Chapter 2
Probabilistic Ranking Queries on Uncertain Data

In this chapter, we formulate the probabilistic ranking queries on uncertain data. We first introduce two basic uncertain data models and basic probabilistic ranking queries. Then, we discuss three extended uncertain data models that suit different application scenarios. Ranking queries on the extended uncertain data models are also developed.

Frequently used definitions and notations are listed in Table 2.1.

2.1 Basic Uncertain Data Models

We consider uncertain data in the *possible worlds* semantics model [23, 12, 24, 7], which has been extensively adopted by the recent studies on uncertain data processing, such as [17, 8, 21]. Technically, uncertain data can be represented in two ways.

2.1.1 Uncertain Object Model

An uncertain object O [18, 19, 20, 21] is conceptually governed by an underlying random variable X. Theoretically, if X is a continuous random variable, the distribution of X can be captured by a *probability density function* (*PDF* for short); if X is a discrete random variable, its distribution can be described by a *probability mass function* (*PMF* for short). In practice, the PDF or PMF of a random variable is often unavailable. Instead, a sample set of instances $x_1, \cdots, x_m$ are used to approximate the distribution of X, where each instance takes a membership probability. For an instance $x_i \in X$ $(1 \leq i \leq m)$, the membership probability of x_i measures the likelihood that x_i will occur. Due to the unavailability of X's PDF or PMF, in this book, we represent an uncertain object O using the set of samples $x_1, \cdots, x_m$ generated by the underlying random variable.

M. Hua, J. Pei, *Ranking Queries on Uncertain Data,* ADVANCES IN DATABASE SYSTEMS 42,
DOI 10.1007/978-1-4419-9380-9_2, © Springer Science+Business Media, LLC 2011

Notation	Description
$O = \{o_1, \cdots, o_m\}$	an uncertain object contains m instances
$\mathscr{O}$	a set of uncertain objects
$T = \{t_1, \cdots, t_n\}$	a table with n tuples
$R : t_{r_1} \oplus \cdots \oplus t_{r_m}$	a generation rule specifying the exclusiveness among $t_{r_1}, \cdots, t_{r_m}$
$\mathscr{R}$	a set of generation rules
W	a possible world
$\mathscr{W}$	a set of possible worlds
$O = o_1, o_2, \cdots$	an uncertain data stream
$W_\omega^t(\mathscr{O})$	a set of uncertain data streams in sliding window W_ω^t
$\mathscr{L}(t_A, t_B)$	a probabilistic linkage between tuples t_A and t_B
$G(V, E, W)$	a simple graph with probabilistic weights W

Table 2.1 Summary of definitions and frequently used notations.

Definition 2.1 (Uncertain object). An **uncertain object** is a set of instances $O = \{o_1, \cdots, o_m\}$ such that each instance o_i $(1 \leq i \leq m)$ takes a **membership probability** $Pr(o_i) > 0$, and $\sum_{i=1}^{m} Pr(o_i) = 1$. ■

The **cardinality** of an uncertain object $O = \{o_1, \cdots, o_m\}$, denoted by $|O|$, is the number of instances contained in O. We denote the set of all uncertain objects as $\mathscr{O}$.

2.1.1.1 Possible Worlds Semantics

In the basic uncertain object model, we assume that the distributions of uncertain objects are independent from each other. Correlations among uncertain objects are discussed in Section 2.3.2. The uncertain objects carry the possible worlds semantics.

Definition 2.2 (Possible worlds of uncertain objects). Let $\mathscr{O} = \{O_1, \cdots, O_n\}$ be a set of uncertain objects. A **possible world** $W = \{o_1, \cdots, o_n\}$ $(o_i \in O_i)$ is a set of instances such that one instance is taken from each uncertain object. The **existence probability** of W is $Pr(W) = \prod_{i=1}^{n} Pr(o_i)$. ■

Let $\mathscr{W}$ denote the set of all possible worlds, we have the following property.

Corollary 2.1 (Number of possible worlds). *For a set of uncertain objects $\mathscr{O} = \{O_1, \ldots, O_n\}$, let $|O_i|$ be the cardinality of object O_i $(1 \leq i \leq n)$, the number of all possible worlds is*

$$|\mathscr{W}| = \prod_{i=1}^{n} |O_i|.$$

Moreover,

$$Pr(\mathscr{W}) = \sum_{w \in \mathscr{W}} Pr(w) = 1$$

■

Example 2.1 (Uncertain objects). Table 1.2 is an example of a uncertain object with 6 instances. Each instance takes an equal membership probability $\frac{1}{6}$. ■

2.1.2 Probabilistic Database Model

In some other studies, the probabilistic database model is used to represent uncertain data. A probabilistic database [17] is a finite set of probabilistic tables defined as follows.

Definition 2.3 (Probabilistic table). A **probabilistic table** contains a set of uncertain tuples T and a set of generation rules $\mathscr{R}$. Each **uncertain tuple** $t \in T$ is associated with a **membership probability** $Pr(t) > 0$. Each **generation rule** (or **rule** for short) $R \in \mathscr{R}$ specifies a set of exclusive tuples in the form of $R : t_{r_1} \oplus \cdots \oplus t_{r_m}$ where $t_{r_i} \in T$ $(1 \leq i \leq m)$, $Pr(t_{r_i} \wedge t_{r_j}) = 0$ $(1 \leq i, j \leq m, i \neq j)$ and $\sum_{i=1}^{m} Pr(t_{r_i}) \leq 1$. ■

The probabilistic database model also follows the possible worlds semantics. The generation rule R constrains that, among all tuples $t_{r_1}, \cdots, t_{r_m}$ involved in the rule, at most one tuple can appear in a possible world. R is a *singleton rule* if there is only one tuple involved in the rule, otherwise, R is a *multi-tuple rule*. The **cardinality** of a generation rule R, denoted by $|R|$, is the number of tuples involved in R.

Definition 2.4 (Possible worlds of a probabilistic table). Given a probabilistic table T, a *possible world* W is a subset of T such that for each generation rule $R \in \mathscr{R}_T$, $|R \cap W| = 1$ if $Pr(R) = 1$, and $|R \cap W| \leq 1$ if $Pr(R) < 1$. The existence probability of W is

$$Pr(W) = \prod_{R \in \mathscr{R}_T, |R \cap W| = 1} Pr(R \cap W) \prod_{R \in \mathscr{R}_T, R \cap W = \emptyset} (1 - Pr(R))$$

■

Corollary 2.2 (Number of possible worlds). *For an uncertain table T with a set of generation rules $\mathscr{R}_T$, the number of all possible worlds is*

$$|\mathscr{W}| = \prod_{R \in \mathscr{R}_T, Pr(R) = 1} |R| \prod_{R \in \mathscr{R}_T, Pr(R) < 1} (|R| + 1)$$

■

Example 2.2 (Probabilistic tables). Table 1.1(a) is an example of a probabilistic table with 6 uncertain tuples and 2 multi-tuple generation rules $R2 \oplus R3$ and $R5 \oplus R6$. The corresponding possible worlds are shown in Table 1.1(b). ■

2.1.3 Converting Between the Uncertain Object Model and the Probabilistic Database Model

Interestingly, the uncertain object model and the probabilistic database model are equivalent.

- **Converting from the uncertain object model to the probabilistic database model.** A set of uncertain objects can be represented by a probabilistic table as follows. For each instance o of an uncertain object O, we create a tuple t_o, whose membership probability is $f(o)$. For each uncertain object $O = \{o_1, \cdots, o_m\}$, we create one generation rule $R_O = t_{o_1} \oplus \cdots \oplus t_{o_m}$.
- **Converting from the probabilistic database model to the uncertain object model.** A probabilistic table can be represented by a set of uncertain objects with discrete instances. For each tuple t in a probabilistic table, we create an instance o_t, whose probability mass function is $f(o_t) = Pr(t)$. For a generation rule $R : t_{r_1} \oplus \cdots \oplus t_{r_m}$, we create an uncertain object O_R, which includes instances $o_{t_{o_1}}, \cdots, o_{t_{o_m}}$ corresponding to $t_{r_1}, \cdots, t_{r_m}$, respectively. Moreover, if $\sum_{i=1}^{m} Pr(t_{r_i}) < 1$, we create another instance $o_\emptyset$ whose probability mass function is $f(o_\emptyset) = 1 - \sum_{i=1}^{m} Pr(t_{r_i})$, and add $u_\emptyset$ to the uncertain object O_R.

