Query Processing over Probabilistic Data with Gaussian Distributions
ガウス分布に基づく確率的データに対する問合せ処理

Tingting Dong ♥

董　テイテイ

The field of uncertain data management has received extensive attention of researchers due to the increasing demand for managing uncertain data in a large variety of real-world applications such as sensor networks, location-based services, monitoring and surveillance. In this thesis, we model uncertainty probabilistically and represent each uncertain object in the database using a Gaussian distribution, which is a typical probability distribution widely used in statistics, pattern recognition, and machine learning. We consider the following three types of queries or searches over probabilistic data with Gaussian distributions: 1) probabilistic range query, 2) nearest neighbor search, and 3) similarity search.

1. Introduction

In recent years, the rapid advancement in data acquiring devices has led to the increasing availability of massive data in a wide variety of fields such as information retrieval, data integration, mobile robotics, sensor networks, location-based services, monitoring and surveillance. One common characteristic of the data acquired in these fields is their uncertain nature. In some cases, it is possible to eliminate the uncertainties completely. However, this is usually very costly, like manual removal of ambiguous matches in data cleaning. In many cases, complete removal is not even possible. Uncertain data has no place in traditional, precise database management systems like inventory and employee. This has created a need for uncertain data management.

Following this trend, the database and data mining community has investigated extensively the problem of modeling and querying uncertain data [1, 2] in recent decades. Hundreds of research work has been done on this topic [3, 4] and several probabilistic database management systems have been proposed [5–7]. The general approach to manage uncertain data is with a probabilistic model [8], which represents uncertainties using probabilities. In this thesis, we focus on the continuous attribute uncertainty and the case where the uncertainty is represented by a Gaussian distribution. In other words, we assume that uncertain objects stored in the database are represented by Gaussian distributions. Under this setting, we consider query processing over probabilistic data with Gaussian distributions.

We focus our effort on Gaussian distribution because it is one of the most typical probability distributions, and is widely used in statistics, pattern recognition, and machine learning [9, 10]. Especially, in the area of spatio-temporal databases, it is frequently used to represent uncertain location information [8, 11–13].

In Fig.1 and Fig.2, we show examples of one-dimensional and two-dimensional Gaussian distribution, respectively. In the multi-dimensional space, Gaussian distribution is represented by

$$p(x) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} \exp\left[-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right].$$

Here, μ denotes the average vector and Σ denotes the covariance matrix. $|\Sigma|$ and Σ^{-1} represent the determinant and inverse matrix of Σ, respectively. $(x-\mu)^T$ represents the transposition of $(x-\mu)$. For simplicity, we call objects represented by Gaussian distributions Gaussian objects afterwards.

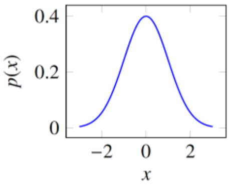

Fig.1: One-dimensional Gaussian distribution

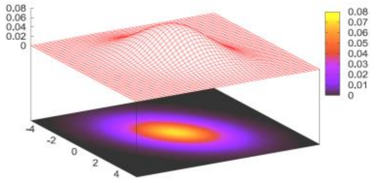

Fig.2: Two-dimensional Gaussian distribution

Generally speaking, it describes the probability that a random point distributes in the space. For example, from Fig.1 in the one-dimensional space, the probability of a point being around the center 0 is about 0.4. The probability decreases as the point spreads from the center. This is the similar in the two-dimensional space in Fig.2. We project the probability surface to a plane and show the decreasing trend with gradient colors. When using Gaussian distribution to represent uncertain location information, it means that the object has the highest probability to be located in the reported location (i.e., the center) and the farther it is, the smaller the probability to be located there becomes.

In the first work, we consider range query, one of the most important queries over spatio-temporal databases. A range query retrieves all the objects that are within the given search range. A range query over uncertain objects, called a probabilistic range query [13, 14], searches for objects within the given search range with probabilities no less than a specified probability threshold. The query object can be either a certain point or an uncertain object represented by a Gaussian distribution. In location-based services, when representing the locations of landmarks by probability distributions like Gaussian distributions, this kind of query can be used to find surrounding landmarks for a self-navigated mobile robot when building the environment map.

