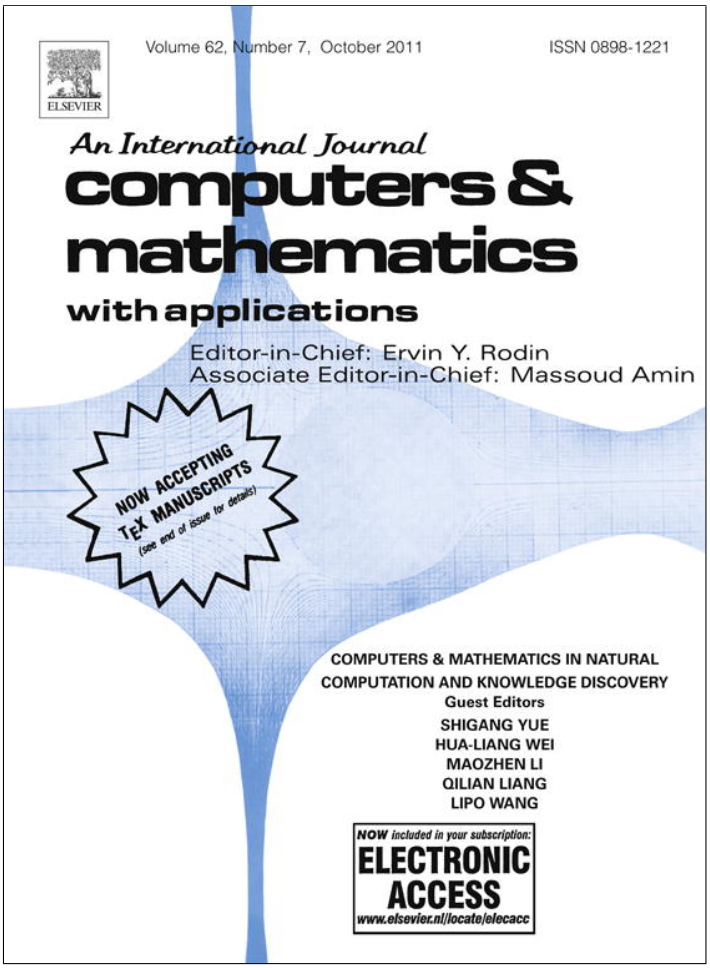

Computers and Mathematics with Applications 62 (2011) 2812–2823

Contents lists available at SciVerse ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

On the semantics of top-k ranking for objects with uncertain data☆

Chonghai Wang, Li Yan Yuan, Jia-Huai You*

Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada T6G 2E8

ARTICLE INFO

Keywords:
Top-k ranking
Uncertain data
Probabilistic database
Constraint
High-dimensional space
Possible world

ABSTRACT

The goal of top-k ranking for objects is to rank the objects so that the best k of them can be determined. In this paper we consider an object to be an entity which consists of a number of attributes whose roles in the object are determined by an aggregation function. The problem of top-ranking in this case is conceptually simple for data that are complete and certain — the aggregation value of an object represents its strength and therefore its rank. For uncertain data, the semantic basis of top-k objects becomes unclear. In this paper, we formulate a semantics of top-k ranking for objects modeled by uncertain data, where the values of an object's attributes are expressed by probability distributions and constrained by some stated conditions. Under this setting, we present a theory of top-k ranking for objects so that their strengths can be determined in the presence of uncertain data. We present our theory in three stages. The first deals with discrete domains, which is extended to include continuous domains. We show that top-k ranking for objects in this context is closely related to high-dimensional space studied in mathematics. In particular, the computation of the volumes of a high-dimensional polyhedron represented by a system of linear inequations is a special case of top-k ranking under our theory. We further extend this theory to add weights to objects' positions and aggregation values in determining ranking results. We show that a number of previous proposals for top-k ranking are special cases of our theory.

1. Introduction

The general problem of top-k ranking is to rank individuals so that the best k of them can be determined. The problem has wide commercial and social implications. For example, we cast our votes to elect our representatives in parliament from a group of candidates, and we rank products of a particular kind, e.g., cars, based on various factors. In general, individuals can be anything on which an ordering makes sense. They can be (concrete or abstract) objects, events, or tuples in a database; e.g., patients waiting for treatment, leads in a criminal case investigation, performance in a sport or artistic competition, and popularity of politicians/movies/songs, etc.

When restricted to databases, that is, if we assume that individuals in a ranking problem can be suitably represented as database objects, the problem of top-k ranking can be formulated conveniently. An object may have one or more grades, or scores, one for each attribute; e.g., a color grade to tell how red it is and a size grade to tell how large it is. Each object can be assigned an overall grade by combining the attribute grades using an aggregation function. Then, the top-k objects are the k objects with the highest overall grades. Here, the meaning of top-k objects is clear and the main challenge is to compute the top-k objects efficiently in a database context, e.g., by using the Threshold Algorithm [1] and its variants [2,3].

The data above is certain. However, when the data in a database is uncertain, or difficult to be characterized quantitatively, the problem of top-k ranking presents an additional challenge – the semantics of top-k ranking. For instance, when we

☆ The work is partially supported by NSERC discovery grants RES0001375 and G121210405, and by 863 Program of China under grant 2009AA01Z150.
* Corresponding author.
E-mail address: you@cs.ualberta.ca (J.-H. You).

0898-1221/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2011.07.047

consider to purchase a car we typically do not have a unique sold price, but we may have a probability distribution of the sold prices; we may be uncertain about a person's height which is known to be between 1.7 and 1.8 m; we may be confident that one candidate is more experienced than another, etc.

The general problem of top-k ranking with uncertain data is highly complex and challenging. One recent approach in databases is to assume a limited form of uncertain data, represented by tuples, each of which holds a *score* representing the importance of the tuple and a *membership probability* indicating the level of confidence of the stored information.

Many different uncertain data models have been proposed for uncertain databases [4–7], with different semantics of top-k tuples [8–12], among which the approach in [8] can be viewed as a limited form of object ranking. It proposes what is called *top-k ranking for attribute-level uncertainty model*, in which an uncertain database is a table of tuples, each possessing one attribute whose value is uncertain. Here a tuple can be thought of as an object with one attribute and its values are represented by a discrete probability distribution. In a related context where tuples' scores are described by continuous probability distributions [13,14], the authors propose top-k ranking for objects where ranking is defined over one attribute.

In general, objects may have more than one attribute with uncertain data. For example, we may want to rent an apartment from a group of the best k choices, based on many factors; for simplicity let us consider *prices* and *locations*. The uncertainty of the former may be described by a probability distribution in a range of dollar values. The judgment of location could be fuzzy too; say we have 4 ranks for locations: *excellent, good, fair, poor*, and we may know that a location is good or excellent but not sure which one it should be. Assuming that the user provides the weights of the two factors on prices and locations (i.e., an aggregation function), we should be able to generate the top-k apartments from the uncertain data. To the best of our knowledge, there is no approach in the literature that defines top-k ranking for objects with multiple attributes with uncertain data.