Example 2.3 (Converting between two models). $R1$ in Table 1.1(a) can be converted to an uncertain object $O_1 = \{R1, \neg R1\}$ where $Pr(R1) = 0.3$ and $Pr(\neg R1) = 0.7$. Moreover, generation rule $R2 \oplus R3$ in Table 1.1(a) can be converted to uncertain object $O_{1,2} = \{R2, R3, \neg R23\}$ where $Pr(R2) = 0.4$, $Pr(R3) = 0.5$ and $Pr(\neg R23) = 0.1$. ■

2.2 Basic Ranking Queries on Uncertain Data

In this section, we discuss various types of ranking queries on the uncertain object model. Since the uncertain object model and the probabilistic database model are equivalent, the queries discussed in this section can also be applied to the probabilistic database model.

Depending on different application scenarios, probabilistic ranking queries can be applied at one of the three *granularity levels*.

- The *instance* probabilistic ranking queries return the instances satisfying query conditions. We develop two classes of instance probabilistic ranking queries. The first are *top-k typicality queries*, which rank instances in an uncertain object according to how typical each instance is. The second are *probabilistic ranking queries*, which rank instances in multiple objects according to the probability that each instance is ranked top-k. The two classes of queries will be discussed in Sections 2.2.1 and 2.2.2, respectively.
- The *object* probabilistic ranking queries find the object satisfying query conditions, which will be discussed in Section 2.2.3.

- The *object set* probabilistic ranking queries apply the query condition to each object set and return the object set that satisfy the query. We defer the discussion on ranking uncertain object sets to Section 2.3.3 in the context of uncertain road networks.

2.2.1 Ranking Instances in An Uncertain Object

Given an uncertain object with a large number of instances that are samples taken from an underlying random variable, how can we understand and analyze this object? An effective way is to find the most typical instances among all instances of the uncertain object. We develop a class of top-k typicality queries which can serve for this purpose.

Example 2.4 (Top-k typicality queries).

Jeff is a junior basketball player who dreams to play in the NBA. As the NBA has more than 400 *active players, they are quite diverse. Jeff may want to know some representative examples of NBA players. Top-k typicality queries can help.*

We can model the group of NBA players as an uncertain object in the space of technical statistics, which can be described by a likelihood function. Each player is an instance of the uncertain object.

- ***Top-**k **simple typicality queries.***
  *Jeff asks, "*Who are the top-3 most typical NBA players?*" The player who has the maximum likelihood of being NBA players is the most typical. This leads to our first typicality measure – the* simple typicality. *A top-k simple typicality query finds the k most typical instances in an uncertain object.*
- ***Top-**k **discriminative typicality queries.***
  *Jeff is particularly interested in becoming a guard. "*Who are the top-3 most typical guards distinguishing guards from other players?*" Simple typicality on the set of guards is insufficient to answer the question, since it is possible that a typical guard may also be typical among other players. Instead, players that are typical among all guards but are not typical among all non-guard players should be found.*

  In order to address this demand, we can model the group of guards as a target uncertain object O_g and the set of other players as the other uncertain object O. The notion of discriminative typicality *measures how an instance is typical in one object but not typical in the other object. Given two uncertain objects O and S, let O be the target object, a top-k discriminative typicality query finds the k instances with the highest discriminative typicality values in O.*
- ***Top-**k **representative typicality queries.***
  NBA guards may still have some sub-groups. For example, the fresh guards and the experienced guards, as well as the shooting guards and the point guards. Jeff wants to learn different types of guards, without a clear idea about what types

there are. So he asks, "Who are the top-3 typical guards in whole representing different types of guards?"

Simple typicality does not provide the correct answer to this question, since the 3 *players with the greatest simple typicality may be quite similar to each other, while some other popular players different from those three may be missed. Discriminative typicality does not help either, because the exact types of guards and their members are unknown.*

To solve this problem, we develop the notion of representative typicality *that measures how an instance is typical in an uncertain object different from the already reported typical instances. Given an uncertain object O, a top-k representative typicality query finds a set of k instances of O with the highest representative typicality scores.* ■

By default, we consider an uncertain object O on attributes $A_1, \cdots, A_n$. Let $A_{i_1}, \cdots, A_{i_l}$ be the attributes on which the typicality queries are applied ($1 \leq i_j \leq n$ for $1 \leq j \leq l$) and $d_{A_{i_1}, \cdots, A_{i_l}}(x, y)$ be the distance between two instances x and y in S on attributes $A_{i_1}, \cdots, A_{i_l}$. When $A_{i_1}, \cdots, A_{i_l}$ are clear from context, $d_{A_{i_1}, \cdots, A_{i_l}}(x, y)$ is abbreviated to $d(x, y)$.

We address the top-k typicality problem in a generic metric space. Therefore, the distance metric d should satisfy the triangle inequality.

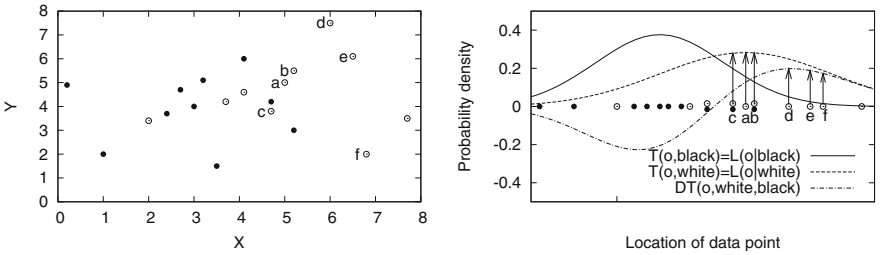
2.2.1.1 Simple Typicality

By intuition and as also suggested by the previous research in psychology and cognitive science (as will be reviewed in Section 3.3.1), an instance o in O is more typical than the others if o is more likely to appear in O. As discussed in Section 2.1.1, the set of instances in O on attributes $A_1, \cdots, A_n$ can be viewed as a set of independent and identically distributed samples of an n-dimensional random vector $\mathscr{X}$ that takes values in the Cartesian product space $D = D_{A_1} \times \cdots \times D_{A_n}$, where D_{A_i} is the domain of attribute A_i ($1 \leq i \leq n$). The likelihood of $o \in O$, given that o is a sample of $\mathscr{X}$, can be used to measure the typicality of o.

Definition 2.5 (Simple typicality). Given an uncertain object O on attributes $A_1, \cdots, A_n$ and a subset of attributes $A_{i_1}, \cdots, A_{i_l}$ ($1 \leq i_j \leq n$ for $1 \leq j \leq l$) of interest, let $\mathscr{X}$ be the n-dimensional random vector generating the instances in O, the **simple typicality** of an instance $o \in O$ with respect to $\mathscr{X}$ on attributes $A_{i_1}, \cdots, A_{i_l}$ is defined as $T_{A_{i_1}, \cdots, A_{i_l}}(o, \mathscr{X}) = L_{A_{i_1}, \cdots, A_{i_l}}(o|\mathscr{X})$ where $L_{A_{i_1}, \cdots, A_{i_l}}(o|\mathscr{X})$ is the likelihood [27] of o on attributes $A_{i_1}, \cdots, A_{i_l}$, given that o is a sample of $\mathscr{S}$. ■

In practice, since the distribution of random vector $\mathscr{X}$ is often unknown, we use $T_{A_{i_1}, \cdots, A_{i_l}}(o, O) = L_{A_{i_1}, \cdots, A_{i_l}}(o|O)$ as an estimator of $T_{A_{i_1}, \cdots, A_{i_l}}(o, \mathscr{X})$, where $L_{A_{i_1}, \cdots, A_{i_l}}(o|O)$ is the posterior probability of an object o on attributes $A_{i_1}, \cdots, A_{i_l}$ given O [27].

$L_{A_{i_1}, \cdots, A_{i_l}}(o|O)$ can be computed using density estimation methods. We adopt the commonly used kernel density estimation method, which does not require any distribution assumption on O. The general idea is to use a kernel function to approximate



(a) A set of points. (b) Typicality.

Fig. 2.1 The simple typicality and discriminative typicality curves of a set of points.

the probability density around each observed sample. More details will be discussed in Chapter 4.

Hereafter, unless specified otherwise, the simple typicality measure refers to the estimator $T_{A_{i_1},\cdots,A_{i_l}}(o,O)$. Moreover, for the sake of simplicity, when $A_{i_1},\cdots,A_{i_l}$ are clear from context, $T_{A_{i_1},\cdots,A_{i_l}}(o,O)$ and $L_{A_{i_1},\cdots,A_{i_l}}(o|O)$ are abbreviated to $T(o,O)$ and $L(o|O)$, respectively.

Given an uncertain object O on attributes $A_{i_1},\cdots,A_{i_l}$ of interest, a predicate P and a positive integer k, a **top-k simple typicality query** returns, from the set of instances in O satisfying predicate P, the k instances having the largest simple typicality values that are computed on attributes $A_{i_1},\ldots,A_{i_l}$.

