Distance-Based Outlier Detection on Uncertain Data of Gaussian Distribution

Shaikh Salman Ahmed and Hiroyuki Kitagawa

Graduate School of Systems and Information Engineering
University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8573, Japan
salman@kde.cs.tsukuba.ac.jp, kitagawa@cs.tsukuba.ac.jp
https://www.kde.cs.tsukuba.ac.jp/

Abstract. Managing and mining uncertain data is becoming important with the increase in the use of devices responsible for generating uncertain data, for example sensors, RFIDs, etc. In this paper, we extend the notion of distance-based outliers for uncertain data. To the best of our knowledge, this is the first work on distance-based outlier detection on uncertain data of Gaussian distribution. Since the distance function for Gaussian distributed objects is very costly to compute, we propose a cell-based approach to accelerate the computation. Experimental evaluations of both synthetic and real data demonstrate effectiveness of our proposed approach.

Keywords: Outlier Detection, Uncertain Data, Cell Based Approach.

1 Introduction

Outlier detection is one of the most important data mining techniques with a vital importance in many application domains including credit card fraud detection, network intrusion detection, environmental monitoring, medical sciences, etc. Although there exists no any universally agreed upon definition of outliers, some definitions are general enough to give a basic idea of outliers. Hawkins [5] 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 [6] 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 [6], 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 [3], [7], [8].

Q.Z. Sheng et al. (Eds.): APWeb 2012, LNCS 7235, pp. 109–121, 2012.

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 our work, uncertainty of data is modelled by the most commonly used PDF i.e., Gaussian distribution. We have 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. Finally we make use of grid structure to further reduce the computation time required for distance-based outlier detection.

The rest of the paper is organized as follows. In section 2, the related work is discussed. Section 3 gives the derivation of distance function and 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 [9].

Distance-based outliers detection approach on deterministic data was introduced by Knorr et.al. in [3]. 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. [10] 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. Recently in [7], 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.

Recently a lot of research has focused on managing, querying and mining of uncertain data [11], [8], 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 [11]. 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 [8], 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. 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 [3] 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 $\Sigma_i = diag(\sigma_{i,1}^2, ..., \sigma_{i,k}^2)$ 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,

$$\mathcal{UDB}Outliers = \{O_i \in \mathcal{GDB} | \sum_{j=1}^{|\mathcal{GDB}|} Pr(|\overrightarrow{\mathcal{A}_i} - \overrightarrow{\mathcal{A}_j}| \leq d) \leq \theta\} \ . \qquad (1)$$

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 $\sigma_{i-j}^2 = \sigma_i^2 + \sigma_j^2$ [1]. Hence we can write $\overrightarrow{\mathcal{A}_i} - \overrightarrow{\mathcal{A}_j} \sim \mathcal{N}(\mu_{i-j}, \sigma_{i-j}^2)$.

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,

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

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

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,*

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

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 $\Sigma_j = diag(\sigma^2_{j,1},...,\sigma^2_{j,k})$ respectively. The probability that the uncertain object O_i lies within d distance of uncertain object O_j is given by,

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

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.

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 $O_i \in \mathcal{G}DB$ and every other object in the $\mathcal{G}DB$ 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.

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

```
Input: database $\mathcal{G}DB$, distance $d$, percentage $p$ , standard deviation $\sigma$
Output: Uncertain Distance Based Outliers
$N \leftarrow$ number of objects in $\mathcal{G}DB$;
$\theta \leftarrow N(1-p)$;                    /*calculating the threshold value*/
for each $O$ in $\mathcal{G}DB$ do
    $EV_O \leftarrow 0$;            /*$EV_O$ denotes the expected value of object $O$*/

    for each $O_i$ in $\mathcal{G}DB$ do
        $EV_O = EV_O + Pr(|\overrightarrow{\mathcal{A}} - \overrightarrow{\mathcal{A}_i}| \leq d)$;
        if $EV_O > \theta$ then
            mark $O$ as non-outlier, GOTO next $O$;
        end if
    end for
    mark $O$ as outlier;
end for
```

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. 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 5.6). 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

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

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

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

$L_3(C_{x,y}), ..., L_n(C_{x,y})$ are defined in a similar way. Where n denotes the number of cell layers and is discussed in section 5.6.