Uncertain information may be presented in forms different from probabilities, for example, by relations. A noticeable example in real life is the practice of getting a *short list*. Consider a simple popularity contest: Given three contestants A, B, and C, suppose we know that A is more popular than B; but there is no information as how A is compared to C, neither B to C. Most observers will conclude that A is the top choice; however, the question of top-2 contestants seems not so obvious.

Relations have been employed in top-k ranking. For example, in [15] a notion of top-k queries is proposed, in which we do not know the exact value of an object, yet information about some relations between objects may be available. As another example of the use of relations in top-k ranking, the well-known algorithm, PageRank [16], is to rank web pages on the Internet. The information used in ranking is the reference relation between web pages (i.e. linkages between web pages). The link structure can be captured by a system of linear equations, from which the page rank of a web page can be computed.

To further motivate the need of constraints, consider the example of selecting the best k apartments again. Sometimes we may know some relation among the values of objects. For example, we may know the rent of apartment A is in the range of [600, 800] and that of apartment B is in [500, 700]. Although the rent is uncertain, we are sure that the rent of apartment B is cheaper than that of apartment A. Clearly, this relation should be taken into account of the final ranking results.

In this paper, we present a new ranking theory where two contributors to uncertainty of data are considered. The first is that the values of an attribute are given in terms of a probability distribution, and the second is that the values of attributes satisfy some stated constraints. We present our theory in three stages. The first assumes discrete domains. In this case, it is convenient and conceptually intuitive to define top-k objects using the notion of *possible worlds*. This material is given in Section 2. This formulation is extended to include continuous domains. We show that top-k ranking for objects in this context is closely related to some mathematical problems in high-dimensional spaces, in particular, the problem of computing volumes of a high-dimensional polyhedron represented by a system of inequations can be viewed as a subproblem of top-k object ranking of our theory. This material is presented in Section 3. Due to this relationship, we can apply the algorithms studied in mathematics for the former to compute top-k objects, where the constraints and aggregation function are linear expressions and the probability distributions are continuous uniform. Further in Section 4, we consider different weights to different positions of objects and add the aggregation values of objects to top-k ranking so that the ranking results are more reasonable.

In Section 5, we compare our ranking theory with related work in the literature. We show that a number of definitions of top-k objects in the literature are just special cases of our ranking theory. In addition, we illustrate by examples that our ranking theory can improve the quality of ranking results or extends the scope of applications for the existing approaches. Section 6 presents a summary and discusses future directions.

2. Top-k ranking for discrete domains

In this section, we present a theory of top-k ranking for objects whose data values are from discrete domains. The theory is formulated using the possible world semantics.

Definition 2.1. An *uncertain database* (or just a *database*) is a 5-tuple $D = \langle O, A, X, P, F\rangle$, where $O = \{o_1, \ldots, o_n\}$ is a set of objects; $A = \{a_1, \ldots, a_m\}$ a set of attributes; $X = \{x_{ij} \mid 1 \le i \le n,\ 1 \le j \le m\}$ where x_{ij} is a variable representing the value of the object o_i under a_j; $P = \{p_{ij} \mid 1 \le i \le n,\ 1 \le j \le m\}$ where p_{ij} is the probability distribution of variable x_{ij}, and $F = \{f_1, f_2, \ldots, f_l\}$ where each f_i is an equation or inequation on X.

In this section, we assume that each variable $x_{ij} \in X$ has a finite discrete domain, and therefore the probability distribution of a variable is also discrete.

Without confusion, given a database D, we will use o_i for objects, a_j for attributes, where $1 \leq i \leq n$ and $1 \leq j \leq m$.

For a database $D = \langle O, A, X, P, F\rangle$, each f_i in F can be written as

$$g(x_{11}, \ldots, x_{1m}, x_{21}, \ldots, x_{nm})\ \mathcal{R}\ 0$$

where $\mathcal{R} \in \{\leq, \geq, <, >, =\}$. We assume that g is a *continuous* function.

Definition 2.2. Let $D = \langle O, A, X, P, F\rangle$ be a database. An aggregation function for D is a mapping $t : \Re^m \to \Re$, where $\Re$ is the set of real numbers.

In our formulation, an application of an aggregation function, written $t(x_{i1}, \ldots, x_{im})$ (sometimes also written $t(o_i)$, for convenience) is to compute the collective value of object o_i across all attributes. We call such a value an *aggregation value* of object o_i.

For example, suppose in the given database there are two objects, o_1, o_2, and three attributes, a_1, a_2, a_3. Suppose an aggregation function is defined as: $t(x, y, z) = 2x + 3y + z$. Then, the aggregation value of object o_1 is $2x_{11} + 3x_{12} + x_{13}$ and that of o_2 is $2x_{21} + 3x_{22} + x_{23}$.

Given n objects and m attributes, we are interested in tuples of the form

$$\eta = (c_{11}, \ldots, c_{1m}, c_{21}, \ldots, c_{nm}) \tag{1}$$

where c_{ij} is a value of x_{ij}, i.e., a value of object o_i under attribute a_j. The probability of this tuple, denoted by $Pr(\eta)$, is defined by

$$Pr(\eta) = p_{11}(c_{11}) \cdots \times p_{21}(c_{21}) \times \cdots \times p_{nm}(c_{nm}). \tag{2}$$

A tuple of (1) represents one possible set of values for the underlying variables. Thus a tuple represents a scenario of all objects having their concrete attribute values. Although we know there is one *actual world* (the set of *actual* values for the variables), we do not know which one it is and thus every such set serves as a "possible world".

If $Pr(\eta) > 0$, the tuple η is nontrivial. Following the general idea of the possible world semantics, we define a *possible world* in this context to be a set of the values in η associated with their variables. Given a tuple η in the form (1), this can be conveniently denoted by

$$\eta' = \{[x_{11}, c_{11}], \ldots, [x_{1m}, c_{1m}], [x_{21}, c_{21}], \ldots, [x_{nm}, c_{nm}]\}.$$

That is, a possible world consists of $n \times m$ elements, each of which is a variable taking a value from its domain. In other words, η is an *assignment* of values to variables for all objects. For notational convenience, we will continue to use the notation of tuple in the form (1) to denote a possible world. Thus, the probability of the possible world η', denoted $Pr(\eta')$, is defined to be that of the corresponding tuple η, i.e., $Pr(\eta') = Pr(\eta)$.

For notational convenience, in the sequel, given an aggregation function t, an object o_i $(1 \leq i \leq n)$, and a tuple η of the form (1), the *aggregation value* of o_i w.r.t. η, denoted $t_{o_i}(\eta)$, is the aggregation value of o_i computed by t when variable x_{ij} take values c_{ij} $(1 \leq j \leq m)$.

Then, whether an object o_i is a top-k object is determined by how many possible worlds that "support" o_i. Formally, let $D = \langle O, A, X, P, F\rangle$ be a database, $\eta = (c_{11}, \ldots, c_{1m}, c_{21}, \ldots, c_{nm})$ a possible world, and t an aggregation function. Given an object o_i, if there are at least $n - k$ other objects $o_{i'}$ such that $t_{o_i}(\eta) \geq t_{o_{i'}}(\eta)$, then we say that the possible world η *supports* object o_i (or, η is *a support* to o_i).

