Query Answering Techniques on Uncertain/Probabilistic Data

Jian Pei[1], Ming Hua[1], Yufei Tao[2], Xuemin Lin[3]

[1]Simon Fraser University

[2]The Chinese University of Hong Kong

[3]The University of New South Wales

Acknowledgement

- Some figures in our slides are borrowed from some papers in the references. Thank you!
- We will have a tutorial on mining uncertain data in the upcoming KDD'08 conference

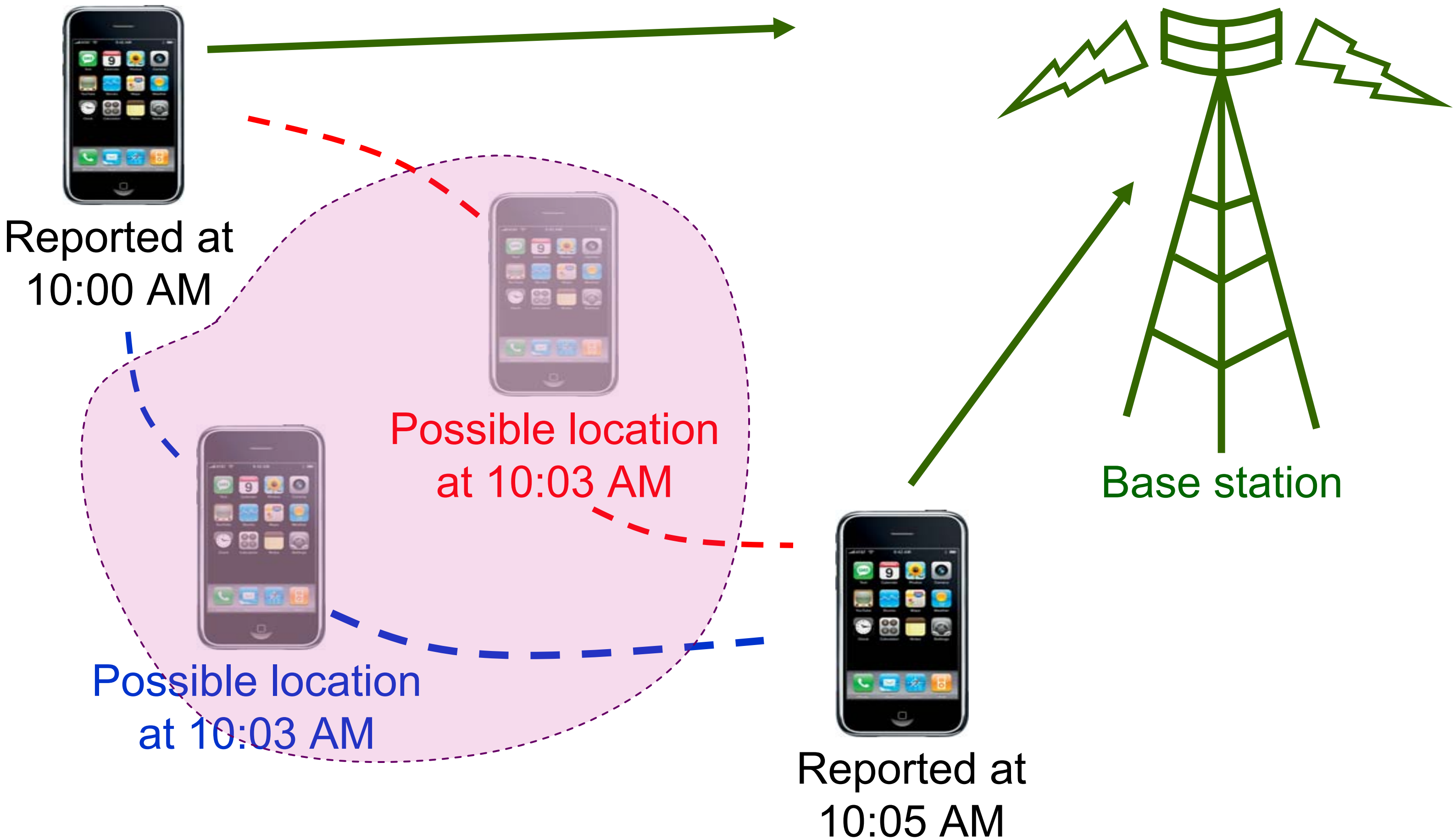
Outline

- Introduction: motivations, applications and challenges
- Models and possible worlds
- Range search queries
- Ranking queries
- Advanced queries
- Summary: challenges and future directions

Uncertainty in Data

- Sensor networks
  - Sensor readings are often imprecise due to sensors and periodic reporting mechanisms
- Mobile equipment
  - A mobile object reports its position periodically, the exact location is often uncertain
- Social data collection
  - Errors and estimations inherent in customer surveys and sampling

