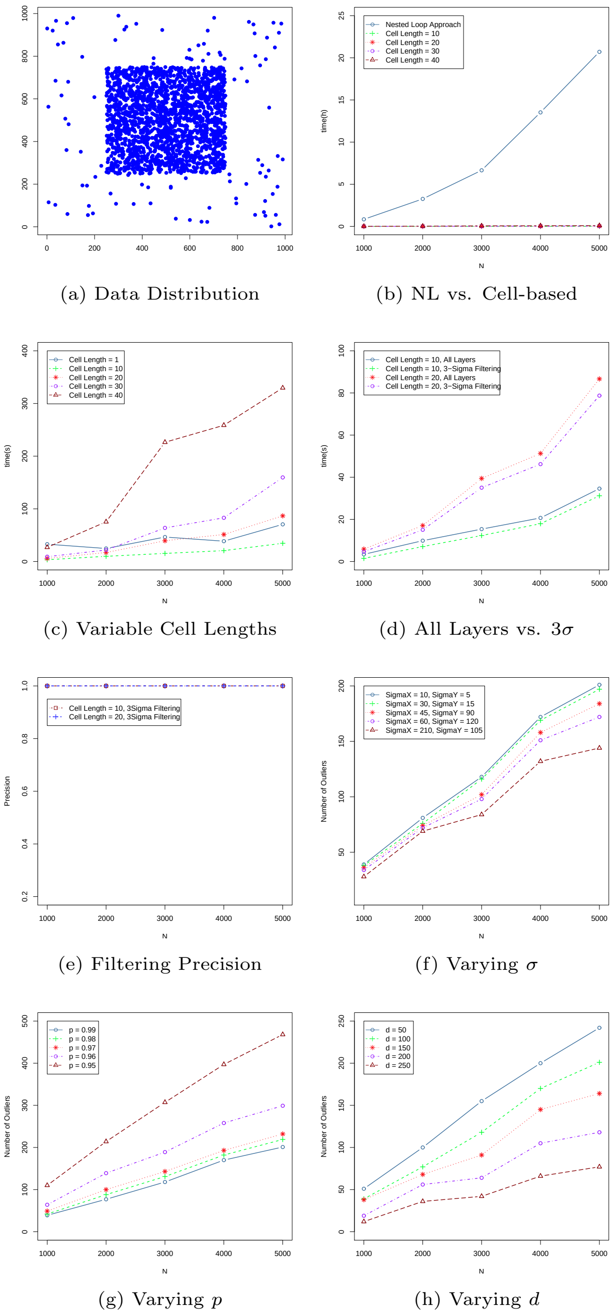
A general observation from several experiments is that smaller the cell-length, shorter the execution time. Since smaller cell-length results in higher values of cell bounds, which helps in pruning majority of objects during cell-based pruning stage and either very few or no cell is left for post-pruning evaluation, reducing the number of distance function evaluations. However very small cell length may also increase the execution time for cell-based algorithm as too small cell length results in a large number of cells and the time required to compute cell layers bounds increases. We need to check a few cell lengths before reaching the appropriate cell-length. A good starting point that we have found through experiments is $l = \frac{\sigma_1 + \ldots + \sigma_k}{k}$.