In other words, η supports object o_i whenever η places o_i ahead of at least $n - k$ other objects, under the aggregation function t. This is like *casting a vote*. η supports o_i when it casts its vote to o_i as a top-k object.

We now bring the constraints into the formulation.

Definition 2.3. Let $D = \langle O, A, X, P, F\rangle$ be a database. A possible world $\eta = (c_{11}, \ldots, c_{1m}, c_{21}, \ldots, c_{nm})$ is said to be *effective* if the values in this possible world satisfy all the inequations and equations in F.

If a support to an object is effective, it will be called an *effective support*.

For each object o, we define the *support set of* o, denoted by S_o, to be the set of all the possible worlds that are effective supports to o.

Definition 2.4. Let $D = \langle O, A, X, P, F\rangle$ be a database, t an aggregation function. The *support strength* of an object o is defined as $\sum_{\eta \in S_o} Pr(\eta)$.

Definition 2.5. Let $D = \langle O, A, X, P, F\rangle$ be a database, t be an aggregation function. The top-k objects in D are the k objects with highest support strengths.

Here we give some examples of ranking problems covered by this formulation of top-k ranking.

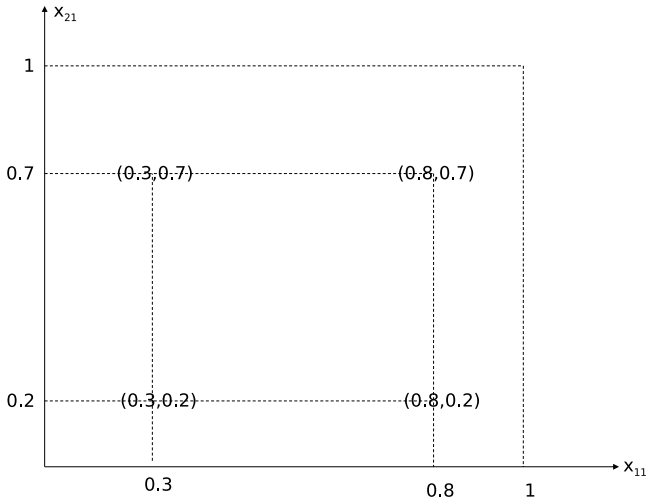

Fig. 1. Example 2.6.

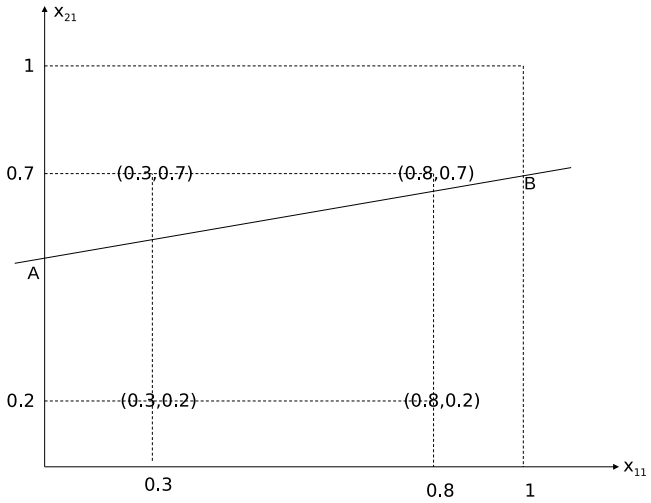

Fig. 2. Example 2.7.

Example 2.6. Suppose there are two objects $O = \{o_1, o_2\}$ and one attribute $A = \{a_1\}$. We thus have two variables $X = \{x_{11}, x_{21}\}$. Assume both domains are [0, 1] and the probability distribution of x_{11} is $p_{11}(x_{11} = 0.3) = 0.7$ (meaning that the probability of the value of x_{11} being 0.3 is 0.7, similarly below) and $p_{11}(x_{11} = 0.8) = 0.3$, and that of x_{21} is $p_{11}(x_{21} = 0.2) = 0.4$ and $p_{21}(x_{21} = 0.7) = 0.6$.

The two variables x_{11}, x_{21} in this example can be viewed intuitively as a 2-dimensional space. A possible world can then be viewed as a point in this space, and the variable–value pairs in a possible world as coordinate values. There are 4 possible worlds in this example, which are shown in Fig. 1. The probability of the possible world (0.3, 0.2) is 0.28. It supports o_1. The probability of the possible world (0.3, 0.7) is 0.42. It supports o_2. The probability of the possible world (0.8, 0.2) is 0.12. It supports o_1. The probability of the possible world (0.8, 0.7) is 0.18. It supports o_1. It can be easily seen that the support strength of o_1 is 0.58 and the support strength of o_2 is 0.42. Thus, o_1 is the top-1 object.

Example 2.7. The conditions are the same as in Example 2.6, but we have a constraint, $x_{21} > 0.2x_{11} + 0.5$. This is shown in Fig. 2. The constraint is captured by the line between A and B in the sense the possible worlds strictly above it (note $>$ in the constraint) satisfy the constraint. Apparently, there are only two effective possible worlds. The possible world (0.3, 0.7) supports o_2, and the possible world (0.8, 0.7) supports o_1. The support strength of o_1 is 0.18 and the support strength of o_2 is 0.42. So o_2 is the top-1 object.

3. Extension to continuous domains in high-dimensional space

In the definition above, we used possible worlds to define top-k ranking, where probability distribution is assumed to be discrete. When the probability distribution is continuous, we have continuous domains for variables. If we continue to use a possible world to represent a point in a high dimensional space, then we are going to have infinitely many possible

worlds. In this case, it is more convenient to represent a point by its coordinate values. This does not change the nature of the semantics even for discrete domains. However, some technical details need to be handled for continuous domains.

The definitions of database $D = \langle O, A, X, P, F\rangle$ and aggregation function are the same as before. Each variable $x_{ij} \in X$ can be viewed a dimension in an $n \times m$ dimensional space, and a tuple $\eta = (c_{11}, \ldots, c_{1m}, c_{21}, \ldots, c_{nm})$ represents a point by its coordinate values. We then can represent the support set in Section 2 by a system of equations and inequations, which over n variables defines a q-dimensional space, where $q \leq n$. This space contains all the points whose coordinate values satisfy all the equations and inequations and does not contain any points whose coordinates conflict with any equation or inequation.

We assume that the domain of each variable in X is bounded finitely, i.e.,

$$l_{ij} \leq x_{ij} \leq u_{ij} \quad (1 \leq i \leq n, 1 \leq j \leq m) \tag{3}$$

where l_{ij} and u_{ij} are real numbers. To get the top-k objects, we need to define some spaces.

The first space, denoted by Γ, is defined by all the inequations and equations in F and the domain of each variable in X.

The second space, denoted V_i w.r.t o_i, is defined by the domain of each variable in X and the constraints for the notion of support — *η supports an object o iff there are at least $n - k$ other objects o' such that $t_o(\eta) \geq t_{o'}(\eta)$*. This space can be represented by systems of equations and inequations and we will describe it later.