Uncertainty in Data



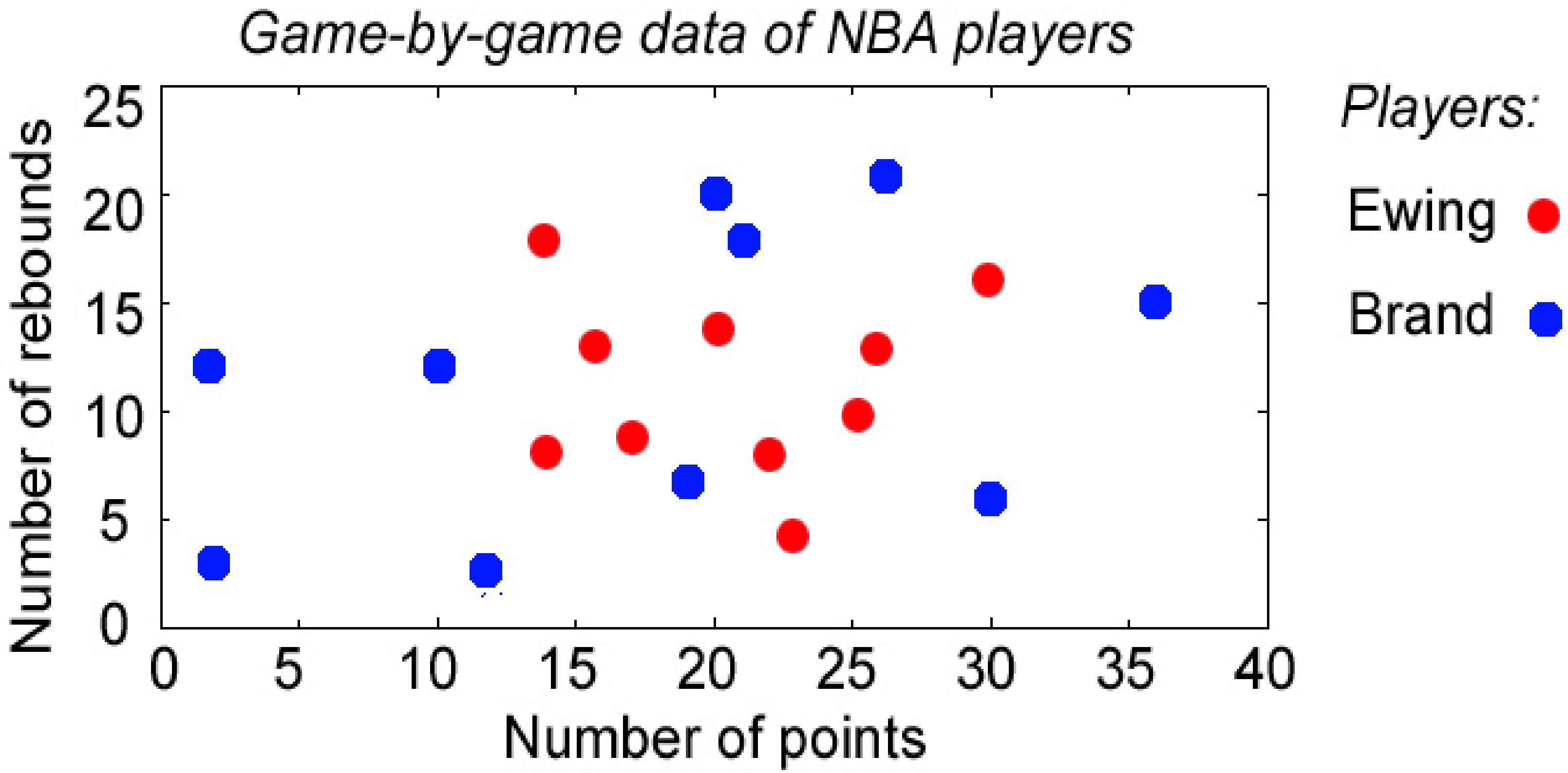
Outline

- Introduction: motivations, applications and challenges
- Models and possible worlds
- Range search queries
- Ranking queries
- Advanced queries
- Summary: challenges and future directions