Our second work studies nearest neighbor query or nearest neighbor search, which is also a common type of query in spatio-temporal databases. This query returns the nearest objects to a query point. When extending

♥Student member, Graduate School of Information Science, Nagoya University, dongtt@db.ss.is.nagoya-u.ac.jp traditional certain objects to uncertain objects, there are two variations for the new query. The first one, called probabilistic nearest neighbor query [15], utilizes a probability threshold to define qualifying objects in a similar way as the range query. The second one, called expected nearest neighbor search, identifies objects based on their expected distances [16, 17] to the query point. We focus on the second one and consider k-expected nearest neighbor search over uncertain objects represented by Gaussian distributions. An application example is to find potential customers nearby for shops or restaurants.

The third work explores similarity search over Gaussian distributions. Given a database of Gaussian distributions and a query Gaussian distribution, we search the database for top-k similar data Gaussian distributions. We utilize the Kullback-Leibler divergence (KL-divergence) to measure the similarity between two Gaussian distributions. As in [18], we can represent feature vectors generated in pattern recognition and machine learning using Gaussian distributions and conduct similarity search over them to make interesting and useful findings.

2. Probabilistic Range Querying over Gaussian Objects

A probabilistic range query returns all the data objects that appear within the given query region (QR) with probabilities no less than a given probability threshold θ. For instance, consider a self-navigated mobile robot moving in a wireless environment. The robot builds a map of the environment by observing nearby landmarks via devices such as sonars and laser range finders. Due to the inherent limitation brought about by sensor accuracy and signal noises, the location information acquired from measuring devices is not always precise. At the same time, the robot also conducts probabilistic localization to estimate its own location autonomously by integrating its movement history and the landmark information. This may cause impreciseness in the location of the robot, too. In consequence, probabilistic queries have evolved to tackle such impreciseness as "find landmarks that are located within 5 meters from my current location with probabilities at least 80%".

In this work we study the case where the locations of data objects are uncertain, whereas the location of the query object is either exact or uncertain. Specifically, data objects are described by Gaussian distributions with different parameters to indicate their respective uncertainty. A query object can be either a certain point in the multi-dimensional space or an uncertain location represented by a multi-dimensional Gaussian distribution. We solve the probabilistic range query problem according to the above setup.

A straightforward approach to this problem is to compute the appearance probability for each data object and output it if this probability is no less than the threshold. However, the probability computation usually requires costly numerical integration for the accurate result, rendering it prohibitively expensive to compute for all the data objects and check if the query constraint is satisfied. Thus, such computations should be reduced as much as possible.

To this end, we propose filtering techniques to generate a set of candidate data objects and compute integrations only for these candidates. Equipped with the filtering techniques, an R-tree-based indexing method is proposed to accelerate query processing. The index structure is inspired by the idea of TPR-tree [19], in which the minimum bounding boxes (MBBs) vary with time. The difference is that in our index a parent MBB not only varies with the probability threshold but also tightly encloses all child MBBs.

Our filtering techniques and index structure are both based on the probabilistic region, called ρ-region, of Gaussian distribution as shown in Fig.3. Inside this ellipsoidal region, the integrated probability of Gaussian distribution is exactly ρ. The ellipsoidal shape of the ρ-region renders it difficult to quickly examine whether the ρ-region intersects QR as well as develop an indexing scheme based on prevalent spatial indexes such as R-tree. Hence, we study the MBB which tightly bounds the ρ-region. The length of the edge of the MBB in each dimension, i.e. w_j or w_k, is determined by ρ and the standard deviation.