Let $D = \langle O, A, X, P, F\rangle$ be a database and t an aggregation function. For an object o_i, we define the *support space to o_i* to be the space $\Upsilon_i = V_i \cap \Gamma$. We will say that *all the points in Υ_i support o_i*.

For $D = \langle O, A, X, P, F\rangle$, we assume that all the probability distributions in P are independent. If all the probability distributions in P are discrete, we get the joint probability mass function of X:

$$f(x_{11}, x_{12}, \ldots, x_{1m}, x_{21}, \ldots, x_{nm}) = \prod_{i=1}^{n}\prod_{j=1}^{m} p_{ij}.$$

We define the *support strength of o_i* as (assuming $\eta = (c_{11}, \ldots, c_{1m}, c_{21}, \ldots, c_{nm})$):

$$\Lambda(o_i) = \sum_{\eta \in \Upsilon_i} f(x_{11} = c_{11}, \ldots, x_{21} = c_{21}, \ldots, x_{nm} = c_{nm}).$$

If some of the probability distributions in P are continuous, we get the joint probability density function of X:

$$f(x_{11}, x_{12}, \ldots, x_{1m}, x_{21}, \ldots, x_{nm}) = \prod_{i=1}^{n}\prod_{j=1}^{m} p_{ij}.$$

Let Θ be the set of points which contain all the points with joint probability density function value greater than 0 and does not contain any point with joint probability density function value equal to 0.

Consider all the spaces below

$$\Upsilon_i \cap \Theta \quad (1 \leq i \leq n). \tag{4}$$

We discuss these spaces in two cases.

First, if all the spaces in expression (4) are empty, we define *support strength* of any object $o_i \in O$ to be 0.

Second, if not all the spaces in (4) are empty, let s ($0 \leq s \leq n \times m$) be the maximal dimension among all spaces in (4). Given an i ($1 \leq i \leq n$), let $\chi_i = \{\eta_1, \ldots, \eta_q\}$, where q is a positive integer, be the set of s-dimensional spaces in $\Upsilon_i \cap \Theta$, and

$$x_{u_{e1}v_{e1}}, \ldots, x_{u_{ed}v_{ed}}$$

where $1 \leq e \leq q$, $1 \leq d \leq n \times m$, $1 \leq u_{ew} \leq n$, $1 \leq v_{ew} \leq m$, $1 \leq w \leq d$, be the variables which take different values in η_e. Then we define the *support strength* of $o_i \Lambda(o_i)$ as follows

- If $\chi_i = \emptyset$ then $\Lambda(o_i) = 0$.
- If χ_i contains q points, then $\Lambda(o_i) = \sum_{(c_{11},\ldots,c_{nm}) \in \chi_i} f(x_{11} = c_{11}, \ldots, x_{nm} = c_{nm})$.
- If χ_i contains q s-dimensional spaces ($1 \leq s \leq n \times m$) then $\Lambda(o_i) = \sum_{\eta_e \in \chi_i} \int \cdots \int_{\eta_e} f(x_{11}, \ldots, x_{nm}) dx_{u_{e1}v_{e1}} \ldots dx_{u_{ed}v_{ed}}$ (if variable x_{ij} can only take a fixed value c_{ij} in η_e, replace x_{ij} with c_{ij} in $\int \cdots \int_{\eta_e} f(x_{11}, \ldots, x_{nm})$).

Then, the top-k objects in D are the k objects with the highest support strengths.

Recall that we have defined V_i w.r.t o_i. Now let us see how to formally express V_i in some systems of equations and inequations. For an object $o_i \in O$, let W_{ij} be a set of $n - k$ objects in $O - \{o_i\}$, i.e., $W_{ij} = \{o_{a_1}, o_{a_2}, \ldots, o_{a_{n-k}}\}$ where $o_{a_i} \neq o_i$. Because we can choose any $n - k$ objects from $O - \{o_i\}$, we know there are C_{n-1}^{n-k} different W_{ij}. So $1 \leq j \leq C_{n-1}^{n-k}$.

Let $W_{ij} = \{o_{a_1}, o_{a_2}, \ldots, o_{a_{n-k}}\}$. Let U_{ij} denote the following set of inequations

$$t(x_{i1}, \ldots, x_{im}) \geq t(x_{a_h 1}, \ldots, x_{a_h m}) \quad 1 \leq h \leq n - k$$
$$l_{ij} \leq x_{ij} \leq u_{ij} \quad (1 \leq i \leq n, 1 \leq j \leq m)$$

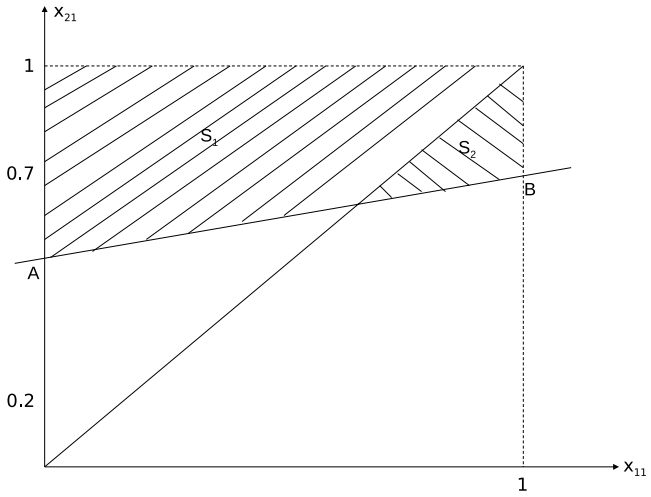

Fig. 3. Example 3.1.

and V_{ij} denote the space defined by U_{ij}. Define

$$V_i = \bigcup_{1 \le j \le C_{n-1}^{n-k}} V_{ij}.$$

For a database $D = \langle O, A, X, P, F \rangle$, where $O = \{o_1, \ldots, o_n\}$ and $A = \{a_1, \ldots, a_m\}$, we assume the maximal dimension of all the support spaces of objects is s. When the probability distributions in P are all continuous uniform distributions over entire domains, we just need to calculate the volume of the s dimension support space to each object o_i to get top-k objects. Because each point has the same probability density function value and the support strength of an object is the product of the volume of the s dimension support space to this object and the probability density function value, we can use the volume of the s dimension support space to an object to measure the support strength of the object. So in this situation, the top-k objects in D are the k objects with largest volumes of s dimension support spaces.

Example 3.1. We keep all the conditions as in Example 2.7, except the probability distribution. We change the probability distribution of x_{11} and x_{21} to be continuous uniform distribution. The example is illustrated in Fig. 3. Since the probability distribution of each variable is continuous uniform distribution, we use volume to measure the support strength to an object. For this example, as the problem is in 2-dimensional space, we can use area to measure the support strength to an object. All the points inside S_1 support o_1 and all the points inside S_2 support o_2. S_1 is the support space to o_1 and S_2 is the support space to o_2. Clearly, in this example the maximal dimension among all support spaces is 2. S_1 can be described by the following inequations:

$$x_{11} \ge x_{21}$$
$$x_{21} > 0.2x_{11} + 0.5$$
$$0 \le x_{11} \le 1$$
$$0 \le x_{21} \le 1.$$

S_2 can be described by the following inequations:

$$x_{21} \ge x_{11}$$
$$x_{21} > 0.2x_{11} + 0.5$$
$$0 \le x_{11} \le 1$$
$$0 \le x_{21} \le 1.$$

As the area of S_2 is larger than the area of S_1, o_2 is the top-1 object.