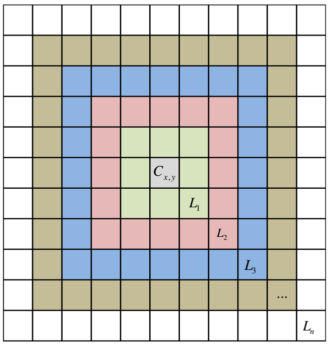


Fig. 1. Cell Layers

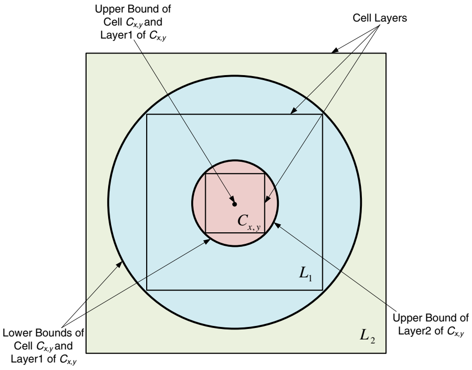

Upper Bound of Cell $C_{x,y}$ and Layer1 of $C_{x,y}$
Cell Layers
$C_{x,y}$
L_1
Upper Bound of Layer2 of $C_{x,y}$
Lower Bounds of Cell $C_{x,y}$ and Layer1 of $C_{x,y}$
L_2

Fig. 2. Cell Bounds

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. 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 $\alpha_x = \alpha_y = 0$ 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 $\alpha_x = \alpha_y = (i-1)\sqrt{2}l$ 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 $\alpha_x = \alpha_y = \sqrt{2}l$ 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 $\alpha_x = \alpha_y = (i+1)\sqrt{2}l$ in Eq.3.

Lookup Table. The above upper bound and lower bound of 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.

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

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 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 [2] 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.

5.5 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. 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.

5.6 Cell Length l and Cell Layers n

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 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 $l = \frac{\sigma_1 + \ldots + \sigma_k}{k}$.

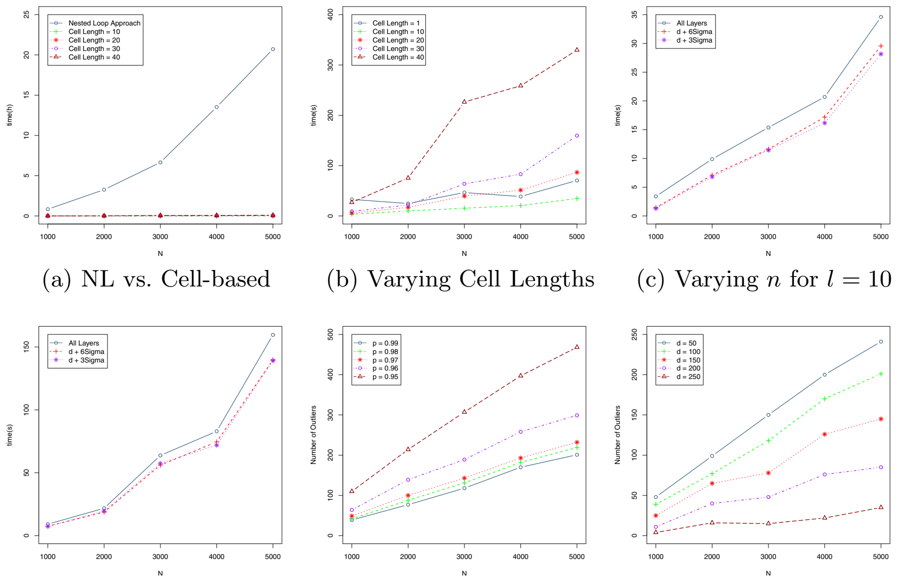
Cell Layers n. Since a Gaussian function decays exponentially with respect to the distance to its mean, the density contribution is small if the mean is far away from the target object. Using this fact, we conducted experiments where n was set to, 1) all layers in the Grid, 2) layers within $d + 6\sigma$ distance of target cells and 3) layers within $d + 3\sigma$ distance of target cells. We found that all three experiments retrieved same number of outliers and as expected, n with layers within $d + 3\sigma$ was faster than the other two. The reason for the same number of outliers in all three choices of n is that the contribution of the cells which are farther than $d + 3\sigma$ is negligibly small and has no effect on outlier detection.