Models of Uncertain Data

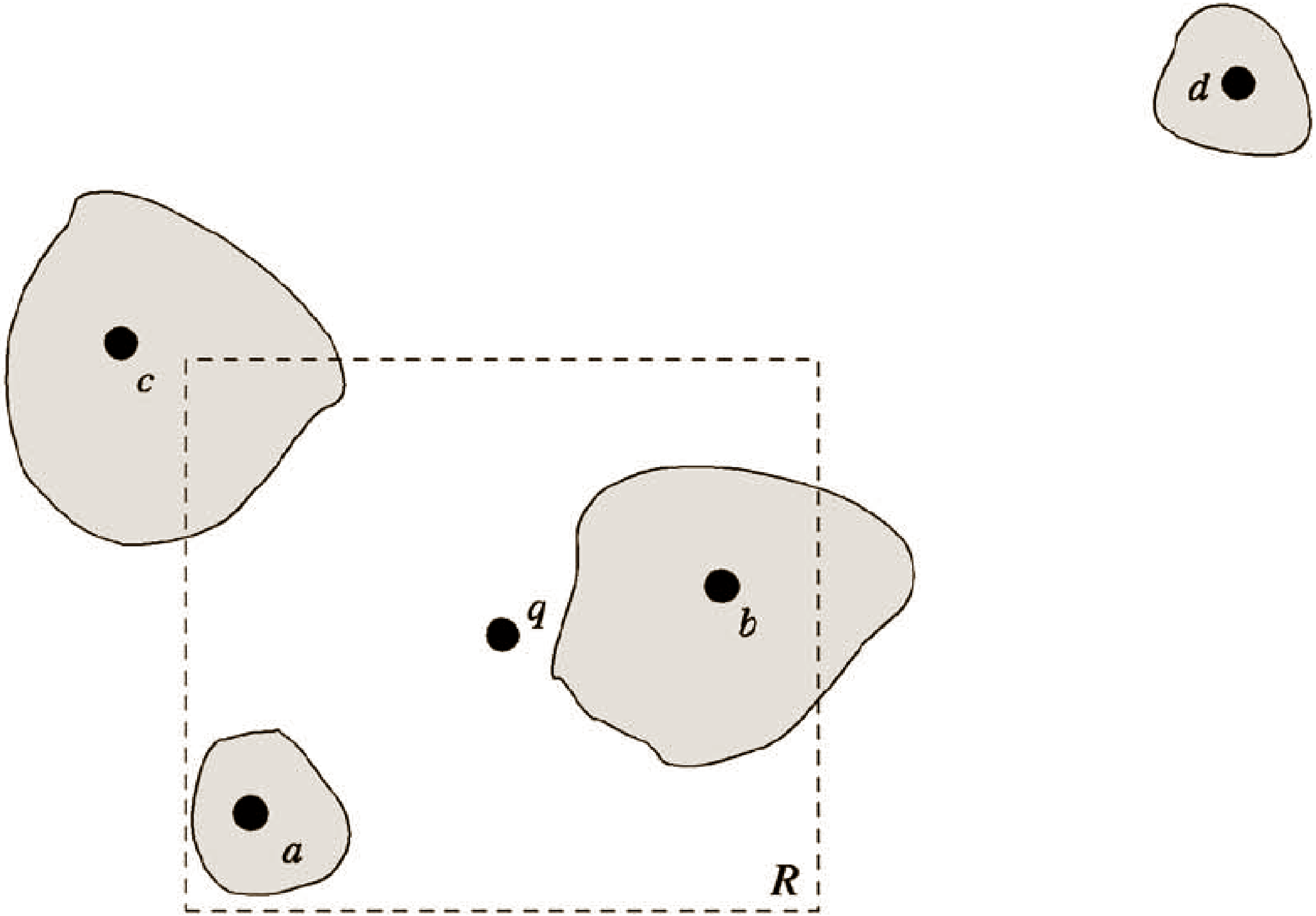
- Uncertain Objects
  - An object is uncertain in a few dynamic attributes
  - Use a sample or a probability density function to capture the distribution
- Probabilistic database
  - The values of each tuple are certain
  - Each tuple carries an existing/membership probability
  - Generation rules: constraints specifying exclusive tuples

Uncertain Objects



Uncertain objects: NBA players

Uncertainty of Mobile Objects



Probabilistic Table

Speed of cars detected by radar

	Time	Radar Location	Car make	Plate No.	Speed	Confidence
t1	11:45	L1	Honda	X-123	130	0.4
t2	11:50	L2	Toyota	Y-245	120	0.7
t3	11:35	L3	Toyota	Y-245	80	0.3
t4	12:10	L4	Mazda	W-541	90	0.4
t5	12:25	L5	Mazda	W-541	110	0.6
t6	12:15	L6	Nissan	L-105	105	1.0

Generation rules: (t2⊕t3), (t4⊕t5)

- The values of each tuple are certain
- Each tuple carries an existing/membership probability
- Generation rules: constraints specifying exclusive tuples

Uncertain object vs. Prob Table

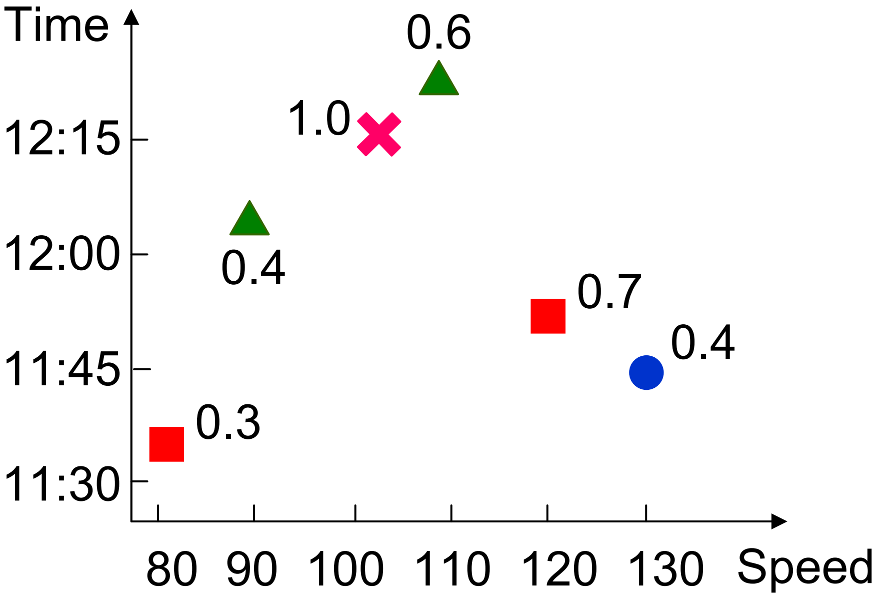
- Uncertain objects with discrete instances can be represented using a probabilistic table
  - One record per instance
  - All instances of an object are constrained by one generation rule
  - Uncertain objects with PDF cannot be represented using a finite probabilistic table
- A probabilistic table can be represented as a set of uncertain objects
  - All tuples in a generation rule are modeled as an uncertain object
  - Use NULL instances to make the sum of membership probabilities in one object to 1