*Example 2.5 (Top-k simple typicality queries). Consider the set of points belong to an uncertain object in Figure 2.1(a). A top-*3 *simple typicality query on attribute X with predicate COLOR = white returns the* 3 *white points having the largest simple typicality values computed on attribute X.*

Figure 2.1(b) projects the points in T to attribute X. The likelihood function of the white points and that of the black points on attribute X are labeled as $L(o|white)$ and $L(o|black)$ in the figure, respectively, while we will discuss how to compute the likelihood values in Chapter 4. Points a, b and c have the highest likelihood values among all white points, and thus should be returned as the answer to the query. ■

2.2.1.2 Discriminative Typicality

Given two uncertain objects O and S, which instance is the most typical in O but not in S? We use the discriminative typicality to answer such a question. By intuition, an instance $o \in O$ is typical and discriminative in O if the difference between its typicality in O and that in S is large.

Definition 2.6 (Discriminative typicality). Given two uncertain objects O and S on attributes $A_1,\cdots,A_n$ (O is the **target object**), let $\mathscr{U}$ and $\mathscr{V}$ be the n-dimensional random vectors generating the instances in O and S, respectively, the **discriminative**

typicality of an instance $o \in O$ on attributes $A_{i_1}, \cdots, A_{i_l}$ ($1 \leq i_j \leq n$ for $1 \leq j \leq l$) is $DT(o, \mathscr{U}, \mathscr{V}) = T(o, \mathscr{U}) - T(o, \mathscr{V})$, where $T(o, \mathscr{U})$ and $T(o, \mathscr{V})$ are the simple typicality values of instance o with respect to $\mathscr{U}$ and $\mathscr{V}$, respectively. ■

In the definition, the discriminative typicality of an instance is defined as the difference of its simple typicality in the target object and that in the rest of the data set. One may wonder whether using the ratio $\frac{T(o,\mathscr{U})}{T(o,\mathscr{V})}$ may also be meaningful. Unfortunately, such a ratio-based definition may not choose a typical instance that has a large simple typicality value with respect to $\mathscr{U}$. Consider an extreme example. Let o be an instance that is very atypical with respect to $\mathscr{U}$ and has a typicality value of nearly 0 with respect to $\mathscr{V}$. Then, o still has an infinite ratio $\frac{T(o,\mathscr{U})}{T(o,\mathscr{V})}$. Although o is discriminative between $\mathscr{U}$ and $\mathscr{V}$, it is not typical with respect to $\mathscr{U}$ at all.

Due to the unknown distribution of random vectors $\mathscr{U}$ and $\mathscr{V}$, we use $DT(o, O, S) = T(o, O) - T(o, S)$ to estimate $DT(o, \mathscr{U}, \mathscr{V})$, where $T(o, O)$ and $T(o, S)$ are the estimators of $T(o, \mathscr{U})$ and $T(o, \mathscr{V})$, respectively.

Given a set of uncertain instances on attributes $A_{i_1}, \ldots, A_{i_l}$ of interest, a predicate P and a positive integer k, a **top-k discriminative typicality query** treats the set of instances satisfying P as the target object, and returns the k instances in the target object having the largest discriminative typicality values computed on attributes $A_{i_1}, \ldots, A_{i_l}$.

Example 2.6 (Top-k discriminative typicality queries). Consider the set of points in Figure 2.1(a) again and a top-3 discriminative typicality query on attribute X with predicate COLOR = white.

The discriminative typicality $DT(o, white, black)$ for each instance $o \in white$ is plotted in the figure, where white and black denote the two uncertain objects, the one with white points as instances and the one with black points as instances, respectively. To see the difference between discriminative typicality and simple typicality, consider instance a, b and c, which have large simple typicality values among all white points. However, they also have relatively high simple typicality values as a member in the subset of black points comparing to other white points. Therefore, they are not discriminative. Points $\{d, e, f\}$ are the answer to the query, since they are discriminative. ■

2.2.1.3 Representative Typicality

The answer to a top-k simple typicality query may contain some similar instances, since the instances with similar attribute values may have similar simple typicality scores. However, in some situations, it is redundant to report many similar instances. Instead, a user may want to explore the uncertain object by viewing typical instances that are different from each other but jointly represent the uncertain object well.

Suppose a subset of instances $A \subset O$ is chosen to represent O. Each instances in $(O - A)$ is best represented by the closest instance in A. For each $o \in A$, we define the representing region of o.

Definition 2.7 (Representing region). Given an uncertain object O on attributes $A_1, \cdots, A_n$ and a subset of instances $A \subset O$, let $D = D_{A_1} \times \cdots \times D_{A_n}$ where D_{A_i} is the domain of attribute A_i ($1 \leq i \leq n$), the **representing region** of an instance $o \in A$ is $D(o,A) = \{x | x \in D, d(x,o) = \min_{y \in A} d(x,y)\}$, where $d(x,y)$ is the distance between objects x and y. ■

To make A representative as a whole, the representing region of each instance o in A should be fairly large and o should be typical in its own representing region.

Definition 2.8 (Group typicality). Given an uncertain object O on attributes $A_1, \cdots, A_n$ and a subset of instances $A \subset O$, let $\mathscr{X}$ be the n-dimensional random vector generating the instances in O, the **group typicality** of A on attributes $A_{i_1}, \cdots, A_{i_l}$ ($1 \leq i_j \leq n$, $1 \leq j \leq l$) is $GT(A, \mathscr{X}) = \sum_{o \in A} T(o, \mathscr{X}_{D(o,A)}) \cdot Pr(D(o,A))$, where $T(o, \mathscr{X}_{D(o,A)})$ is the simple typicality of o with respect to $\mathscr{X}$ in o's representing region $D(o,A)$ and $Pr(D(o,A))$ is the probability of $D(o,A)$. ■

Since the distribution of $\mathscr{X}$ is unknown, we can estimate the group typicality $GT(A, \mathscr{X})$ as follows. For any instance $o \in A$, let $N(o,A,O) = \{x | x \in O \cap D(o,A)\}$ be the set of instances in O that lie in $D(o,A)$, $Pr(D(o,A))$ can be estimated using $\frac{|N(o,A,O)|}{|O|}$. The group typicality $GT(A, \mathscr{X})$ is estimated by $GT(A,O) = \sum_{o \in A} T(o, N(o,A,O)) \cdot \frac{|N(o,A,O)|}{|O|}$, where $T(o, N(o,A,O))$ is the estimator of simple typicality $T(o, \mathscr{X}_{D(o,A)})$, since $N(o,A,O)$ can be viewed as a set of independent and identically distributed samples of $\mathscr{X}$ that lie in $D(o,A)$.

The group typicality score measures how representative a group of instances is. The *size-k most typical group problem* is to find k instances as a group such that the group has the maximum group typicality. Unfortunately, the problem is NP-hard, since it has the discrete k-median problem as a special case, which was shown to be NP-hard [28].

Moreover, top-k queries are generally expected to have the monotonicity in answer sets. That is, the result of a top-k query is contained in the result of a top-k' query where $k < k'$. However, an instance in the most typical group of size k may not be in the most typical group of size k' ($k < k'$). For example, in the data set illustrated in Figure 2.2, the size-1 most typical group is $\{A\}$ and the size-2 most typical group is $\{B,C\}$, which does not contain the size-1 most typical group. Therefore, the size-k most typical group is not suitable to define the top-k representative typicality.

To enforce monotonicity, we adopt a greedy approach.

Definition 2.9 (Representative typicality). Given an uncertain object O and a **reported answer set** $A \subset O$, let $\mathscr{X}$ be the random vector with respect to instances in O, the **representative typicality** of an instance $o \in (O - A)$ is $RT(o,A, \mathscr{X}) = GT(A \cup \{o\}, \mathscr{X}) - GT(A, \mathscr{X})$, where $GT(A \cup \{o\}, \mathscr{X})$ and $GT(A, \mathscr{X})$ are the group typicality values of subsets $A \cup \{o\}$ and A, respectively. ■

In practice, we use $RT(o,A,O) = GT(A \cup \{o\}, O) - GT(A,O)$ to estimate $RT(o,A, \mathscr{X})$, where $GT(A,O)$ and $GT(A \cup \{o\}, O)$ are the estimators of $GT(A, \mathscr{X})$ and $GT(A \cup \{o\}, \mathscr{X})$, respectively.

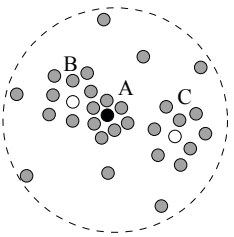


Fig. 2.2 Non-monotonicity of size-k most typical group.