Example 3.2. We keep all the conditions as in Example 3.1, except the constraint. We change the constraint to $x_{21} = 0.2x_{11} + 0.5$. The example is shown in Fig. 4. We can use the length of a line to measure the support strength of an object. All the points in line BC supports o_1 and all the points in line AB supports o_2. Line BC is the support space to o_1 and line AB is the support space to o_2. In this example, the maximal dimension among all support spaces is 1. Because the length of AB is longer than the length of BC, o_2 is the top-1 object. This example shows that constraints can be equations.

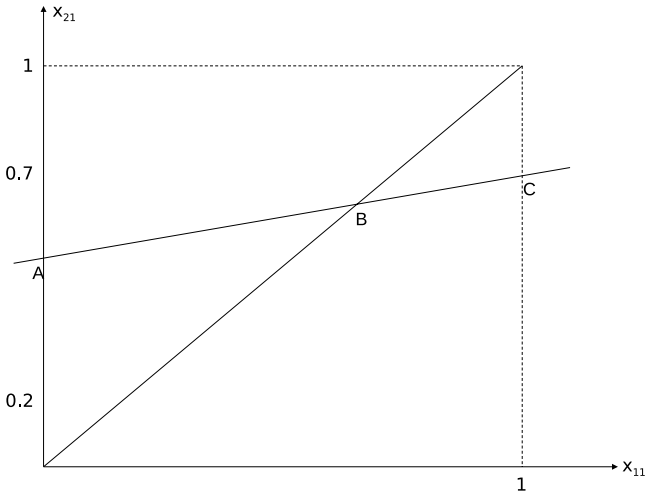

Fig. 4. Example 3.2.

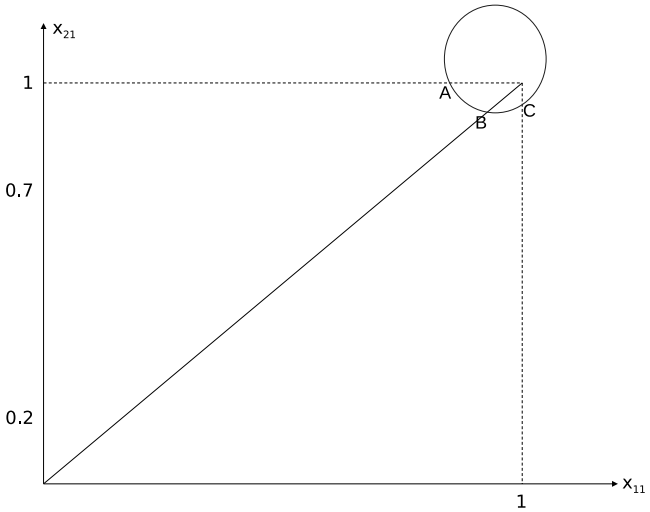

Fig. 5. Example 3.3.

Example 3.3. We keep all the conditions in Example 3.1, except the constraint. We change the constraint to $(x_{11} - 0.9)^2 + (x_{21} - 1.1)^2 = 0.04$. The example is shown in Fig. 5. All the points in the arc BC support o_1 and all the points in the arc AB support o_2. Because the length of the arc AB is longer than the length of the arc BC, o_2 is the top-1 object. This example shows that constraints can be non-linear equations or inequations.

Example 3.4. We keep all the conditions in Example 3.1, except the probability distribution. We change the probability distribution of $x_{11}p_{11}$ and the probability distribution of $x_{21}p_{21}$ to continuous non-uniform distribution. The example is also shown in Fig. 3. We cannot use volume to measure the support strength. We have to use integration. As all the points inside S_1 support o_1, we can use the following integration to compute the support strength of o_1: $\iint_{S_1} p_{11}p_{21}dx_{11}dx_{21}$. Similarly, we can use the following integration to compute the support strength of o_2: $\iint_{S_2} p_{11}p_{21}dx_{11}dx_{21}$.

3.1. Computation

Here we presented a limited study on the method of computation. We restrict F to linear inequations and the aggregation function to be linear. We also restrict the probability distributions in P to continuous uniform distributions over the whole domains. Let the maximal dimension of all the support spaces to objects be s. We only need to compute the volumes of s dimension support spaces to objects to get the top-k objects. Since the aggregation function is linear, we can see that U_{ij} only contains linear inequations. As F only contains linear inequations, the s dimension support space to object o_i is the union of the high-dimensional spaces each of which is a high-dimensional polyhedron represented by a system of linear inequations. Thus, under these restrictions, the computation of top-k objects can be transformed to the subproblems each of which computes the volume of a high-dimensional polyhedron represented by a system of linear inequations.

Given an algorithm for the computation of volumes of a high dimensional polyhedron represented by a system of linear inequations (e.g. see [17]), we can apply it to compute top-k objects.

Example 3.5. Let $D = \langle O, A, X, P, F\rangle$ be a database, with $O = \{o_1, o_2, o_3\}$, $A = \{a_1, a_2\}$, $X = \{x_{ij} \mid 1 \le i \le 3, 1 \le j \le 2\}$, and $F = \{x_{11} \ge x_{21}\}$ (i.e., the value of o_1 under a_1 is equal or greater than o_2 under a_1). Assume the aggregation function is $t(x_{i1}, x_{i2}) = x_{i1} + x_{i2}$ and the domain of each variable is [0, 1]. We want to find top-2 objects.

The support space of o_1 composes of two parts, which are the high-dimensional polyhedra represented by the following two systems of linear inequalities, respectively:

$$0 \le x_{ij} \le 1 \quad 1 \le i \le 3, 1 \le j \le 2$$
$$x_{11} \ge x_{21}$$
$$x_{11} + x_{12} \ge x_{21} + x_{22}$$

and

$$0 \le x_{ij} \le 1 \quad 1 \le i \le 3, 1 \le j \le 2$$
$$x_{11} \ge x_{21}$$
$$x_{11} + x_{12} \ge x_{31} + x_{32}.$$

We can find that the maximal dimension of all the support spaces to objects in D is 6. The support space of o_1 is 6 dimensions. So we just need to compute the volume of the support space of o_1. Then we can use an algorithm, for example, the one in [17], to compute the volume of a high-dimensional polyhedron represented by a system of inequations. The volume of the support space of object o_1 is the sum of the volumes of the two polyhedra. The support space of o_2 and o_3 also has 6 dimensions. We can compute the volume of the support space of objects o_2 and o_3 similarly. Then the top-2 objects are the 2 objects with largest 2 volumes of support spaces.

4. Further extensions

From a point in a high dimensional space, we can rank the underlying objects according to their aggregation values. In this ranking, each object has a position. In the formulation of the previous section, there is no difference for an object to be ranked at the first position or 2nd position. Each position in the top k positions has the same influence to the final ranking result, and each position lower than k has no influence to the final ranking result. This is reasonable in some applications. But sometimes it is desirable to assign weights to different positions. For example, if an object ranks the first in a point, it should get more support than the object ranked the second from the same point. This concept is common in real life. For example, in a sporting event a gold medal weighs more than a silver medal.