Prob Table vs. Uncertain object

A probabilistic table		A set of uncertain objects
A tuple	⟷	An instance
A generation rule	⟷	An uncertain object

	Time	Radar Loc	Car Make	Plate No	Speed	Conf
t1	11:45	L1	Honda	X-123	130	0.4
t2	11:50	L2	Toyota	Y-245	120	0.7
t3	11:35	L3	Toyota	Y-245	80	0.3
t4	12:10	L4	Mazda	W-541	90	0.4
t5	12:25	L5	Mazda	W-541	110	0.6
t6	12:15	L6	Nissan	L-105	105	1.0

$Rules: (t2 \oplus t3), (t4 \oplus t5)$



Possible Worlds

- A possible world
  - a possible snapshot that may be observed
- Uncertain object model
  - A possible world = a set of instances of uncertain objects
  - At most one instance per object in a possible world
- Probabilistic database model
  - A possible world = a set of tuples
  - At most one tuple per generation rule in a possible world
- A possible world carries an existence probability

Possible Worlds of Probabilistic Data

$0.112 = 0.4 \times 0.7 \times 0.4 \times 1.0$

t2 and t3 never appear in the same possible world!

$0.4 = 0.112 + 0.168 + 0.048 + 0.072$

	Time	Radar Loc	Car Make	Plate No	Speed	Conf
t1	11:45	L1	Honda	X-123	130	0.4
t2	11:50	L2	Toyota	Y-245	120	0.7
t3	11:35	L3	Toyota	Y-245	80	0.3
t4	12:10	L4	Mazda	W-541	90	0.4
t5	12:25	L5	Mazda	W-541	110	0.6
t6	12:15	L6	Nissan	L-105	105	1.0

$Rules: (t2 \oplus t3), (t4 \oplus t5)$

A probabilistic table

World	Prob.
$PW^1=\{t1,t2,t6,t4\}$	0.112
$PW^2=\{t1,t2,t5,t6\}$	0.168
$PW^3=\{t1,t6,t4,t3\}$	0.048
$PW^4=\{t1,t5,t6,t3\}$	0.072
$PW^5=\{t2,t6,t4\}$	0.168
$PW^6=\{t2,t5,t6\}$	0.252
$PW^7=\{t6,t4,t3\}$	0.072
$PW^8=\{t5,t6,t3\}$	0.108

Possible worlds

Another Example

	Time	Radar Loc	Car Make	Plate No	Speed	Conf
t1	11:45	L1	Honda	X-123	130	0.4
t2	11:50	L2	Toyota	Y-245	120	0.7
t3	11:35	L3	Toyota	Y-245	80	0.3
t4	12:10	L4	Mazda	W-541	90	0.4
t5	12:25	L5	Mazda	W-541	110	0.6
t6	12:15	L6	Nissan	L-105	105	1.0

World	Prob.
$PW^1=\{t1,t2,t6,t4\}$	0.16
$PW^2=\{t1,t2,t5,t6\}$	0.24
$PW^3=\{t2,t6,t4\}$	0.12
$PW^4=\{t2,t5,t6\}$	0.18
$PW^5=\{t6,t4,t3\}$	0.12
$PW^6=\{t5,t6,t3\}$	0.18

$$Rules:(t2 \oplus t3),\ (t4 \oplus t5)\ ,\ (t1 \rightarrow t2)$$

Outline

- Introduction: motivations, applications and challenges
- Models and possible worlds
- Range search queries
- Ranking queries
- Advanced queries
- Summary: challenges and future directions

Data taxonomy

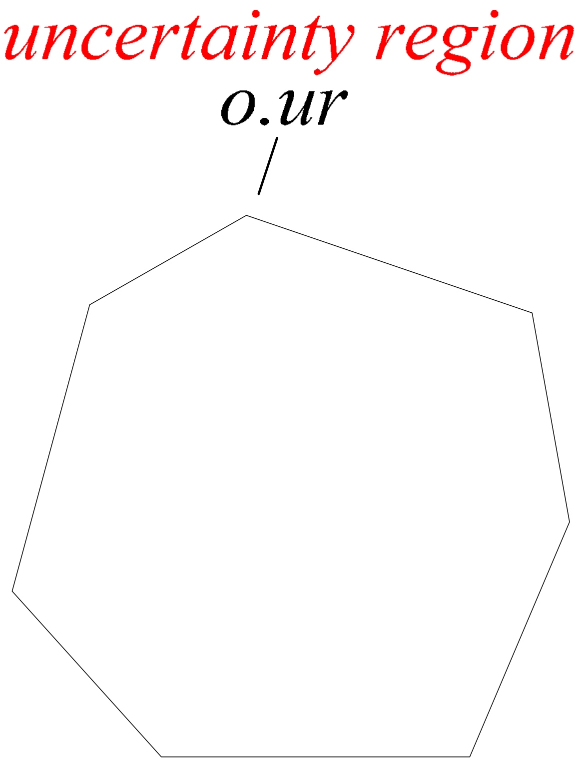
- Each uncertain object is represented by a pdf.