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

```
Input: database $\mathcal{G}DB$, 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 $\mathcal{G}DB$;
/*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 $\mathcal{G}DB$ 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*/
$n = \lceil \frac{max(|\overrightarrow{\mathcal{A}_p} - \overrightarrow{\mathcal{A}_q}|)}{l} \rceil$, where $O_p, O_q \in \mathcal{G}DB$;/*$n$ denotes the number of cell layers*/
/*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 $Grid \leq \theta$ 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
      $EV_O \leftarrow 0$;        /*$EV_O$ denotes the expected value of object $O$*/
      for each $O_j$ in $C_i$ and higher layers of $C_i$ in $Grid$ do
        $EV_O \leftarrow EV_O + Pr(|\overrightarrow{\mathcal{A}} - \overrightarrow{\mathcal{A}_j}| \leq d)$;
        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) NL vs. Cell-based (b) Varying Cell Lengths (c) Varying n for $l = 10$

(d) Varying n for $l = 30$ (e) Varying p (f) Varying d

Fig. 3. Experiments on synthetic data (default $d = 100$, $p = 0.99$, n =layers within $d + 3\sigma$ and $l = 10$)

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 tuples respectively with parameters, distance d=100, p=0.99, n = "layers within $d+3\sigma$ distance of target cell"and l=10. 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.

It is obvious from Fig.3a 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.

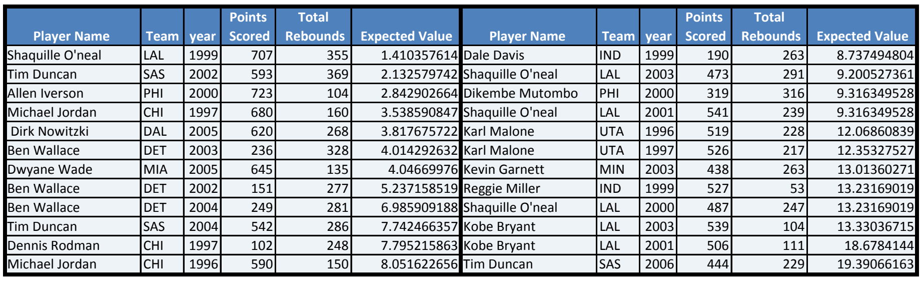
Cell Length *l* and Cell Layers *n*. As discussed in section 5.6, smaller the cell length, shorter the execution time, which is obvious from Fig. 3b. However, very small cell length may increase the execution time, due to the increase in time required for the computation of look-up table as shown in Fig. 3b for cell length = 1. From Fig.3c and Fig.3d, we can observe that cell layers within $d+3\sigma$ distance of target cell produces better results than all the layers in the Grid.

Varying Parameters p and d. Varying parameters p and d has an effect on number of outliers mined by the algorithm as shown by the plots in Fig.3e and Fig.3f 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 contribution. Larger d means larger number of objects may fall within d distance of object under consideration, hence increasing the expected value 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 [4]. 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 as an uncertain object with 2 important statistics on his performance i.e., number of points and number of total rebounds. Both statistics are defined by means as the real observed values and standard deviations $\sigma_{points} = 20$ and $\sigma_{rebounds} = 10$.



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

Fig. 4. 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.4, 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 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 uncertain data streams of Gaussian distribution.

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

References

1. Weisstein, E.W.: Normal Difference Distribution. From MathWorld - A Wolfram Web Resource, `http://www.mathworld.wolfram.com/NormalDifferenceDistribution.html`
2. Nievergelt, J., Hinterberger, H., Sevick, K.C.: The Grid File: An Adaptable, Symmetric multikey File Structure. ACM Transaction on Database Systems (1984)
3. Knorr, E.M., Ng, R.T., Tucakov, V.: Distance-Based Outliers: Algorithms and Applications. The VLDB Journal 8, 237–253 (2000)
4. NBA All-time Player Stats, Opendata by Socrata, `http://opendata.socrata.com`
5. Hawkins, D.: Identification of Outliers. Chapman and Hall (1980)
6. Barnett, V., Lewis, T.: Outliers in Statistical Data. John Wiley (1994)
7. Orair, G.H., Teixeira, C.H.C., Meira, W.: Distance-Based Outlier Detection: Consolidation and Renewed Bearing. Proc. of the VLDB Endowment (2010)
8. Wang, B., Xiao, G., Yu, H., Yang, X.: Distance-Based Outlier Detection on Uncertain Data. In: IEEE 9th International Conference on Computer and Information Technology (2009)
9. Kriegel, H.-P., Kröger, P., Zimek, A.: Outlier Detection Techniques. Tutorial at 16th ACM SIGKDD Conference (2010)
10. Ramaswamy, S., Rastogi, R., Shim, K.: Efficient Algorithms for Mining Outliers from Large Data Sets. In: Proceedings International Conference on Management of Data. ACM, SIGMOD (2000)
11. Aggarwal, C.C., Yu, P.S.: Outlier Detection with Uncertain Data. In: SIAM International Conference on Data Mining (2008)

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

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

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.

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

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 $\Sigma_{i-j} = \Sigma_i + \Sigma_j$ [1].

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,

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

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,

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

■