In this section, we define the position of an object in a point to be the number of objects with higher aggregation values.

In our original ranking theory, the aggregation values of objects are used to give an order of objects in a point. After the order is determined, the aggregation value itself will not be used in ranking again. For example, candidate A is preferred over candidate B for trustworthiness, but the extent of this preference is not considered previously. Therefore, sometimes it is desirable to include the aggregation values in the process of ranking. In this section, we extend our ranking theory in Section 3 so that different positions can get different weights and aggregation values of objects themselves can be included in top-k ranking.

We note that the notion of *parameterized ranking functions* introduced in [10] embodies a similar concept.

We now give the details. Let database $D = \langle O, A, X, P, F\rangle$ and aggregation function t be the same as before.

The space V_i^b w.r.t. o_i is defined by the domain of each variable in X and the constraints for the notion of *b-support – a point η in a high dimensional space gives a b-support to an object o iff there are exactly b other objects o' such that $t_{o'}(\eta) > t_o(\eta)$.*

As in the previous section, we let Γ be the space defined by the inequations or equations in F. We define $\Upsilon_i^b = V_i^b \cap \Gamma$ to be the *b-support space* to o_i. Note that o_i is in the b-th position in all the points of the b-support space to o_i.

Let us use $Position(o_i)$ to represent the position of an object in a point. Let $\omega : \Re \times N \to \Re$ be a weight function. The expression $\omega(t(o_i), Position(o_i))$ specifies a weight for an object in a position. This weight function includes the aggregation value of an object. $\omega(t(o_i), Position(o_i))$ can be defined in many different ways. It can be independent of $t(o_i)$ or $Position(o_i)$. If $\omega(t(o_i), Position(o_i)) = 1$ $((1 \le Position(o_i)) \le k)$ and $\omega(t(o_i), Position(o_i)) = 0$ $((k + 1 \le Position(o_i)) \le n)$, then we get the same top-k ranking definition as the one in Section 3.

For $D = \langle O, A, X, P, F\rangle$, we still assume that all the probability distributions in P are independent. If all the probability distributions in P are discrete, we get the joint probability mass function of X:

$$f(x_{11}, x_{12}, \ldots, x_{1m}, x_{21}, \ldots, x_{nm}) = \prod_{i=1}^{n}\prod_{j=1}^{m} p_{ij}.$$

We define the *support strength* of o_i as $\Lambda(o_i) = \sum_{b=0}^{n-1} \sum_{(c_{11},\ldots,c_{1m},c_{21},\ldots,c_{nm})\in\Upsilon_i^b} \omega(t(o_i), b) f(x_{11} = c_{11}, \ldots, x_{1m} = c_{1m}, x_{21} = c_{21}, \ldots, x_{nm} = c_{nm})$.

If some of the probability distributions in P are continuous, we get the joint probability density function of X:

$$f(x_{11}, x_{12}, \ldots, x_{1m}, x_{21}, \ldots, x_{nm}) = \prod_{i=1}^{n}\prod_{j=1}^{m} p_{ij}$$

We define Θ to be the same as in Section 3.

We observe all the spaces below

$$\Theta \cap \Upsilon_i^b \quad (1 \le i \le n, 0 \le b \le n-1). \tag{5}$$

We discuss these spaces in two cases.

First, if all the spaces in (5) are empty, we define *the support strength* of any object $o_i \in O$ to be 0.

Second, if not all the spaces in (5) are empty, let s ($0 \le s \le n \times m$) be the maximal dimension of spaces in (5). Given an i ($1 \le i \le n$), let $\chi_i^b = \{\eta_1, \ldots, \eta_q\}$, where q is a positive integer, be the set of s-dimensional spaces in $\Upsilon_i^b \cap \Theta$ and

$$x_{u_{e1}v_{e1}}, \ldots, x_{u_{ed}v_{ed}}$$

where $1 \le e \le q$, $1 \le d \le n \times m$, $1 \le u_{ew} \le n$, $1 \le v_{ew} \le m$, $1 \le w \le d$, be the variables which can take different values in η_e, then we define *the b-support strength of* o_i, $\Lambda^b(o_i)$, as follows

- If $\chi_i^b = \emptyset$ then $\Lambda^b(o_i) = 0$;
- If χ_i^b contains q points, then $\Lambda^b(o_i) = \sum_{(c_{11},\ldots,c_{nm})\in\chi_i^b} \omega(t(o_i), b) f(x_{11} = c_{11}, \ldots, x_{nm} = c_{nm})$.
- If χ_i^b contains qs-dimension spaces ($1 \le s \le n \times m$) then $\Lambda^b(o_i) = \sum_{\eta_e\in\chi_i^b} \int\cdots\int_{\eta_e} \omega(t(o_i), b) f(x_{11}, \ldots, x_{nm})\, dx_{u_{e1}v_{e1}} \cdots dx_{u_{ed}v_{ed}}$. (If variable x_{ij} can only take a fixed value c_{ij} in η_e, replace x_{ij} with c_{ij} in $\int\cdots\int_{\eta_e} \omega(t(o_i), b) f(x_{11}, \ldots, x_{nm})$.)

We define *the support strength of* o_i: $\Lambda(o_i) = \sum_{b=0}^{n-1} \Lambda^b(o_i)$.

The top-k objects in D are the k objects with highest support strength (if smaller values of weights are considered more important, then the top-k objects in D are the k objects with lowest support strength).

Here we show how to express V_i^b as some systems of inequations. For an object $o_i \in O$, let W_{ij}^b ($0 \le b \le n-1$) be a set of b objects in $O - \{o_i\}$. Because we can choose any b objects in $O - \{o_i\}$, there are C_{n-1}^b different W_{ij}^b for each b. So for each b, $1 \le j \le C_{n-1}^b$.

Let $W_{ij}^b = \{o_{a_1}, o_{a_2}, \ldots, o_{a_b}\}$. And let $O - W_{ij}^b - \{o_i\} = \{o_{a_{b+1}}, o_{a_{b+2}}, \ldots, o_{a_{n-1}}\}$. Let U_{ij}^b denote the following set of inequations:

$$\begin{aligned}
&t(x_{a_h 1}, \ldots, x_{a_h m}) > t(x_{i1}, \ldots, x_{im}) \quad 1 \le h \le b\\
&t(x_{i1}, \ldots, x_{im}) \ge t(x_{a_h 1}, \ldots, x_{a_h m}) \quad b+1 \le h \le n-1\\
&l_{ij} \le x_{ij} \le u_{ij} \quad (1 \le i \le n, 1 \le j \le m).
\end{aligned}$$

Let V_{ij}^b denote the space defined by U_{ij}^b. Let

$$V_i^b = \bigcup_{1\le j\le C_{n-1}^b} V_{ij}^b.$$

When we restrict F and the aggregation function to linear expressions and require that the probability distribution in P be continuous uniform distributions, we can also use the method introduced in Section 3.1 for computation.