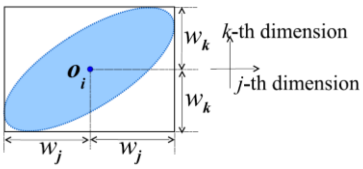

Fig.3: MBB of ρ-region

During query processing, we filter Gaussian objects based on their MBBs. For example, for a Gaussian object if the MBB of its θ-region is included in QR, the integrated appearance probability of this object inside QR is definitely greater than θ. To process queries more efficiently, we index MBBs of all the data objects in the database. The proposed index structure is dynamic and can be determined immediately upon the query. Thus, we do not need to reconstruct it for each query. Extensive experiments on real datasets demonstrate the efficiency and effectiveness of our proposed approach.

3. k-Expected Nearest Neighbor Search over Gaussian Objects

As one of the commonest queries over location information, the distance-based nearest neighbor search has many applications in various fields such as databases, pattern recognition, recommendation systems, and cluster analysis. Given a query point q, this kind of query searches the database for top-k objects that are closest to q. There have been considerable efforts made to extend nearest neighbor search over traditional location information to uncertain location information. One representative example is the expected distance [16, 17], which defines the distance over uncertain location information. Following this trend, in this work, we represent uncertain location information by Gaussian distributions and assume that the closeness between each Gaussian object and the query point is measured by their expected squared Euclidean distance [17], which is frequently used in many areas such as pattern mining and cluster analysis. Under this setting, we consider the problem of k-expected nearest neighbor search over Gaussian objects.

As shown in Fig.4, an application example is to find nearby mobile users to a given query location such as a shop or a restaurant. Since the location of a mobile user is uncertain, it is a common practice to represent uncertain locations using Gaussian distributions in the area of spatial databases [14]. The restaurant or shop may consider sending coupons or notifications to a number of nearest users for promotion.

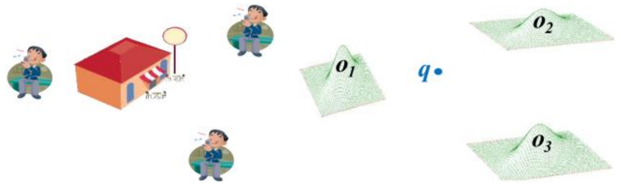

Fig.4: An application example

The expected distance used in this work, called *ESQED*, is defined using an integral as follows:

$$ESQED(o,q) = \int \|x-q\|^2 \cdot p(x)dx.$$

This distance is normally computed by numerical integration such as the Monte Carlo methods. This kind of methods requires a sufficiently large number of samples (e.g., 10^5) to ensure accuracy. Hence, it will lead to very high time cost if we process queries by comparing distances of all objects one by one in real time. This naïve solution is intolerable for the vast majority of real-world applications which demand immediate responses. What is more, the world of today is being flooded with streams of large data and is in the age of big data. This calls for novel approaches that can handle a large dataset and support efficient query processing.

To find an efficient solution, we analyze properties of expected distance on Gaussian distribution mathematically and derive the lower bound and upper bound of this distance. Meanwhile, we employ the filter-and-refine paradigm to accelerate query processing. By filtering, we prune unpromising objects whose lower bound distances are larger than upper bound or expected distances of candidate objects without computing their actual distances. In refinement, we compute exact distances of candidate objects and finally return the top-k smallest ones. To further improve the performance, we utilize R-tree to index objects and their lower bound distances and upper bound distances. We propose three novel algorithms to support efficient query processing. The experimental result demonstrates that our proposed approaches can achieve great efficiency and are applicable to real-world applications.

4. Top-*k* Similarity Search over Gaussian Distributions Based on KL-Divergence

In this work, we study the problem of processing similarity search queries over objects represented by Gaussian distributions. As shown in Fig.5, given a database of Gaussian distributions D and a query Gaussian distribution q, our objective is to find top-k Gaussian distributions from D that are similar to q. We assume that a large number of objects represented by non-correlated Gaussian distributions are stored in the database. By non-correlated Gaussian distribution, we mean that all dimensions are independent with each other, i.e., the covariance matrix of each Gaussian distribution is diagonal. In this work, we focus on non-correlated Gaussian distributions since they are frequently used in machine learning and statistics. Hereafter, we use the term Gaussian distributions for non-correlated ones. Given a query Gaussian distribution, our task is to retrieve from the database the Gaussian distributions that are similar to the query. The top-k Gaussian distributions with the highest similarity scores are returned as the answer to the query.