- Numeric
  - Sensor values, locations of moving objects, etc.

- Categorical
  - RFID data, OCR-generated data, text labeling, etc.

Numeric pdf

- The location of a vehicle.
- Its pdf has value 0 anywhere outside its uncertainty region.

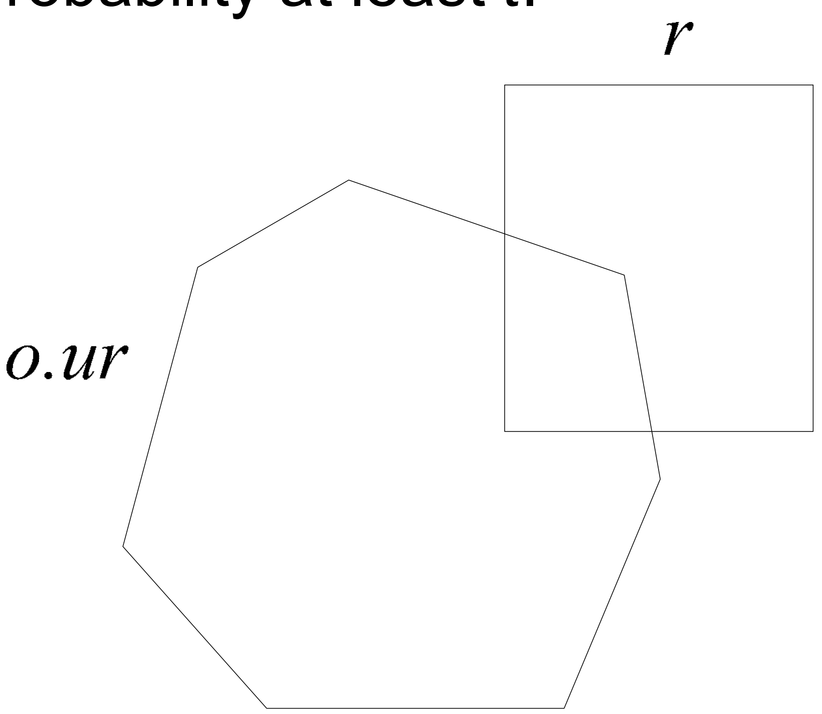


Range search

- Given a rectangle r and a probability threshold t, find all the objects that appear in r with probability at least t.

- <u>Appearance probability</u>

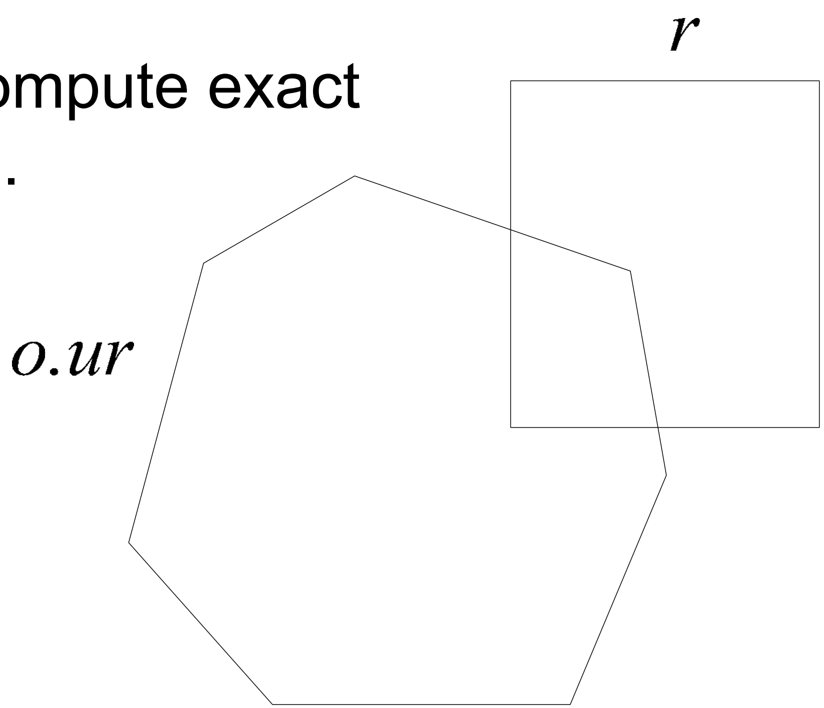
$$\int_{o.ur \cap r} o.pdf(x)\mathrm{d}x$$



Filter-refinement processing

- Why?
  - It can be expensive to compute exact appearance probabilities.