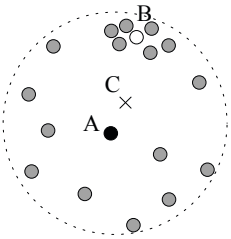
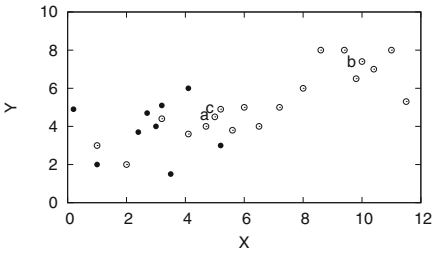
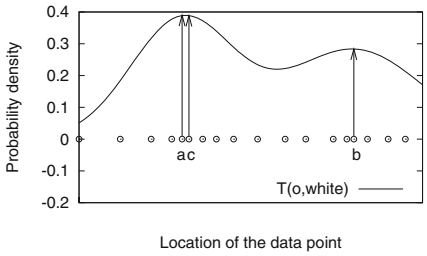


Fig. 2.3 Medians, means and typical objects.



(a) A set of points.



(b) Representative typicality.

Fig. 2.4 The answer to a top-2 representative typicality query on a set of points.

Given an uncertain object O on attributes $A_{i_1},\ldots,A_{i_l}$ of interest, a predicate P and a positive integer k, a **top-k representative typicality query** returns k instances $o_1,\ldots,o_k$ from the set of instances in O satisfying predicate P, such that o_1 is the instance having the largest simple typicality, and, for $i > 1$,

$$o_i = \arg \max_{o \in O - \{o_1,\ldots,o_{i-1}\}} RT(o, \{o_1,\ldots,o_{i-1}\}, O).$$

The representative typicality values are computed on attributes $A_{i_1},\ldots,A_{i_l}$.

Example 2.7 (Top-k representative typicality queries). Consider the set of points in Figure 2.4(a) and a top-2 representative typicality query on attribute X with predicate COLOR = white.

We project the white points to attribute X and plot the simple typicality scores of the white points, as shown in Figure 2.4(b). Points a and c have the highest simple typicality scores. However, if we only report a and c, then the dense region around a is reported twice, but the dense region around b is missed. A top-2 representative typicality query will return a and b as the answer. ■

2.2.2 Ranking Uncertain Instances in Multiple Uncertain Objects

Given multiple uncertain objects, how can we select a small subset of instances meeting users' interests? Ranking queries (also known as top-k queries) [3, 4, 5, 6] are a class of important queries in data analysis that allows us to select the instances ranked top according to certain user specified scoring functions. We consider the top-k selection query model [29].

Definition 2.10 (Top-k selection queries). For a set of instances S, each instance $o \in S$ is associated with a set of attributes A. Given a predicate P on A, a ranking function $f : S \to R$ and a integer $k > 0$, a **top-k selection query** $Q^k_{P,f}$ returns a set of instances $Q^k_{P,f}(S) \subseteq S_P$, where S_P is the set of instances satisfying P, $|Q^k_{P,f}(S)| = \min\{k, |S_P|\}$ and $f(o) > f(o')$ for any instances $o \in Q^k_{P,f}(S)$ and $o' \in S_P - Q^k_{P,f}(S)$.■

To keep our presentation simple, we assume that the top-k selection queries in our discussion select all instances in question. That is $S_P = S$. Those selection predicates can be implemented efficiently as filters before our ranking algorithms are applied. Moreover, we assume that the ranking function f in a top-k selection query can be efficiently applied to an instance o to generate a score $f(o)$. When it is clear from context, we also write $Q^k_{P,f}$ as Q^k for the sake of simplicity.

2.2.2.1 Ranking Probabilities

How can we apply a top-k selection query to a set of uncertain objects? Since each object appears as a set of instances, we have to rank the instances in the possible worlds semantics.

A top-k selection query can be directly applied to a possible world that consists of a set of instances. In a possible word, a top-k selection query returns k instances. We define the rank-k probability and top-k probability for instances and objects as follows.

Given a set of uncertain objects and a *ranking function* f, all instances of the uncertain objects can be ranked according to the ranking function. For instances o_1 and o_2, $o_1 \preceq_f o_2$ if o_1 is ranked higher than or equal to o_2 according to f. The *ranking order* $\preceq_f$ is a total order on all instances.

Definition 2.11 (rank-k Probability and top-k probability). For an instance o, the **rank-k probability** $Pr(o,k)$ is the probability that o is ranked at the k-th position in possible worlds according to f, that is

$$Pr(o,k) = \sum_{W \in \mathscr{W} \text{ s.t. } o = W_f(k)} Pr(W) \tag{2.1}$$

where $W_f(k)$ denotes the instance ranked at the k-th position in W.

The **top-k probability** $Pr^k(o)$ is the probability that o is ranked top-k in possible worlds according to f, that is,

$$Pr^k(o) = \sum_{j=1}^{k} Pr(o, j). \tag{2.2}$$

■

2.2.2.2 Ranking Criteria

Given a *rank parameter* $k > 0$ and a probability threshold $p \in (0,1]$, a *probability threshold top-k query* (*PT-k query* for short) [30, 31] finds the instances whose top-k probabilities are no less than p.

Definition 2.12 (PT-k query and top-(k,l) query). Given a *rank parameter* $k > 0$ and a probability threshold $p \in (0,1]$, a **probabilistic threshold top-k query** (*PT-k query* for short) [30, 31] finds the instances whose top-k probabilities are no less than p.

Alternatively, a user can use an *answer set size constraint* $l > 0$ to replace the probability threshold p and issue a **top-(k,l) query** [32, 33], which finds the top-l tuples with the highest top-k probabilities. ■

Now, let us consider the reverse queries of PT-k queries and top-(k,l) queries.

For an instance o, given a probability threshold $p \in (0,1]$, the *p-rank* of o is the minimum k such that $Pr^k(o) \geq p$, denoted by $MR_p(o) = \min\{k | Pr^k(o) \geq p\}$.

Definition 2.13 (RT-k query and top-(p,l) query). Given a *probability threshold* $p \in (0,1]$ and a *rank threshold* $k > 0$, a **rank threshold query** (*RT-k query* for short) to retrieve the instances whose p-ranks are at most k. RT-k queries are reverse queries of PT-k queries.

Alternatively, a user can replace the rank threshold by an *answer set size constraint* $l > 0$ and issue a **top-(p,l) query**, which returns the top-l instances with the smallest p-ranks. Clearly, top-(p,l) queries are reverse queries of top-(k,l) queries. ■

Interestingly, it is easy to show the following.

Corollary 2.3 (Answers to PT-k and RT-k queries). *Given a set of uncertain objects S, an integer $k > 0$ and a real value $p \in (0,1]$, the answer to a PT-k query with rank parameter k and probability threshold p and that to a RT-k query with rank threshold k and probability threshold p are identical.*

Proof. An instance satisfying the PT-k query must have the p-rank at most k, and thus satisfies the RT-k query. Similarly, an instance satisfying the RT-k query must have the top-k probability at least p, and thus satisfies the PT-k query. ■

PT-k queries and RT-k queries share the same set of parameters: a rank parameter k and a probability threshold. Thus, as shown in Corollary 2.3, when the parameters are the same, the results are identical. For top-(k,l) queries and top-(p,l) queries, even they share the same value on the answer set size constraint l, the answers generally may not be the same since the rank parameter and the probability threshold select different instances.

TID	Rank	Prob.	Top-k probabilities			
			$k=1$	$k=2$	$k=3$	$k=4$
o_1	1	0.5	0.5	0.5	0.5	0.5
o_2	2	0.3	0.15	0.3	0.3	0.3
o_3	3	0.7	0.245	0.595	0.7	0.7
o_4	4	0.9	0.0945	0.45	0.8055	0.9

Table 2.2 Top-k probabilities of a set of tuples.

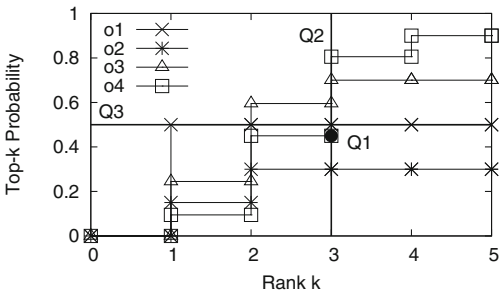


Fig. 2.5 Ranking queries on uncertain tuples.

Example 2.8 (PT-k query and Top-(k,l) query). Consider a set of uncertain instances in Table 2.2. Suppose each instance belongs to one uncertain object and all objects are independent. In Figure 2.5, we plot the top-k probabilities of all instances with respect to different values of k.

A PT-3 query with probability threshold $p = 0.45$ returns instances $\{o_1, o_3, o_4\}$ whose top-3 probabilities are at least 0.45. Interestingly, the PT-3 query with probability threshold $p = 0.45$ can be represented as a point $Q1(3, 0.45)$ in Figure 2.5. As the answers to the query, the top-k probability curves of o_1, o_3 and o_4 lie northeast to $Q1$.