Algorithm 2 Distance-based Outlier on Uncertain Data: Cell Based Approach

```
Input: database $\mathcal{GDB}$, distance $d$, percentage $p$ , standard deviation $\sigma$
Output: Distance Based Outliers on Uncertain Data of Gaussian Distribution
Compute and store cell bounds into lookup table using cell length $l$ and maximum distance between any two objects in $\mathcal{GDB}$;
/*$n$ denotes the number of cell layers in $d + 3\sigma$ distance*/
$n = \lceil \frac{d+3\sigma}{l} \rceil$;
/*Initialize the count $Count_i$ of each cell $C_i$ in grid $Grid$*/
for each $C_i$ in $Grid$ do
    $Count_i \leftarrow 0$;
end for
/*Mapping database objects to appropriate cells*/
for each $O$ in $\mathcal{GDB}$ do
    map $O$ to an appropriate cell $C_i$;
    $Count_i \leftarrow Count_i$+1;            /*increase cell count by 1*/
end for
$\theta \leftarrow N(1-p)$;            /*calculating the threshold value*/
/*Pruning of outlier and non-outlier cells using cell layers' bounds*/
for each $C_i$ in $Grid$ do
    for $j = 0 \rightarrow n$ do
        Calculate minimum and maximum contribution of cell $C_i$ using upper and lower bounds respectively of 0 to $j^{th}$ neighbouring cell layers of $C_i$;
        if $MinContribution(C_i) > \theta$ then
            prune $C_i$ as non-outlier cell, GOTO Next $C_i$;
        else if $MaxContribution(C_i)$+ expected contribution of $C_i$ from rest of the cell layers in $Grid \leqq \theta$ then
            prune $C_i$ as outlier cell, GOTO Next $C_i$;
        end if
    end for
end for
/*Nested-loop approach using Grid File Index for objects in un-pruned cells*/
for each $C_i$ in $Grid$ do
    if $C_i$ not pruned as outlier or non-outlier cell and $Count_i$ != 0 then
        for each $O$ in $C_i$ do
            $EV_O \leftarrow 0$; /*$EV_O$ is the expected value of object $O$*/
            for each $O_j$ in $C_i$ and higher layers of $C_i$ within $n$ layers do
                $EV_O \leftarrow EV_O + Pr(|\overrightarrow{\mathcal{A}} - \overrightarrow{\mathcal{A}_j}| \leqq d)$;
                if $EV_O > \theta$ then
                    $O$ can not be outlier, GOTO next $O$;
                end if
            end for
            mark $O$ as outlier;
        end for
    end if
end for
```

6. Empirical Study

We conducted extensive experiments on synthetic and real datasets to evaluate the effectiveness and accuracy of our proposed cell-based algorithm. All algorithms were implemented in C#, Microsoft Visual Studio 2008. All experiments were performed on a system with an Intel Core 2 Duo E8600 3.33GHz CPU and 2GB main memory running Windows 7 Professional OS. All programs run in main memory and no I/O cost is considered.



(a) Data Distribution

(b) NL vs. Cell-based

(c) Variable Cell Lengths

(d) All Layers vs. 3σ

(e) Filtering Precision

(f) Varying σ

(g) Varying p

(h) Varying d

図 4: Experiments on synthetic data (default $d = 100$, $p = 0.99$, n =cell layers within $d + 3\sigma$ distance from cell under consideration)

6. 1 Experiments on Synthetic Data

Unless specified, the experiments are performed on 5 uniformly distributed 2-dimensional datasets of sizes varying from 1000 to 5000 objects respectively. Five percent outliers are explicitly introduced in each dataset as shown in Fig. 4a. Uncertainty is simulated by representing each object as Gaussian distributed with means between 0 and 1000 and standard deviation $\sigma_x = 15$ and $\sigma_y = 15$ in x and y dimensions respectively. The default values of different parameters used in experiments are, distance d=100, p=0.99, n = "layers within $d + 3\sigma$ distance of target cell" and l=10.

It is obvious from Fig.4b that the time taken by the Nested-loop algorithm is very high and the execution time increases dramatically as the number of tuples in database increases.

Cell Length l. As discussed in section 5. 7, smaller the cell length, shorter the execution time, which is obvious from Fig. 4c. However, very small cell length may increase the execution time, due to the increase in time required for the computation of look-up table as shown in Fig. 4c for cell length = 1.

3σ-Filtering. From Fig.4d, we can observe that 3σ-filtering is capable of saving some of the computation cost. Moreover, we found that the precision of cell-based algorithm with 3σ-filtering is 100% as shown in Fig. 4e. The reason for this high precision is negligibly small contribution by the cells which are farther than $d + 3\sigma$ from the target cell.

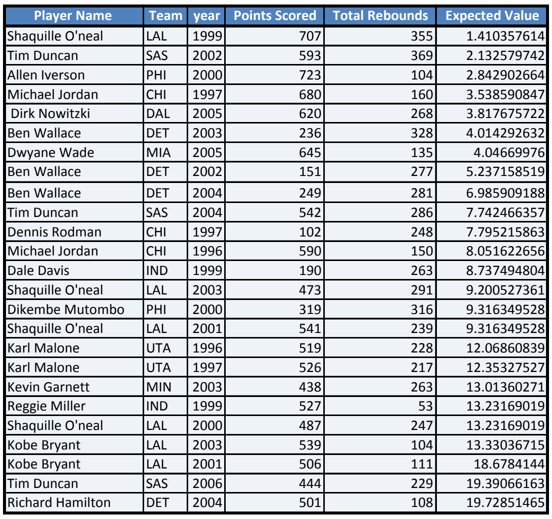
Varying σ. Varying σ values has an effect on both, the number of outliers and the time required for computation. However we have only presented the difference in number of outliers as shown in Fig.4f due to shortage of space. As variance increases, number of outliers decreases. Due to large variance, farther objects start contributing to the target object, hence reducing the number of outliers.