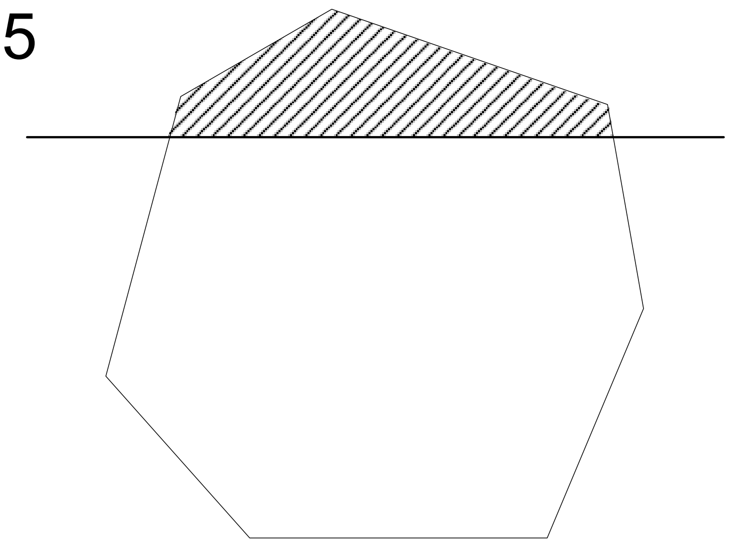
$$\int_{o.ur \cap r} o.pdf(x)\mathrm{d}x \geq t\,?$$



PCR (Tao et al., VLDB 05, TODS 07)

- Probabilistically constrained region (PCR)
  - A rectangle
  - Takes a parameter $0 < p < 0.5$

$$\int o.pdf(x)\mathrm{d}x = p$$

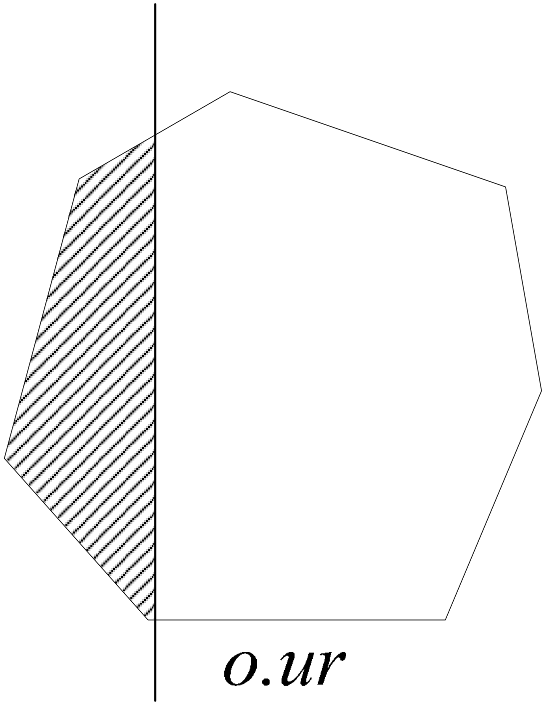


$o.ur$

PCR (Tao et al., VLDB 05, TODS 07)

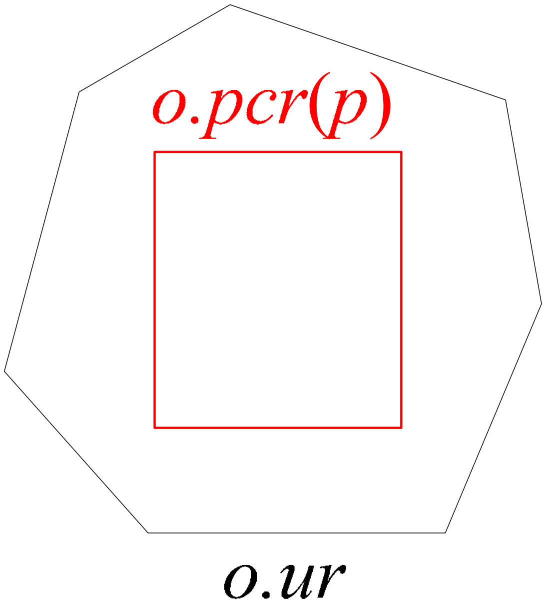
- Probabilistically constrained region (PCR)
  - Takes a parameter $0 < p < 0.5$