Alternatively, a top-(k,l) query with $k = 3$ and $l = 2$ returns 2 instances $\{o_4, o_3\}$, which have the highest top-3 probabilities. The query can be represented as a vertical line $Q2(k = 3)$ in Figure 2.5. The answer set includes the 2 curves which have the highest intersection points with $Q2$. ■

Example 2.9 (RT-k query and Top-(p,l) query). Consider the uncertain instances in Table 2.2 again. An RT-3 query with probability threshold $p = 0.45$ returns $\{o_1, o_3, o_4\}$. The answer is the same as the answer to the PT-3 query with the same probability threshold as shown in Example 2.8.

A top-(p,l) query with $p = 0.5$ and $l = 2$ returns $\{o_1, o_3\}$ whose 0.5-ranks are the smallest. The query can be represented as a horizontal line $Q3(probability = 0.5)$ in Figure 2.5. The 2 curves having the leftmost intersections with $Q3$ are the answers.■

2.2.3 Ranking Uncertain Objects

At the object level, the rank-k probability and top-k probability are defined as follows.

Definition 2.14 (Object rank-k probability and top-k probability). For an uncertain object O, the **object rank-k probability** $Pr(O,k)$ is the probability that any instance $o \in O$ is ranked at the k-th position in possible worlds according to f, that is

$$Pr(O,k) = \sum_{o \in O} Pr(o,k). \tag{2.3}$$

The **object top-k probability** $Pr^k(O)$ is the probability that any instance in O is ranked top-k in possible worlds, that is

$$Pr^k(O) = \sum_{o \in O} Pr^k(O) = \sum_{o \in O} \sum_{j=1}^{k} Pr(o,j). \tag{2.4}$$

■

The probabilistic ranking queries discussed in Section 2.2.2 can be applied at the object level straightforwardly. Therefore, we skip the definitions of those queries.

2.3 Extended Uncertain Data Models and Ranking Queries

In this section, we develop three extended uncertain data models and ranking queries on those models, to address different application interest.

2.3.1 Uncertain Data Stream Model

As illustrated in Scenario 2 of Example 1.1, the instances of an uncertain object may keep arriving in fast pace and thus can be modeled as a data stream. The instances are generated by an underlying *temporal random variable* whose distribution evolves over time. To keep our discussion simple, we assume a synchronous model. That is, each time instant is a positive integer, and at each time instant t ($t > 0$), an instance is collected for an uncertain data stream. To approximate the current distribution of a temporal random variable, practically we often use the observations of the variable in a recent time window as the sample instances.

Definition 2.15 (Uncertain data stream, sliding window).

An **uncertain data stream** is a (potentially infinite) series of instances $O = o_1, o_2, \cdots$. Given a time instants t ($t > 0$), $O[t]$ is the instance of stream O.

A **sliding window** W_ω^t is a selection operator defined as $W_\omega^t(O) = \{O[i] | (t-\omega) < i \leq t\}$, where $\omega > 0$ is called the **width** of the window.

For a set of uncertain data streams $\mathscr{O} = \{O_1, \cdots, O_n\}$, sliding window $W_\omega^t(\mathscr{O}) = \{W_\omega^t(O_i) | 1 \leq i \leq n\}$. ■

2.3.1.1 Connections with the Uncertain Object Model

The distribution of an uncertain data stream O in a given sliding window W_ω^t is static. Thus, the set of instances $W_\omega^t(O)$ can be considered as an uncertain object. The membership probabilities for instances depend on how the instances are generated from the underlying random variable of $W_\omega^t(O)$. For example, if the instances are drawn using simple random sampling [34], then all instances take the same probability $\frac{1}{\omega}$. On the other hand, using other techniques like particle filtering [35] can generate instances with different membership probabilities. In this book, we assume that the membership probabilities of all instances are identical. Some of our developed methods can also handle the case of different membership probabilities, which will be discussed in Section 6.5.

Definition 2.16 (Uncertain object in a sliding window). Let O be an uncertain data stream. At time instant $t > 0$, the set of instances of O in a sliding window W_ω^t is an uncertain object denoted by $W_\omega^t(O)$ $(1 \leq i \leq n)$, where each instant $o \in W_\omega^t(O)$ has the membership probability $Pr(o) = \frac{1}{\omega}$. ■

In this book, we assume that the distributions of uncertain data streams are independent from each other. Handling correlations among uncertain data streams is an important direction that we plan to investigate as future study that will be discussed in Section 6.5. The uncertain data in a sliding window carries the possible worlds semantics.

Definition 2.17 (Possible worlds of uncertain data streams). Let $\mathscr{O} = \{O_1, \ldots, O_n\}$ be a set of uncertain data streams. A **possible world** $w = \{v_1, \ldots, v_n\}$ in a sliding window W_ω^t is a set of instances such that one instance is taken from the uncertain object of each stream in W_ω^t, i.e., $v_i \in W_\omega^t(O_i)$ $(1 \leq i \leq n)$. The **existence probability** of w is $Pr(w) = \prod_{i=1}^{n} Pr(v_i) = \prod_{i=1}^{n} \frac{1}{\omega} = \omega^{-n}$.

The complete set of possible worlds of sliding window $W_\omega^t(\mathscr{O})$ is denoted by $\mathscr{W}(W_\omega^t(\mathscr{O}))$. ■

Corollary 2.4 (Number of possible worlds). *For a set of uncertain data streams $\mathscr{O} = \{O_1, \ldots, O_n\}$ and a sliding window $W_\omega^t(\mathscr{O})$, the total number of possible worlds is $|\mathscr{W}(W_\omega(t))| = \omega^n$.* ■

When it is clear from the context, we write $\mathscr{W}(W_\omega^t(\mathscr{O}))$ as $\mathscr{W}$ and $W_\omega^t(\mathscr{O})$ as W or W^t for the sake of simplicity.

Time instant	# Time	Speeds at A	Speeds at B	Speeds at C	Speeds at D
$t-2$	00 : 01 : 51	$a_1 = 15$	$b_1 = 6$	$c_1 = 14$	$d_1 = 4$
$t-1$	00 : 16 : 51	$a_2 = 16$	$b_2 = 5$	$c_2 = 8$	$d_2 = 7$
t	00 : 31 : 51	$a_3 = 13$	$b_3 = 1$	$c_3 = 2$	$d_3 = 10$
$t+1$	00 : 46 : 51	$a_4 = 11$	$b_4 = 6$	$c_4 = 9$	$d_4 = 3$
$\cdots$					

Table 2.3 An uncertain data stream. (Sliding window width $\omega = 3$. W_3^t contains time instant $t-2$, $t-1$ and t. W_3^{t+1} contains time instant $t-1$, t and $t+1$.)

Example 2.10 (Uncertain streams).

As discussed in Example 1.1, speed sensors are deployed to monitor traffic in a road network. The vehicle speed at each monitoring point can be modeled as a ***temporal random variable****. To capture the distribution of such a temporal random variable, a speed sensor at the monitoring point reports the speed readings every* 30 *seconds. Therefore, the speed readings reported by each speed sensor is an* ***uncertain stream****. Each reading is an* ***instance*** *of the stream. A* ***sliding window*** *of length* 3 *at time t contains the last* 3 *readings (that is, the readings in the last* 90 *seconds) of each speed sensor.*

Suppose there are four monitoring points A, B, C and D with speed readings shown in Table 2.3. At time t, sliding window W_3^t contains the records of speeds at time $t-2$, $t-1$ and t. $W_3^t(A) = \{a_1, a_2, a_3\}$ can be modeled as an ***uncertain object****. So are $W_3^t(B)$, $W_3^t(C)$ and $W_3^t(D)$. Each instance in W_3^t takes* ***membership probability*** $\frac{1}{3}$*. There are $3^4 = 81$ possible worlds. Each possible world takes one instance from each object. For example, $\{a_1, b_3, c_2, d_1\}$ is a possible world. The existence probability of each possible world is $(\frac{1}{3})^4 = \frac{1}{81}$.* ■

2.3.1.2 Continuous Probabilistic Threshold Top-k Queries

Probabilistic threshold top-k queries can be applied on a sliding window of multiple uncertain data streams. We treat the instances of an uncertain data stream falling into the current sliding window as an uncertain object, and rank the streams according to their current sliding window.

Definition 2.18 (Continuous probabilistic threshold top-k query). Given a probabilistic threshold top-k query Q_p^k, a set of uncertain data streams $\mathscr{O}$, and a sliding window width ω, the **continuous probabilistic threshold top-k query** is to, for each time instant t, report the set of uncertain data streams whose top-k probabilities in the sliding window $W_\omega^t(\mathscr{O})$ are at least p. ■

Example 2.11 (Continuous Probabilistic Threshold Top-k Queries). Consider the uncertain streams in Table 2.3 with sliding window size $\omega = 3$ and continuous probabilistic threshold top-2 query with threshold $p = 0.5$.