Varying parameters p and d. Varying parameters p and d has an effect on the number of outliers mined by the algorithm as shown by the plots in Fig.4h and Fig.4g respectively. Increasing p results in smaller threshold value, resulting in only a few and relatively stronger outliers.

Varying d has an effect on the distance function probability. Larger d means larger number of objects may fall within d distance of object under consideration, hence increasing the probability support and reducing the number of outliers.

6. 2 Experiments on Real Dataset

For experiments on real dataset, we used NBA Playoffs Player statistics from 1996 to 2006 available at [5]. The dataset contains the annual performance statistics of NBA players. The filtered dataset used in the experiments contain 2081 tuples, with threshold related parameter p set to 0.99. Therefore value of threshold $\theta = N(1 - p) = 20.81$. Each player is represented with 2 important statistics on his performance i.e., number of points and number of total rebounds. Uncertainty is simulated by representing each player as Gaussian distributed with means as the real observed values of his performance statistics and standard deviations $\sigma_{points} = 20$ and $\sigma_{rebounds} = 10$.



Player Name	Team	year	Points Scored	Total Rebounds	Expected Value
Shaquille O'neal	LAL	1999	707	355	1.410357614
Tim Duncan	SAS	2002	593	369	2.132579742
Allen Iverson	PHI	2000	723	104	2.842902664
Michael Jordan	CHI	1997	680	160	3.538590847
Dirk Nowitzki	DAL	2005	620	268	3.817675722
Ben Wallace	DET	2003	236	328	4.014292632
Dwyane Wade	MIA	2005	645	135	4.04669976
Ben Wallace	DET	2002	151	277	5.237158519
Ben Wallace	DET	2004	249	281	6.985909188
Tim Duncan	SAS	2004	542	286	7.742466357
Dennis Rodman	CHI	1997	102	248	7.795215863
Michael Jordan	CHI	1996	590	150	8.051622656
Dale Davis	IND	1999	190	263	8.737494804
Shaquille O'neal	LAL	2003	473	291	9.200527361
Dikembe Mutombo	PHI	2000	319	316	9.316349528
Shaquille O'neal	LAL	2001	541	239	9.316349528
Karl Malone	UTA	1996	519	228	12.06860839
Karl Malone	UTA	1997	526	217	12.35327527
Kevin Garnett	MIN	2003	438	263	13.01360271
Reggie Miller	IND	1999	527	53	13.23169019
Shaquille O'neal	LAL	2000	487	247	13.23169019
Kobe Bryant	LAL	2003	539	104	13.33036715
Kobe Bryant	LAL	2001	506	111	18.6784144
Tim Duncan	SAS	2006	444	229	19.39066163
Richard Hamilton	DET	2004	501	108	19.72851465

図 5: NBA Players with Expected Value less than θ

Our experiments on NBA dataset mined the outstanding players during 1996 and 2006. From the expected values in Fig.5, Shaquille O'neal is the most outstanding player with maximum points scored and second maximum total rebounds. He has the outstanding performance from 1999 to 2003 except the year 2002. On the other hand, Tim Duncan's performance seems to decline during the course of his career as he was a strong outlier in 1999 and became weak outlier in 2006.

7. Conclusion and Future Work

In this paper, we extend the notion of distance-based outlier detection on uncertain data of Gaussian distribution. This is the first approach of distance-based outlier detection where the objects are modelled by Gaussian distribution. We derive distance function for distance-based outlier detection on uncertain data of Gaussian distribution and propose a cell-based algorithm to efficiently detect outliers by pruning majority of outliers and non-outliers cells. We also utilize 3σ-filtering and grid-file index to further reduce the computation time required for the cell-based algorithm. Extensive experiments on synthetic and real data demonstrate the efficiency and scalability of our proposed algorithm.

In future, we are planning to extend this work in two dimensions. First, designing an adaptive algorithm with respect to cell length, in order to increase the efficiency of our cell-based algorithm. Second, expanding this work for general uncertainty model.