$$\int o.pdf(x)\mathrm{d}x = p$$



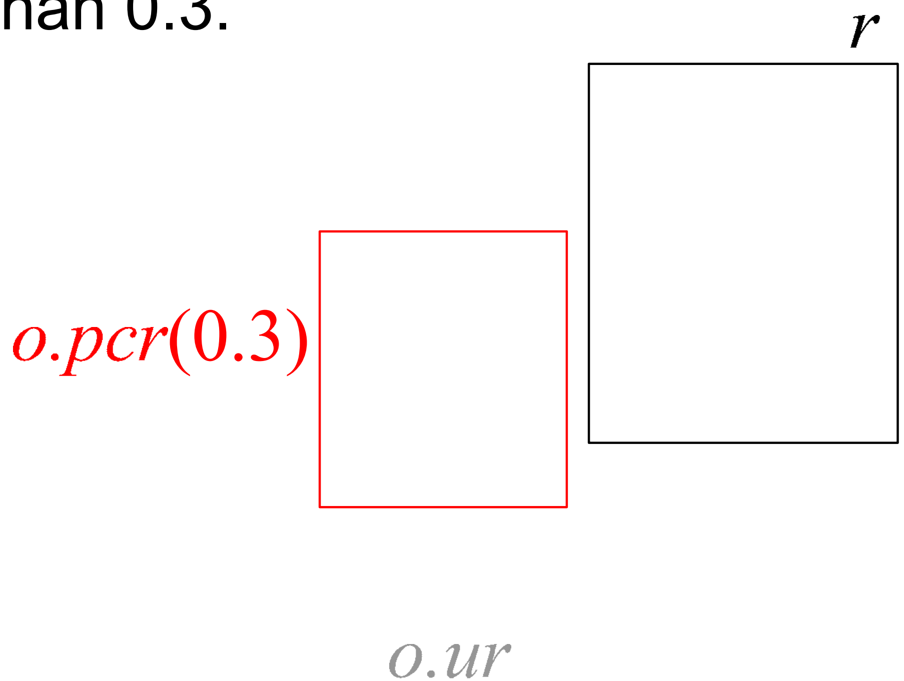
PCR (Tao et al., VLDB 05, TODS 07)

- Probabilistically constrained region (PCR)
  - Takes a parameter $0 < p < 0.5$



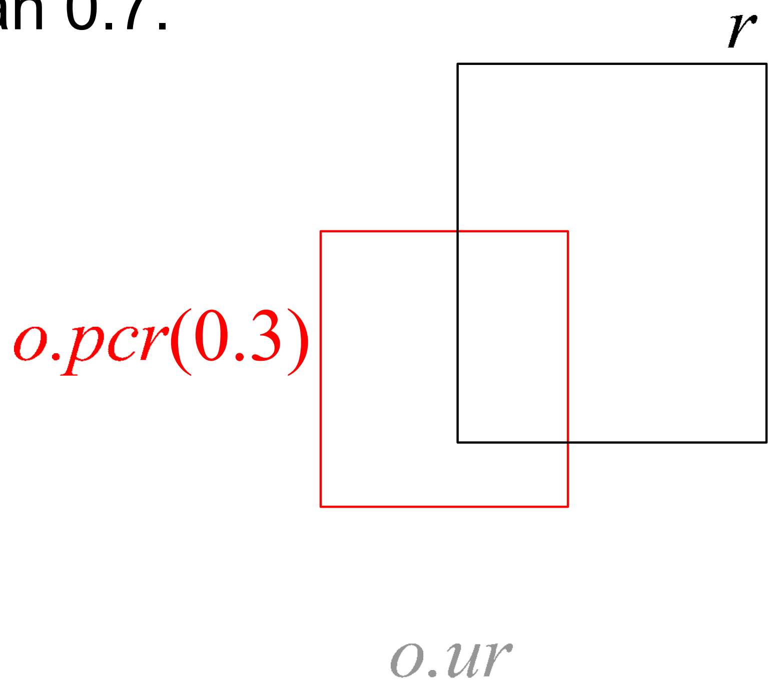
Reasoning with PCR 1

- *o* appears in *r* with probability at most 0.3.
  - *o* can be pruned as long as the probability threshold *t* is larger than 0.3.



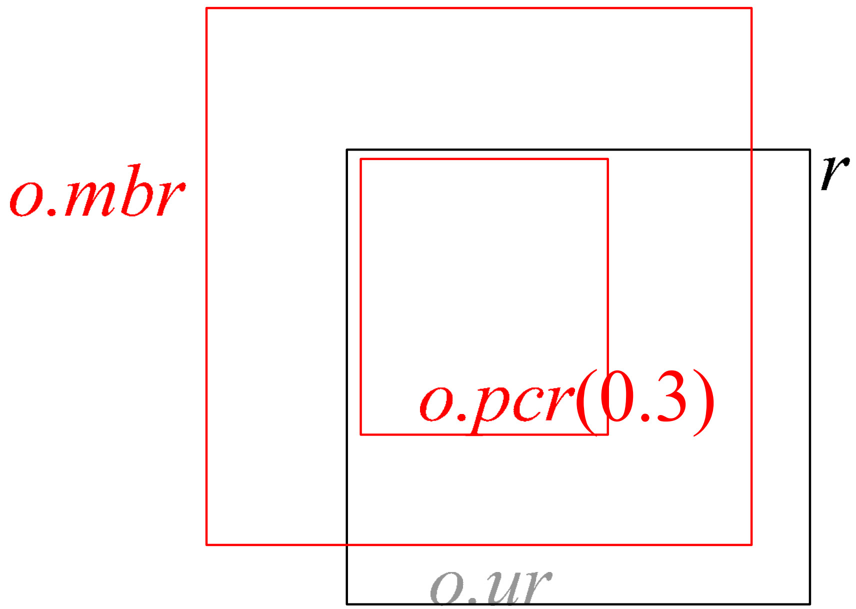
Reasoning with PCR 2

- *o* appears in *r* with probability at most 0.7.
  - *o* can be pruned as long as the probability threshold *t* is larger than 0.7.