5. Comparison with related work

5.1. Comparison with our previous work

In [18], we propose a ranking theory for uncertain data with constraints. For a database $D = \langle O, A, X, P, F\rangle$, the ranking theory defines a semantics for top-k objects in D. Both the ranking theory in this paper and the ranking theory in [18] rank objects with uncertain data. The uncertainty in both cases is presented in two formats: probability distribution of values of objects and relations among values of objects.

However, these two theories give different semantics for objects with uncertain data. The theory proposed in [18] ranks *top-k object sequences*, which are sequences of distinct objects from O. The top-k object sequence with highest *appearance probability* is chosen as the top-k objects. This is actually a "team" ranking, where an object sequence is considered as a whole. It focuses on the whole effects when k objects combine together in some order. It compares all the possible sequences of k objects to see which one is stronger under some criteria. An object is a top-k object because the sequence with the highest strength contains this object, somewhat independent of the desirability of the object alone.

The ranking theory in this paper is *individual ranking*, which chooses the k objects with highest support strengths to be the top-k objects. An object is chosen as a top-k object because of the strength of itself, independent of the choices of the other top-k objects.

5.2. Constraints in top-k ranking

To illustrate why we introduce constraints in our ranking theory, we show two examples. The first illustrates that our ranking theory improves ranking quality over the previous approach in [15]. The second shows that Pagerank [16] falls into our theory.

Agrawal et al. [15] propose a method of ranking objects when some relations between objects are specified. These relations can be some arbitrary orderings which may or may not satisfy antisymmetry or transitivity. In the paper, an example of actors is presented, where we want to rank their popularity without any knowledge of the exact values of degrees of popularity. Instead, we know some preferences between actors, such as some are more popular than some others. A simple method is then used to find the k most popular actors. Assume there are m different sets of preferences each of which is specified by a partial order on actors. Given a partial order, the method in the paper finds a total order that satisfies the partial order. Then this total order is updated by putting the actor without any preferences with other actors at the end of the order. Then for each actor, a score $n - i + 1$ is assigned, where n is the number of actors and i the position of the actor in this order. Then for each set of preferences, each actor has a score. An actor thus has m scores. The paper gives an aggregation function designed by the authors themselves to get an aggregation score which combines the m scores for each actor. Then the top-k actors are the k actors with highest aggregation scores.

To see the difference of the method in [15] from ours, assume there is only one set of preferences which consist of a partial order. Using the method in [15], two actors can rank before or after each other if there are no preferences between them. For instance, suppose there are 100 actors $A_1, \ldots, A_{100}$. The partial order of the preferences is: A_1 is more popular than $A_2, \ldots, A_{98}$ and A_{99} is more popular than A_{100}. Using the method in [15], we cannot tell which one between A_1 and A_{99} is more popular. But our theory says A_1 has more support strength than A_{99}. Here we assume each actor has a popularity score and this score is an uniform probability distribution in [0, 1]. And we assume the weight function is only related to positions of actors in possible worlds and positions in front have higher weights. The partial order is the constraints. Then we get our ranking result which ranks A_1 ahead of A_{99}. Due to the preferences, A_1 is more popular than most other actors and A_{99} is more popular than only one actor, A_1 has more chances to be ranked ahead of A_{99}. These observations lead to the conclusion that out ranking theory gives more reasonable ranking results than the one in [15].

PageRank [16] is an algorithm for page ranking in the World Wide Web. The relations between pages can be described by a group of linear equations:

$$PR(p_i) = \frac{1-d}{N} + d \sum_{p_j \in M(p_i)} \frac{PR(p_j)}{L(p_j)}$$

where p_i is a page, $PR(p_i)$ is the page rank of page p_i, $M(p_i)$ is the set of pages that link to p_i, d is the *damping factor*, $L(p_j)$ is the number of outbound links on page p_j, and N is the total number of pages.

The ranking problem can be thought as an extreme case of our ranking theory given in Section 4, where pages are objects and there is only one attribute (let us call it *page rank*). Thus, we have N variables, $X = \{x_{11}, x_{21}, \ldots, x_{N1}\}$. Let us assume the domain of each variable is [0, 1]. Let us further assume that the probability distribution of each variable is a continuous uniform distribution over the entire domain. For simplicity, assume the joint probability density function is 1. Under our theory, the set of equations above can be viewed as the constraint set F. Let the aggregation function be a simple one, $t(o_i) = x_{i1}$, and the weight function be $\omega(t(o_i), Position(o_i)) = t(o_i) = x_{i1}$. Clearly, there is exactly one solution for this system of equations, which corresponds to the coordinate values of a single point in the N-dimensional space. Note that this point is the support space of each page, and the support strength of a page is the value of the page rank of the page.

If some equations are missing in the group of equations above such that the number of variables is more than the number of equations for some reasons, our ranking theory can still provide the page ranks for pages, because all we need to do is to compute the support strength for each object, and therefore to determine the page rank of each page.

5.3. Comparison with top-k object ranking with discrete probability distribution

The question of top-k ranking in uncertain databases has attracted much attention recently. Many different top-k tuple ranking definitions have been proposed [8–12]. As remarked in Section 1, in [8] the authors propose the definition of top-k ranking for attribute-level uncertainty model in uncertain databases, where an uncertain database is a table of N tuples with one attribute whose value is uncertain. The values of the uncertain attribute of a tuple is described by a discrete probability distribution. A possible world consists of N tuples each of which takes one value for the domain of the attribute, according to its probability distribution. The probability of a possible world is the product of the probabilities of all the values of the tuples in this possible world. Let Ω be the set of all the possible worlds. The rank of a tuple in a possible world W is defined to be the number of tuples whose values are higher than this tuple. The rank of a tuple t_i in W

$$rank_W(t_i) = |\{t_j \in W \mid v_j > v_i\}|.$$

The expected rank of t_i is

$$r(t_i) = \sum_{W \in \Omega, t_i \in W} Pr(W) \cdot rank_W(t_i)$$

Then the top-k tuples are the k tuples with lowest expected ranks.

We can think of a tuple above as an object. Consider the definition of the database $D = \{O, A, X, P, F\}$ in Section 4. The object set O is the set of tuples with one attribute. Thus we have n variables, $X = \{x_{11}, x_{21}, \ldots, x_{n1}\}$, each with a discrete probability distribution. There are no constraints. So F is empty. Assume the weight function is $\omega(t(o_i), Position(o_i)) = Position(o_i)$, where $Position(o_i)$ is $rank_W(t_i)$ (a possible world W can be thought of as a point). So the definition of top-k ranking for attribute-level uncertainty model in [8] can be thought as a special case of our extended ranking theory in Section 4.

For the definition of top-k ranking in [8], our definition can improve the quality of the ranking results by considering different weights for positions. The weights of positions in [8] are fixed. But in real applications, we may wish to adjust the weights of positions depending on different conditions. Sometimes we may want to include the values of objects into the process of ranking. This is allowed in our ranking theory but not in [8].