8. Acknowledgment

This work has been partly supported by Grant-in-Aid for Scientific Research from JSPS (#2124000).

文献

[1] Salman Ahmed Shaikh and Hiroyuki Kitagawa,: "Distance-based Outlier Detection on Uncertain Data of Gaussian Distribution", The 14th Asia-Pacific Web Conference (AP-Web), 2012. (to appear)

[2] Weisstein, Eric W.:"Normal Difference Distribution", From MathWorld - A Wolfram Web Resource. http://mathworld.wolfram.com/NormalDifferenceDistribution.

[3] J. Nievergelt, H. Hinterberger and K.C. Sevick.:"The Grid File: An Adaptable, Symmetric multikey File Structure", ACM Transaction on Database Systems, 1984.

[4] Edwin M. Knorr , Raymond T. Ng , Vladimir Tucakov.: "Distance-Based Outliers: Algorithms and Applications", The VLDB Journal, 2000.

[5] NBA All-time Player Stats, Opendata by Socrata. http://opendata.socrata.com.

[6] Hawkins D., "Identication of Outliers", Chapman and Hall, 1980.

[7] Barnett V., Lewis T., "Outliers in Statistical Data", John Wiley, 1994.

[8] Gustavo H. Orair , Carlos H. C. Teixeira , Wagner Meira.: "Distance-Based Outlier Detection: Consolidation and Renewed Bearing", Proc. of the VLDB Endowment, 2010.

[9] Bin Wang, Gang Xiao, Hao Yu and Xiaochun Yang.: "Distance-Based Outlier Detection on Uncertain Data", IEEE 9th International Conference on Computer and Information Technology, 2009.

[10] Hans-Peter Kriegel, Peer Krger, Arthur Zimek.: "Outlier Detection Techniques", Tutorial at 16th ACM SIGKDD Conference, 2010.

[11] Sridhar Ramaswamy , Rajeev Rastogi , Kyuseok Shim.: "Efficient Algorithms for Mining Outliers from Large Data Sets", Proceedings International Conference on Management of Data, ACM, SIGMOD, 2000.

[12] Aggarwal, C.C., Yu, P.S.: "Outlier Detection with Uncertain Data", SIAM International Conference on Data Mining, 2008.

[13] Edwin M. Knorr , Raymond T. Ng.: "Algorithms for Mining Distance-Based Outliers in Large Datasets", In Proceedings of 24rd International Conference on Very Large Data Bases, 1998.

[14] M. Mahoney and P. Chan.: "Learning rules for anomaly detection of hostile network traffic", In Proceedings of the Third IEEE International Conference on Data Mining, 2003.

[15] Noor Alaydie, Farshad Fotouhi, Chandan K. Reddy, Hamid Soltanian-Zadeh.: "Noise and Outlier Filtering in Heterogeneous Medical Data Sources", Workshops on Database and Expert Systems Applications, DEXA, 2010.

[16] Arturo Elas, Alberto Ochoa-Zezzatti, Alejandro Padilla and Julio Ponce.: "Outlier Analysis for Plastic Card Fraud Detection a Hybridized and Multi-Objective Approach", Hybrid Artificial Intelligent Systems, Lecture Notes in Computer Science, 2011.

[17] Hugo Garces, Daniel Sbarbaro.: "Outliers detection in environmental monitoring databases", Engineering Applications of Artificial Intelligence, 24(2), 2011.

[18] Maimon O. and Rockach L.: "Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers", Kluwer Academic Publishers, 2005.

[19] Nguyen Hoang Vu, Vivekanand Gopalkrishnan and Ira Assent.: "An Unbiased Distance-Based Outlier Detection Approach for High-Dimensional Data", DASFAA, 2011.

[20] You Wan and Fuling Bian.: "Cell-based outlier detection algorithm: a fast outlier detection algorithm for large datasets", PAKDD, 2008.

[21] Bin Jiang and Jian Pei.: "Outlier Detection on Uncertain Data: Objects, Instances, and Inference", ICDE, 2011.