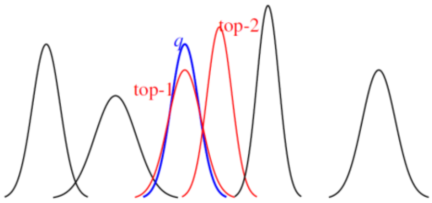

Fig.5: A one-dimensional query example (k = 2)

To capture the similarity between a data Gaussian distribution and a query Gaussian distribution, we choose Kullback-Leibler divergence (KL-divergence), which is a representative measure for quantifying the similarity between two probability distributions. Given two continuous probability distributions $p_1(x)$ and $p_2(x)$, the KL-divergence between them is

$$D_{KL}(p_1 \,\|\, p_2) = \int p_1(x) \ln \frac{p_1(x)}{p_2(x)} dx.$$

In information theory, KL-divergence $D_{KL}(p_1\|p_2)$ is interpreted as a measure of the inefficiency of assuming that the distribution is p_2 when the true distribution is p_1. In other words, it measures the information lost when p_2 is used to approximate p_1. The smaller the KL-divergence is, the more similar the two probability distributions are. Accordingly, the problem of KL-divergence-based top-k similarity search over Gaussian distributions is actually equivalent to finding top-k Gaussian objects having the smallest KL-divergences with the query Gaussian object.

KL-divergence is introduced in [20], and has then become the commonest divergence measures used in practice. It is well-known that KL-divergence is a non-metric measure which violates the properties of a standard distance function in metric spaces such as the Euclidean space with the Euclidean distance. Specifically, it is asymmetric and does not satisfy the triangular inequality. Hence, existing index structures based on distance functions for metric spaces like M-tree [21] cannot be employed to solve this problem.

As KL-divergence is asymmetric, given a data Gaussian object p and a query Gaussian object q, there are two options when using it as the similarity measure between them: $D_{KL}(p\|q)$ or $D_{KL}(q\|p)$. It is not easy to decide which one to use, and may vary according to different applications. Both of them are common in the literature. Thus, in this work, we study both types and analyze their mathematical properties.

A naïve solution is to sequentially compute the KL-divergence with the query Gaussian distribution for each Gaussian distribution in the database, and select the ones with top-k smallest KL-divergences. However, this method poses remarkable computing overhead and hence is not scalable to large datasets. In consequence, we employ the filter-and-refine paradigm to improve the efficiency of query processing. It first generates a set of promising candidates and filters unpromising ones without computing their similarities, and then candidate objects are refined to obtain the final results.

Based on our analysis, we propose two types of approaches utilizing the notions of rank aggregation [22] and skyline queries [23]. The first type presorts all objects in the database on their attributes and computes result objects by merging candidates from presorted lists. We modify the representative threshold algorithm (TA) [22] and propose two algorithms for efficient query processing. The second one transforms the problem to the computation of dynamic skyline queries [24]. We extend and modify the branch-and-bound skyline (BBS) algorithm [24], which is proposed to answer skyline queries, and develop a novel algorithm to solve this problem. We demonstrated the efficiency and effectiveness of our approaches through a comprehensive experimental performance study.

5. Conclusions

In this thesis, we studied uncertain data management due to the increasing uncertain data in a wide variety of fields in recent decades. We focused our attention on the case where the uncertainty is represented by a Gaussian distribution. We considered three types of queries or searches over probabilistic data with Gaussian distributions: 1) probabilistic range query, 2) nearest neighbor search, and 3) similarity search.

Answering queries over uncertain databases poses a number of challenges since managing uncertainty usually means costly probability computations. Hence, we developed efficient solutions for these three problems and conducted comprehensive performance studies using both synthetic and real datasets.

[References]

[1] C.C. Aggarwal. *Managing and Mining Uncertain Data*. Springer, 2009.