At time instant t, the sliding window contains uncertain objects $W_3^t(A)$, $W_3^t(B)$, $W_3^t(C)$ and $W_3^t(D)$. The top-k probabilities of those uncertain objects are:

$Pr^2(W_3^t(A)) = 1$, $Pr^2(W_3^t(B)) = \frac{2}{27}$, $Pr^2(W_3^t(C)) = \frac{5}{9}$ and $Pr^2(W_3^t(D)) = \frac{10}{27}$. Therefore, the probabilistic threshold top-k query returns $\{A, C\}$ at time instant t.

At time instant $t+1$, the top-k probabilities of the uncertain objects are: $Pr^2(W_3^{t+1}(A)) = 1$, $Pr^2(W_3^{t+1}(B)) = \frac{2}{27}$, $Pr^2(W_3^{t+1}(C)) = \frac{4}{9}$ and $Pr^2(W_3^{t+1}(D)) = \frac{13}{27}$. The probabilistic threshold top-k query returns $\{A\}$ at time instant $t+1$.

The methods of answering probabilistic threshold top-k queries will be discussed in Chapter 6.

2.3.2 Probabilistic Linkage Model

In the basic uncertain object model, we assume that each instance belongs to a unique object, though an object may have multiple instances. It is interesting to ask what if an instance may belong to different objects in different possible worlds. Such a model is useful in probabilistic linkage analysis, as shown in the following example.

Example 2.12 (Probabilistic linkages). Survival-after-hospitalization is an important measure used in public medical service analysis. For example, to obtain the statistics about the death population after hospitalization, Svartbo et al. [36] study survival-after-hospitalization by linking two real data sets, the hospitalization registers and the national causes-of-death registers in some counties in Sweden. Such technique is called record linkage [37], which finds the linkages among data entries referring to the same real-world entities from different data sources. However, in real applications, data is often incomplete or ambiguous. Consequently, record linkages are often uncertain.

Probabilistic record linkages are often used to model the uncertainty. For two records, a state-of-the-art probabilistic record linkage method [37, 38] can estimate the probability that the two records refer to the same real-world entity. To illustrate, consider some synthesized records in the two data sets as shown in Table 2.4. The column probability is calculated by a probability record linkage method.

Two thresholds δ_M and δ_U are often used $(0 \leq \delta_U < \delta_M \leq 1)$: when the linkage probability is less than δ_U, the records are considered not-matched; when the linkage probability is between δ_U and δ_M, the records are considered possibly matched; and when the linkage probability is over δ_M, the records are considered matched. Many previous studies focus on building probabilistic record linkages effectively and efficiently.

If a medical doctor wants to know, between John H. Smith *and* Johnson R. Smith*, which patient died at a younger age. The doctor can set the two thresholds $\delta_M = 0.4$ and $\delta_U = 0.35$ and compare the matched pairs of records. Suppose $\delta_M = 0.4$ and $\delta_U = 0.35$, then* John H. Smith *is matched to* J. Smith*, whose age is* 61*, and* Johnson R. Smith *is matched to* J. R. Smith*, whose age is* 45*. Therefore, the medical doctor concludes that* Johnson R. Smith *died at a younger age than* John H. Smith*. Is the answer correct?*

LID	hospitalization registers			causes-of-death registers			Probability
	Id	Name	Disease	Id	Name	Age	
l_1	a_1	John H. Smith	Leukemia	b_1	Johnny Smith	32	0.3
l_2	a_1	John H. Smith	Leukemia	b_2	John Smith	35	0.3
l_3	a_1	John H. Smith	Leukemia	b_3	J. Smith	61	0.4
l_4	a_2	Johnson R. Smith	Lung cancer	b_3	J. Smith	61	0.2
l_5	a_2	Johnson R. Smith	Lung cancer	b_4	J. R. Smith	45	0.8

Table 2.4 Record linkages between the hospitalization registers and the causes-of-death registers.

If we consider all possible worlds corresponding to the set of linkages shown in Table 2.4 (the concept of possible world on probabilistic linkages will be defined in Definition 2.20), then the probability that Johnson R. Smith *is younger than* John H. Smith *is* 0.4*, while that probability that* John H. Smith *is younger than* Johnson R. Smith *is* 0.6*. Clearly, between the two patient,* John H. Smith *died at a younger age than* Johnson R. Smith *with higher probability. How to compute this probability will be discussed in Chapter 7.*

In this example, we can consider each linked pair of records as an uncertain instance and each record as an uncertain object. Two uncertain objects from different data sets may share zero or one instance. Therefore, the uncertain objects may not be independent. We develop the probabilistic linkage model to describe such uncertain data. ■

Let $\mathscr{E}$ be a set of real-world entities. We consider two tables A and B which describe subsets $\mathscr{E}_A, \mathscr{E}_B \subseteq \mathscr{E}$ of entities in $\mathscr{E}$. Each entity is described by at most one tuple in each table. In general, $\mathscr{E}_A$ and $\mathscr{E}_B$ may not be identical. Tables A and B may have different schemas as well.

Definition 2.19 (Probabilistic linkage). Consider two tables A and B, each describing a subset of entities in $\mathscr{E}$, a **linkage function** $\mathscr{L} : A \times B \rightarrow [0,1]$ gives a score $\mathscr{L}(t_A, t_B)$ for a pair of tuples $t_A \in A$ and $t_B \in B$ to measure the likelihood that t_A and t_B describe the same entity in $\mathscr{E}$. A pair of tuples $l = (t_A, t_B)$ is called a **probabilistic record linkage** (or **linkage** for short) if $\mathscr{L}(l) > 0$. $Pr(l) = \mathscr{L}(t_A, t_B)$ is the **membership probability** of l. ■

Given a linkage $l = (t_A, t_B)$, the larger the membership probability $Pr(l)$, the more likely the two tuples t_A and t_B describe the same entity. A tuple $t_A \in A$ may participate in zero, one or multiple linkages. The number of linkages that t_A participates in is called the **degree** of t_A, denoted by $d(t_A)$. Symmetrically, we can define the degree of a tuple $t_B \in B$.

For a tuple $t_A \in A$, let $l_1 = (t_A, t_{B_1}), \cdots, l_{d(t_A)} = (t_A, t_{B_{d(t_A)}})$ be the linkages that t_A participates in. For each tuple $t_A \in A$, we can write a **mutual exclusion rule** $R_{t_A} = l_1 \oplus \cdots \oplus l_{d(t_A)}$ which indicates that at most one linkage can hold based on the assumption that each entity can be described by at most one tuple in each table. $Pr(t_A) = \sum_{i=1}^{d(t_A)} Pr(l_i)$ is the probability that t_A is matched by some tuples in B. Since the linkage function is normalized, $Pr(t_A) \leq 1$. We denote by $R_A = \{R_{t_A} | t_A \in A\}$ the

set of mutual exclusion rules for tuples in A. R_{t_B} for $t_B \in B$ and R_B are defined symmetrically.

$(\mathscr{L}, A, B)$ specifies a bipartite graph, where the tuples in A and those in B are two independent sets of nodes, respectively, and the edges are the linkages between the tuples in the two tables.

2.3.2.1 Connections with the Uncertain Object Model

Given a set of probabilistic linkage $\mathscr{L}$ between tuple sets A and B, we can consider each tuple $t_A \in A$ as an uncertain object. For any tuple $t_B \in B$, if there is a linkage $l = (t_A, t_B) \in \mathscr{L}$ such that $Pr(l) > 0$, then t_B can be considered as an instance of object t_A whose membership probability is $Pr(l)$. In contrast to the basic uncertain object model where each instance only belongs to one object, in the probabilistic linkage model, a tuple $t_B \in B$ may be the instance of multiple objects $\{t_{A_1}, \cdots, t_{A_d}\}$, where t_{A_i} is a tuple in A with linkage $(t_{A_i}, t_B) \in \mathscr{L}$ $(1 \leq i \leq d)$. A mutual exclusion rule $R_{t_B} = (t_{A_i}, t_B) \oplus \cdots \oplus (t_{A_d}, t_B)$ specifies that t_B should only belong to one object in a possible world. Alternatively, we can consider each tuple $t_B \in B$ as an uncertain object and a tuple $t_A \in A$ is an instance of t_B if there is a linkage $(t_A, t_B) \in \mathscr{L}$.

A linkage function can be regarded as the summarization of a set of possible worlds.

Definition 2.20 (Possible worlds). For a linkage function $\mathscr{L}$ and tables A and B, let $\mathscr{L}_{A,B}$ be the set of linkages between tuples in A and B. A **possible world** of $\mathscr{L}_{A,B}$, denoted by $W \subseteq \mathscr{L}_{A,B}$, is a set of pairs $l = (t_A, t_B)$ such that (1) for any mutual exclusive rule R_{t_A}, if $Pr(t_A) = 1$, then there exists one pair $(t_A, t_B) \in W$, symmetrically, for any mutual exclusive rule R_{t_B}, if $Pr(t_B) = 1$, then there exists one pair $(t_A, t_B) \in W$; and (2) each tuple $t_A \in A$ participates in at most one pair in W, so does each tuple $t_B \in B$.