Appendix:

Proof of Lemma. Let O be a k-dimensional uncertain object with attributes $\overrightarrow{\mathcal{A}} = (x_1, ..., x_k)$, mean $\overrightarrow{\mu} = (\mu_1, ..., \mu_k)^T$ and a diagonal covariance matrix $\Sigma = diag(\sigma_1^2, ..., \sigma_k^2)$. The probability density function of O can be expressed as

$$f(\overrightarrow{\mathcal{A}}) = \frac{1}{\sqrt{(2\pi)^k det\Sigma}} exp\Bigg\{ - \frac{(\overrightarrow{\mathcal{A}} - \overrightarrow{\mu})^T \Sigma^{-1} (\overrightarrow{\mathcal{A}} - \overrightarrow{\mu})}{2} \Bigg\} .$$

Since Σ is diagonal, the distribution functions are independent in coordinates. Hence the k-dimensional normal distribution function is given by the product of k 1-dimensional normal distribution functions.

$$f(\overrightarrow{\mathcal{A}}) = \prod_{1 \leq i \leq k} \frac{1}{\sqrt{2\pi\sigma_i^2}} exp\Bigg\{ - \frac{(x_i - \mu_i)^2}{2\sigma_i^2} \Bigg\} .$$

Let O_i and O_j are two 2-dimensional uncertain objects with attributes $\overrightarrow{\mathcal{A}_i} = (x_{i,1}, x_{i,2})$ and $\overrightarrow{\mathcal{A}_j} = (x_{j,1}, x_{j,2})$, means $\overrightarrow{\mu_i} = (\mu_{i,1}, \mu_{i,2})^T$ and $\overrightarrow{\mu_j} = (\mu_{j,1}, \mu_{j,2})^T$ and diagonal covariance matrices $\Sigma_i = diag(\sigma_{i,1}^2, \sigma_{i,2}^2)$ and $\Sigma_j = diag(\sigma_{j,1}^2, \sigma_{j,2}^2)$ respectively. The difference between normal random vectors of O_i and O_j is given by $\overrightarrow{\mathcal{A}_i} - \overrightarrow{\mathcal{A}_j} \sim \mathcal{N}(\overrightarrow{\mu_{i-j}}, \Sigma_{i-j})$, where $\overrightarrow{\mu_{i-j}} = \mu_i - \mu_j$ and $\Sigma_{i-j} = \Sigma_i + \Sigma_j$ [2].

Since Σ_i and Σ_j are diagonal matrices, the distribution functions are independent in coordinates. Hence the 2-dimensional normal difference distribution of uncertain objects O_i and O_j is given by,

$$\begin{aligned} f(\overrightarrow{\mathcal{A}_i} - \overrightarrow{\mathcal{A}_j}) = & \frac{1}{2\pi\sqrt{(\sigma_{i,1}^2 + \sigma_{j,1}^2)(\sigma_{i,2}^2 + \sigma_{j,2}^2))}} \\ & exp\Bigg\{ - \Bigg(\frac{(x - \alpha_1)^2}{(\sigma_{i,1}^2 + \sigma_{j,1}^2)} + \frac{(y - \alpha_2)^2}{(\sigma_{i,2}^2 + \sigma_{j,2}^2)} \Bigg) \Bigg\} , \end{aligned} \tag{5}$$

where $\alpha_1 = \mu_{i,1} - \mu_{j,1}$ and $\alpha_2 = \mu_{i,2} - \mu_{j,2}$ are the differences between the means of objects O_i and O_j and $\sigma_{i,1}^2$, $\sigma_{j,1}^2$, $\sigma_{i,2}^2$ and $\sigma_{j,2}^2$ are the variances of the uncertain objects O_i and O_j in dimensions 1 and 2 respectively.

Hence the probability that the uncertain object O_i lies within d-distance of uncertain object O_j is given by,

$$\begin{aligned} Pr(|\overrightarrow{\mathcal{A}_i} - \overrightarrow{\mathcal{A}_j}| \leq d) = & \frac{1}{2\pi\sqrt{(\sigma_{i,1}^2 + \sigma_{j,1}^2)(\sigma_{i,2}^2 + \sigma_{j,2}^2)}} \int_0^d \int_0^{2\pi} \\ \exp\Bigg\{ - \Bigg(& \frac{(r\cos\theta - \alpha_1)^2}{2(\sigma_{i,1}^2 + \sigma_{j,1}^2)} + \frac{(r\sin\theta - \alpha_2)^2}{2(\sigma_{i,2}^2 + \sigma_{j,2}^2)} \Bigg) \Bigg\} r \, d\theta \, dr \ \blacksquare \end{aligned} \tag{6}$$