[2] D. Suciu, D. Olteanu, C. Ré, and C. Koch. *Probabilistic Databases*. Morgan & Claypool Publishers, 2011.

[3] C.C. Aggarwal and P.S. Yu. A survey of uncertain data algorithms and applications. *IEEE TKDE*, 21(5), pp.609–623, 2009.

[4] Y. Wang, X. Li, X. Li, and Y. Wang. A survey of queries over uncertain data. *Knowledge and Information Systems*, 37(3), pp.485–530, 2013.

[5] P. Agrawal, O. Benjelloun, A.D. Sarma, C. Hayworth, S.U. Nabar, T. Sugihara, and J. Widom. Trio: A system for data, uncertainty, and lineage. In *VLDB*, pp.1151–1154, 2006.

[6] L. Antova, C. Koch, and D. Olteanu. Query language support for incomplete information in the maybms system. In *VLDB*, pp.1422–1425, 2007.

[7] S. Singh, C. Mayfield, S. Mittal, S. Prabhakar, S.E. Hambrusch, and R. Shah. Orion 2.0: native support for uncertain data. In *ACM SIGMOD*, pp.1239–1242, 2008.

[8] Y. Tao, X. Xiao, and R. Cheng. Range search on multidimensional uncertain data. *ACM TODS*, 32(3), pp.15:1–15:54, 2007.

[9] C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.

[10] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern Classification*. Wiley-Interscience, 2nd edition, 2000.

[11] P.K. Agarwal, S.W. Cheng, and K. Yi. Range searching on uncertain data. *ACM Trans. Algorithms*, 8(4), pp.43:1–43:17, 2012.

[12] K. Patroumpas, M. Papamichalis, and T.K. Sellis. Probabilistic range monitoring of streaming uncertain positions in geosocial networks. In *SSDBM*, pp.20–37, 2012.

[13] Y. Ishikawa, Y. Iijima, and J.X. Yu. Spatial range querying for Gaussian-based imprecise query objects. In *ICDE*, pp.676–687, 2009.

[14] T. Dong, C. Xiao, and Y. Ishikawa. Probabilistic range querying over Gaussian objects. *The Institute of Electronics, Information and Communication Engineers Transactions*, 97-D(4), pp.694–704, 2014.

[15] H.P. Kriegel, P. Kunath, and M. Renz. Probabilistic nearest neighbor query on uncertain objects. In *DASFAA*, pp.337–348, 2007.

[16] V. Ljosa and A.K. Singh. APLA: Indexing arbitrary probability distributions. In *ICDE*, pp.946–955, 2007.

[17] P.K. Agarwal, A. Efrat, S. Sankararaman, and W. Zhang. Nearest-neighbor searching under uncertainty. In *ACM PODS*, pp.225–236, 2012.

[18] C. Böhm, A. Pryakhin, and M. Schubert. The Gauss-tree: Efficient object identification in databases of probabilistic feature vectors. In *ICDE*, 2006.

[19] S. Šaltenis, C.S. Jensen, S.T. Leutenegger, and M.A. Lopez. Indexing the positions of continuously moving objects. In *ACM SIGMOD*, pp.331–342, 2000.

[20] S. Kullback and R.A. Leibler. On information and sufficiency. *Ann. Math. Statist.*,22(1), pp.79–86, 1951.

[21] P. Ciaccia, M. Patella, and P. Zezula. M-tree: An efficient access method for similarity search in metric spaces. In *VLDB*, pp.426–435, 1997.

[22] R. Fagin, A. Lotem, and M. Naor. Optimal aggregation algorithms for middleware. In *ACM PODS*, pp.102–113, 2001.

[23] S. Börzsönyi, D. Kossmann, and K. Stocker. The skyline operator. In *ICDE*, pages 421–430, 2001.

[24] D. Papadias, Y. Tao, G. Fu, and B. Seeger. Progressive skyline computation in database systems. *ACM TODS*, 30(1), pp.41–82, 2005.