$\mathscr{W}_{\mathscr{L},A,B}$ denotes the set of all possible worlds of $\mathscr{L}_{A,B}$. ■

We study the ranking query answering on probabilistic linkage model in Chapter 7.

2.3.3 Uncertain Road Network

As illustrated in Scenario 3 of Example 1.1, the weight of each edge in a graph may be an uncertain object. An uncertain road network is a probabilistic graph defined as follows.

Definition 2.21 (Probabilistic graph). A **probabilistic graph** $G(V, E, W)$ is a simple graph containing a set of vertices V, a set of edges $E \subseteq V \times V$, and a set of weights W defined on edges in E. For each edge $e \in E$, $w_e \in W$ is a real-valued random variable in $(0, +\infty)$, denoting the travel time along edge e. ■

As discussed in Section 2.1.1, the distribution of w_e is often unavailable and can be estimated by a set of *samples* $\{x_1, \cdots, x_m\}$, where each sample $x_i > 0$ $(1 \leq i \leq m)$ takes a *membership probability* $Pr(x_i) \in (0,1]$ to appear. Moreover, $\sum_{i=1}^{m} Pr(x_i) = 1$.

2.3.3.1 Paths and Weight Distribution

A **simple path** P is a sequence of non-repeated vertices $\langle v_1, \ldots, v_{n+1} \rangle$, where $e_i = (v_i, v_{i+1})$ is an edge in E $(1 \leq i \leq n)$. v_1 and v_{n+1} are called the **start vertex** and the **end vertex** of P, respectively. For the sake of simplicity, we call a simple path a **path** in the rest of the paper. Given two vertices u and v, the complete set of paths between u and v is denoted by $\mathscr{P}_{u,v}$.

For paths $P = \langle v_1, \ldots, v_{n+1} \rangle$ and $P' = \langle v_{i_0}, v_{i_0+1}, \ldots, v_{i_0+k} \rangle$ such that $1 \leq i_0 \leq n+1-k$, P is called a **super path** of P' and P' is called a **subpath** of P. Moreover, $P = \langle P_1, P_2, \ldots, P_m \rangle$ if $P_1 = \langle v_1, \ldots, v_{i_1} \rangle$, $P_2 = \langle v_{i_1+1}, \ldots, v_{i_2} \rangle$, $\ldots$, $P_m = \langle v_{i_{m-1}+1}, \ldots, v_{n+1} \rangle$, $1 < i_1 < i_2 < \cdots < i_{m-1} \leq n$. P_j $(1 \leq j \leq m)$ is called a **segment** of P.

The **weight** of path $P = \langle v_1, \ldots, v_{n+1} \rangle$ is the sum of the weights of all edges in P, that is $w_P = \sum_{i=1}^{n} w_{e_i}$, where w_{e_i} is the weight of edge $e_i = (v_i, v_{i+1})$ with probability mass function $f_{e_i}(x)$. Since each w_{e_i} is a discrete random variable, w_P is also a discrete random variable. A **sample** of P is $x_P = \sum_{i=1}^{n} x_i$, where x_i $(1 \leq i \leq n)$ is a sample of edge $e_i = (v_i, v_{i+1})$. We also write $x_P = \langle x_1, \ldots, x_n \rangle$ where $x_1, \ldots, x_n$ are called the **components** of x_P.

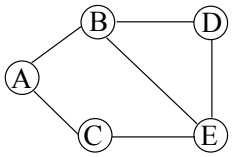
The **probability mass function** of w_P is

$$f_P(x) = Pr[w_P = x] = \sum_{x_1 + \ldots + x_n = x} Pr[w_{e_1} = x_1, \ldots, w_{e_n} = x_n] \tag{2.5}$$

In road networks, the travel time on a road segment e may be affected by the travel time on other roads connecting with e. Therefore, the weights of adjacent edges in E may be correlated. Among all edges in path P, the correlation between the weights w_{e_i} and $w_{e_{i+1}}$ of two adjacent edges e_i and e_{i+1} $(1 \leq i \leq n)$ can be represented using different methods, depending on the types of correlations. To keep our discussion general, in this paper we represent the correlations between w_{e_i} and $w_{e_{i+1}}$ using the joint distribution over the sample pairs $(x_i, x_{i+1}) \in w_{e_i} \times w_{e_{i+1}}$. The **joint probability mass function**[1] of w_{e_i} and $w_{e_{i+1}}$ is $f_{e_i,e_{i+1}}(x_i, x_{i+1}) = f_{e_{i+1},e_i}(x_{i+1}, x_i) = Pr[w_{e_i} = x_i, w_{e_{i+1}} = x_{i+1}]$. Correspondingly, the **conditional probability** of w_{e_i} given $w_{e_{i+1}}$ is $f_{e_i|e_{i+1}}(x_i|x_{i+1}) = \frac{f_{e_i,e_{i+1}}(x_i,x_{i+1})}{f_{e_{i+1}}(x_{i+1})}$.

Theorem 2.1 (Path weight mass function). *The probability mass function of a simple path* $P = \langle v_1, \ldots, v_{n+1} \rangle$ $(e_i = (v_i, v_{i+1})$ *for* $1 \leq i \leq n)$ *is*

[1] The joint travel time distribution among connected roads can be obtained from roadside sensors. The sensors report the speeds of vehicles passing the sensors. The speeds can be transformed into travel time. A set of travel time values reported by sensors at the same time is a sample of the joint travel time distribution.



(a) A graph.

Edge e_i	Weight: value(probability) x_{i1}	x_{i2}	x_{i3}
e_1: AB	10(0.3)	15(0.3)	20(0.4)
e_2: AC	5(0.2)	10(0.3)	15(0.5)
e_3: BD	20(0.4)	25(0.4)	30(0.2)
e_4: BE	5(0.2)	25(0.6)	40(0.2)
e_5: CE	10(0.5)	20(0.1)	45(0.4)
e_6: DE	10(0.3)	20(0.6)	50(0.1)

(b) Probabilistic weights of edges.

	20	25	30
10	0.15	0.15	0
15	0.15	0.15	0
20	0.1	0.1	0.2

(c) $f_{e_1,e_3}(x_{1i},x_{3j})$.

	10	20	50
20	0.1	0.2	0.1
25	0.1	0.3	0
30	0.1	0.1	0

(d) $f_{e_3,e_6}(x_{3i},x_{6j})$.

Fig. 2.6 A probabilistic graph.

$$f_P(x) = \sum_{x_1+\ldots+x_n=x} \frac{\prod_{i=1}^{n-1} f_{e_i,e_{i+1}}(x_i,x_{i+1})}{\prod_{j=2}^{n-1} f_{e_j}(x_j)} \tag{2.6}$$

Proof. Since P is a simple path, each edge $e_i \in P$ $(1 \leq i \leq)$ is only adjacent with e_{i-1} (if $i > 1$) and e_{i+1} (if $i < n$) in P. Therefore, given w_{e_i}, the weights $w_{e_1},\ldots,w_{e_{i-1}}$ are conditionally independent on $w_{e_{i+1}},\ldots,w_{e_n}$. Equation 2.6 follows with basic probability theory. ∎

In sequel, the **cumulative distribution function** of w_P is

$$F_P(x) = Pr[w_P \leq x] = \sum_{0<x_i\leq x} f_P(x_i) \tag{2.7}$$

We call $F_P(x)$ the x**-weight probability** of path P.

Example 2.13 (Probabilistic graph and paths). A probabilistic graph is shown in Figure 2.6, where the weight of each edge is represented by a set of samples and their membership probabilities.

Path $P = \langle A,B,D,E\rangle$ consists of edges AB, BD and DE. The joint probabilities of (w_{AB},w_{BD}) and (w_{BD},w_{DE}) are shown in Figures 2.6(c) and 2.6(d), respectively. The probability that $w_P = 45$ is

$$\begin{aligned} & Pr[w_P = 45] \\ &= Pr[w_{e_1} = 15, w_{e_3} = 20, w_{e_6} = 10] + Pr[w_{e_1} = 10, w_{e_3} = 25, w_{e_6} = 10] \\ &= \frac{f_{e_1,e_3}(15,20)\times f_{e_3,e_6}(20,10)}{f_{e_3}(20)} + \frac{f_{e_1,e_3}(10,25)\times f_{e_3,e_6}(25,10)}{f_{e_3}(25)} \\ &= 0.075 \end{aligned}$$

∎

2.3.3.2 Path Queries

We formulate the probabilistic path queries on uncertain road networks.