5.4. Comparison with top-k object ranking with continuous probability distribution

In [14], the authors propose to rank records with uncertain scores in databases. In some applications, the score of a record t_i is modeled as a probability density function f_i defined on a score interval $[lo_i, up_i]$. The interval-based score representation can induce a partial order over database records. If a record's lower bound is higher than another record's upper bound, we can order this record ahead of the other record. Otherwise, there is no order between these two records. Thus, we get a partial order among these records. The linear extensions of the partial order are all the possible total orders consistent with the partial order. In the paper, each linear extension is associated with a probability, and different ranking queries are considered, one of which is called *Uncertain Top Rank (UTop-Rank)*. A UTop-Rank(i, j) query reports the most probable record to appear at any rank $i \cdots j$ in possible linear extensions. A l-UTop-Rank(i, j) query reports the l most probable records to appear at a rank $i \cdots j$. It can be shown that if we treat records as objects, the answer to k-Utop-Rank$(1, k)$ query is the top-k objects defined in Section 3.

In [14], there is another definition of ranking query called *Uncertain Top Prefix (UTop-Prefix)*. A UTop-Prefix(k) query reports the most probable linear extension prefix of k records. This definition is just a special case of our definition of top-k objects in [18].

In [13], the authors propose a new definition which is called *parameterized ranking function* to rank tuples with uncertain scores which are captured by continuous probability distribution. We assume we are given a probabilistic dataset consisting of n tuples, where each tuple has an uncertain score which is described by a continuous probability distribution. A tuple is also associated with an existence probability to represent the probability of existence of this tuple in the dataset (a tuple may or may not exist in the dataset). The uncertain tuples and attribute scores are independent of each other. A possible world consists of some tuples with fixed values. A tuple may have an infinite number of values (from a continuous domain), so there can be an infinite number of possible worlds. We use $r(t)$ to denote the position of tuple t in a possible world. If a tuple does not exist in a possible world, we denote its position by ∞. $Pr(r(t) = j)$ is the probability that t is ranked at position j. $\omega : T \times N \to C$ is a weight function that maps a tuple-rank pair to a complex number. The parameterized ranking function is defined as: $\Upsilon_\omega(t) = \sum_{i>0} \omega(t, i) Pr(r(t) = i)$. A top-$k$ query returns the k tuples with the highest $|\Upsilon_\omega|$ values.

If we treat tuples as objects and add the following three restrictions, the answer of top-k query is then the top-k objects defined in Section 4. First, we assume each tuple's existence probability is 1. That is, it is certain that all the tuples in a given dataset do exist. It follows that each possible world contains n tuples. Second, we restrict the weight function to map to real numbers. Finally, we assume that the domain of each tuple's score is defined by an interval (whose bounds are real numbers). Under these restrictions, let the object set O be the set of tuples, the attribute be the only one in A, the variable set be $X = \{x_{11}, x_{21}, \ldots, x_{n1}\}$, and F be empty. Then, the answer of top-k query here is just a special case of our definition of top-k objects in Section 4.

Both definitions in [13,14] consider only one attribute. But our theory can handle multiple attributes with uncertain values. So our theory extends the application areas of the ranking on uncertain scores with continuous probability distributions.

6. Conclusion

In this paper, we have presented a ranking theory for objects with uncertain data. We apply the possible worlds semantics to define top-k objects for discrete uncertain data. Then we extend the definition of top-k ranking in high-dimensional space for objects with discrete or continuous uncertain data. We have identified the problem of top-k ranking for objects to be an extension of the problem of computing volumes of high-dimensional polyhedron represented by a system of linear inequations. We further extend this theory to add weights to objects' positions and aggregation values in determining ranking.

As future work, it is important to investigate the computational properties of the theoretical framework presented in this paper. As the general problem is highly complex, one direction is approximation algorithms, and the other is to extend efficient algorithms for one attribute [13] to multiple attributes where uncertainty is generally sparse. In this case, we believe that the theory presented in this paper can be practically useful. Another important future direction is how to discover appropriate aggregation functions and constraints for particular application domains. Machine learning techniques should be helpful for this purpose.

References

[1] R. Fagin, A. Lotem, M. Naor, Optimal aggregation algorithms for middleware, Journal of Computer and System Sciences 66 (4) (2003) 614–656.
[2] K. Chakrabarti, V. Ganti, J. Han, D. Xin, Ranking objects by exploiting relationships: computing top-*k* over aggregation, in: Proc. SIGMOD, 2006, pp. 371–382.
[3] S. Chaudhuri, L. Gravano, A. Marian, Optimizing top-*k* selection queries over multimedia repositories, IEEE Transactions on Knowledge and Data Engineering 16 (8) (2004) 992–1009.
[4] S. Abiteboul, P.C. Kanellakis, G. Grahne, On the representation and querying of sets of possible worlds, in: Proc. SIGMOD, 1987, pp. 34–48.
[5] D. Barbara, H. Garcia-Molina, D. Porter, The management of probabilistic data, IEEE Transactions on Knowledge and Data Engineering 4 (5) (1992) 487–502.
[6] N.N. Dalvi, D. Suciu, Management of probabilistic data: foundations and challenges, in: Proc. PODS, 2007 pp. 1–12.
[7] N. Friedman, L. Getoor, D. Koller, A. Pfeffer, Learning probabilistic relational models, in: Proc. IJCAI, 1999, pp. 1300–1309.
[8] G. Cormode, F. Li, K. Yi, Semantics of ranking queries for probabilistic data and expected rank, in: Proc. ICDE, 2009, pp. 305–316.
[9] M. Hua, J. Pei, W. Zhang, X. Lin, Ranking queries on uncertain data: a probabilistic threshold approach, in: Proc. SIGMOD, 2008, pp. 673–686.
[10] J. Li, B. Saha, A. Deshpande, A unified approach to ranking in probabilistic databases, in: Proc. VLDB, 2009, pp. 502–513.
[11] M.A. Soliman, I.F. Ilyas, K.C.-C. Chang, Top-k query processing in uncertain databases, in: Proc. ICDE, 2007 pp. 896–905.
[12] K. Yi, F. Li, G. Kollios, D. Srivastava, Efficient processing of top-*k* queries in uncertain databases with x-relations, IEEE Transactions on Knowledge and Data Engineering 20 (12) (2008) 1699–1711.
[13] J. Li, A. Deshpande, Ranking continuous probabilistic datasets, Proceedings of VLDB 3 (1) (2010) 638–649.
[14] M.A. Soliman, I.F. Ilyas, Ranking with uncertain scores, in: Proc. ICDE, 2009, pp. 317–328.
[15] R. Agrawal, R. Rantzau, E. Terzi, Context-sensitive ranking, in: Proc. SIGMOD, 2006, pp. 383–394.
[16] S. Brin, L. Page, The anatomy of a large-scale hypertextual web search engine, Computer Networks 30 (1–7) (1998) 107–117.
[17] J. Cohen, T. Hickey, Two algorithms for determining volumes of convex polyhedra, Journal of the Association for Computing Machinery 26 (3) (1979) 401–414.
[18] C. Wang, L. Y. Yuan, J.-H. You, A ranking theory for uncertain data with constraints, in: Proc. IEEE International Conference on Computer Science and Information Technology, vol. 4, 2009, pp. 104–108.