Definition 2.22 (Probabilistic path queries). Given probabilistic graph $G(V,E,W)$, two vertices $u,v \in V$, a weight threshold $l > 0$, and a probability threshold $\tau \in (0,1]$, a **probabilistic path query** $Q_l^\tau(u,v)$ finds all paths $P \in \mathscr{P}_{u,v}$ such that $F_P(l) \geq \tau$. ■

There can be many paths between two vertices in a large graph. Often, a user is interested in only the "best" paths and wants a ranked list. Thus, we define weight- and probability-threshold top-k path queries.

Definition 2.23 (Top-k probabilistic path queries). Given probabilistic graph $G(V,E,W)$, two vertices $u,v \in V$, an integer $k > 0$, and a weight threshold $l > 0$, a **weight-threshold top-k path query** $WTQ_l^k(u,v)$ finds the k paths $P \in \mathscr{P}_{u,v}$ with the largest $F_P(l)$ values.

For a path P, given **probability threshold** $\tau \in (0,1]$, we can find the smallest weight x such that $F_P(x) \geq \tau$, which is called the **τ-confident weight**, denoted by

$$F_P^{-1}(\tau) = \min\{x | x \in w_P \wedge Pr[w_P \leq x] \geq \tau\} \tag{2.8}$$

A **probability-threshold top-k path query** $PTQ_\tau^k(u,v)$ finds the k paths $P \in \mathscr{P}_{u,v}$ with the smallest $F_P^{-1}(\tau)$ values. ■

Example 2.14 (Path Queries). In the probabilistic graph in Figure 2.6, there are 4 *paths between A and D, namely* $P_1 = \langle A,B,D\rangle$, $P_2 = \langle A,B,E,D\rangle$, $P_3 = \langle A,C,E,B,D\rangle$, *and* $P_4 = \langle A,C,E,D\rangle$. *Suppose the weights of all edges are independent in this example.*

Given a weight threshold $l = 48$ *and a probability threshold* $\tau = 0.8$, *a probabilistic path query* Q_l^τ *finds the paths whose weights are at most* 48 *of probability at least* 0.8. *According to the cumulative distribution functions of the paths, we have* $F_{P_1}(48) = 0.92$, $F_{P_2}(48) = 0.14$, $F_{P_3}(48) = 0.028$, *and* $F_{P_4}(48) = 0.492$. *Thus, the answer is* $\{P_1\}$.

*The weight-threshold top-*3 *path query* $WTQ_l^3(A,D)$ *finds the top-*3 *paths P having the largest* 48*-weight probability values* $F_P(48)$. *The answer to* $WTQ_l^2(A,D)$ *is* $\{P_1,P_4,P_2\}$.

*The probability-threshold top-*3 *path query* $PTQ_\tau^3(A,D)$ *finds the top-*3 *paths P having the smallest* 0.8*-confidence weights* $F_P^{-1}(0.8)$. *Since* $F_{P_1}(40) = 0.7$ *and* $F_{P_1}(45) = 0.92$, *the smallest weight that satisfies* $F_P(x) \geq 0.8$ *is* 45. *Thus,* $F_{P_1}^{-1}(0.8) = 45$. *Similarly, we have* $F_{P_2}^{-1}(0.8) = 75$, $F_{P_3}^{-1}(0.8) = 105$, *and* $F_{P_4}^{-1}(0.8) = 75$. *Therefore, the answer to* $PTQ_\tau^2(A,D)$ *is* $\{P_1,P_2,P_4\}$. ■

To keep our presentation simple, in the rest of the book, we call probabilistic path queries, weight- and probability-threshold top-k queries as **path queries**, **WT top-k queries**, and **PT top-k queries**, respectively.

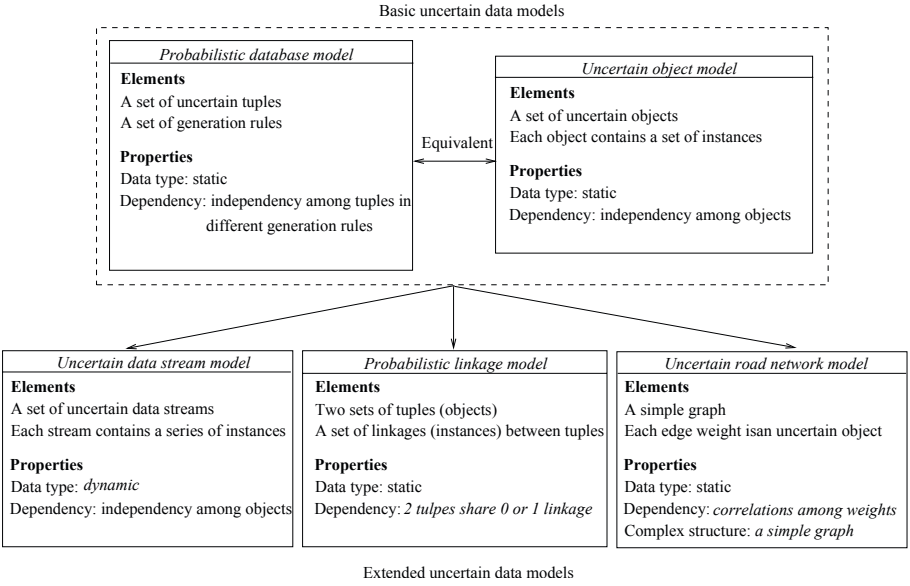


Fig. 2.7 The basic and extended uncertain data models adopted in this book.

2.3.3.3 Connections with the Uncertain Object Model

In the uncertain road network model, the weight of each edge can be considered as an uncertain object with a set of instances. The weight of a path is an aggregate sum of the uncertain objects corresponding to the edges in the path. Therefore, a probabilistic query essentially ranks a set of aggregate sums of uncertain objects.

2.4 Summary

In this chapter, we reviewed two basic uncertain data models, the uncertain object model and the probabilistic database model, as well as the possible worlds semantics that have been extensively adopted in other research on uncertain data.

Moreover, we proposed three extended uncertain data models that extend the uncertain object model from different aspects.

- The *uncertain data stream model* captures the dynamic nature of uncertain data. Each uncertain data stream is a series of (potentially) unlimited instances. Given a sliding window that selects the time span of interest, the instances of each uncertain data stream in the sliding window can be considered as an uncertain object. The uncertain data stream model suits the needs of applications that in-

Uncertain data model	Data type	Structure of data	Data dependency
Probabilistic database model	Static	No structure	Independent
Uncertain object model	Static	No Structure	Independent
Uncertain stream model	Dynamic	No structure	Independent
Probabilistic linkage model	Static	Tree structure	Dependent
Uncertain road network model	Static	Graph structure	Dependent

(a) Uncertain data models adopted in this book.

Problem	Ranking query types			Model
	Granularity	Ranking scope	Ranking criteria	
Typicality queries	Instance	Single object	Probability	Uncertain data model
Probabilistic ranking queries	Instance/ object	Multiple objects	Score & probability	Probabilistic database model
Top-k stream monitoring	Object	Multiple objects	Score & probability	Uncertain stream model
Linkage ranking queries	Instance/ object	Multiple objects	Score & probability	Probabilistic linkage model
Probabilistic path queries	Object set	Multiple objects	Score & probability	Uncertain road network model

(b) Ranking queries addressed in this book.

Table 2.5 Uncertain data models and ranking queries discussed in this book.

volve uncertain data with evolving distributions, such as traffic monitoring and environmental surveillance.

- The *probabilistic linkage model* introduces dependencies among different uncertain objects. It contains two object sets $\mathscr{O}_A$ and $\mathscr{O}_B$ and a set of linkages $\mathscr{L}$. Each linkage matches one object in $\mathscr{O}_A$ with another object in $\mathscr{O}_B$ with a confidence value indicating the how likely the two objects refer to the same real-life entity. Two objects from different object sets may share one instance. The probabilistic linkage model finds important applications in data integration.
- The *uncertain road network model* considers a set of uncertain objects in a simple graph. The weight of each edge in a simple graph is an uncertain object represented by a set of instances. The weights of adjacent edges may involve dependencies. A probabilistic path query finds the optimal paths between two end vertices that have small weights with high confidence. The uncertain road network model is important in applications like real-time trip planning.

Figure 2.7 and Table 2.5(a) summarize the five uncertain data models and their relationship.

Last, we formulated five ranking problems on uncertain data (listed in Table 2.5(b)) and discussed the semantics as well as the potential challenges in query evaluation. Chapters 4 to 8 will discuss the query answering techniques for the proposed ranking problems.



http://www.springer.com/978-1-4419-9379-3

Ranking Queries on Uncertain Data
Hua, M.; Pei, J.
2011, XVI, 224 p., Hardcover
ISBN: 978-1-4419-9379-3