Reasoning with PCR 3

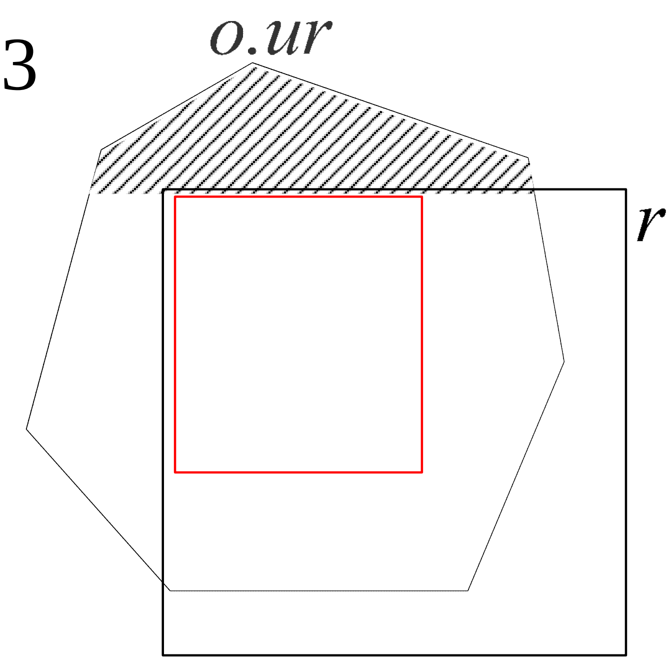
- *o* appears in *r* with probability at least 0.4.
  - *o* can be validated as long as the probability threshold *t* is at most 0.4.



Reasoning with PCR 3

- *o* appears in *r* with probability at least 0.4.

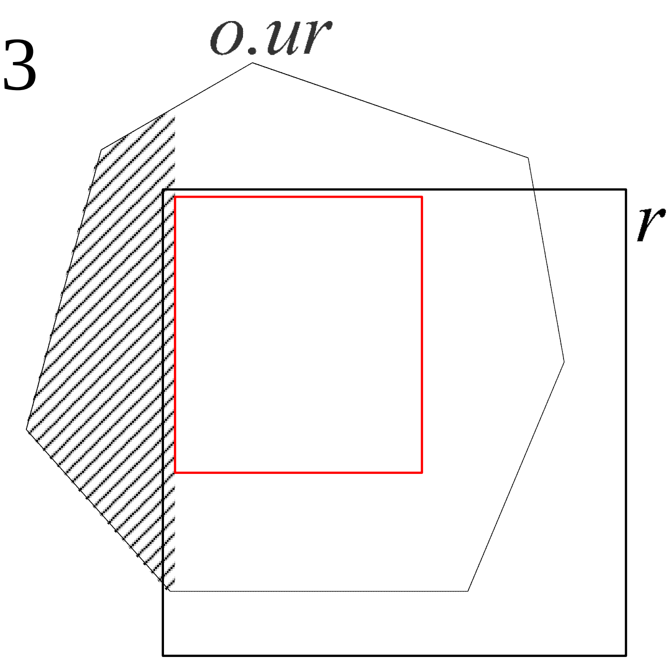
$$\int o.pdf(x)\mathrm{d}x = 0.3$$



Reasoning with PCR 3

- *o* appears in *r* with probability at least 0.4.

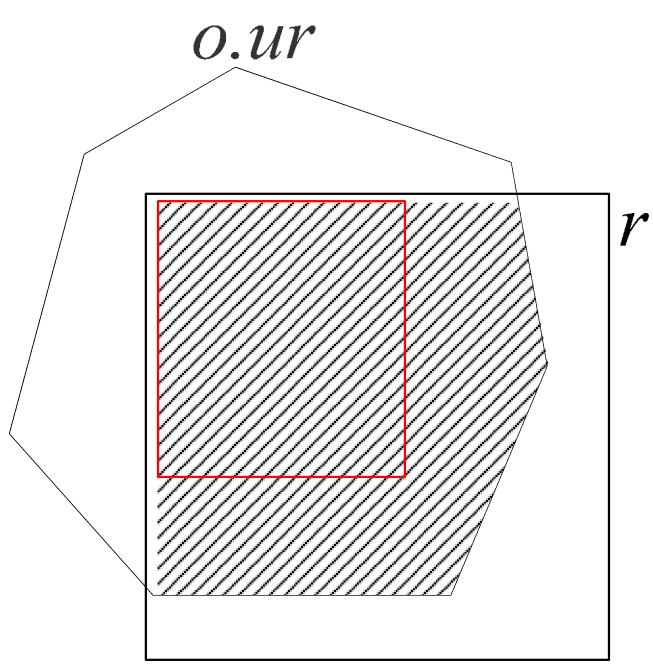
$$\int o.pdf(x)\mathrm{d}x = 0.3$$



Reasoning with PCR 3

- *o* appears in *r* with probability at least 0.4.

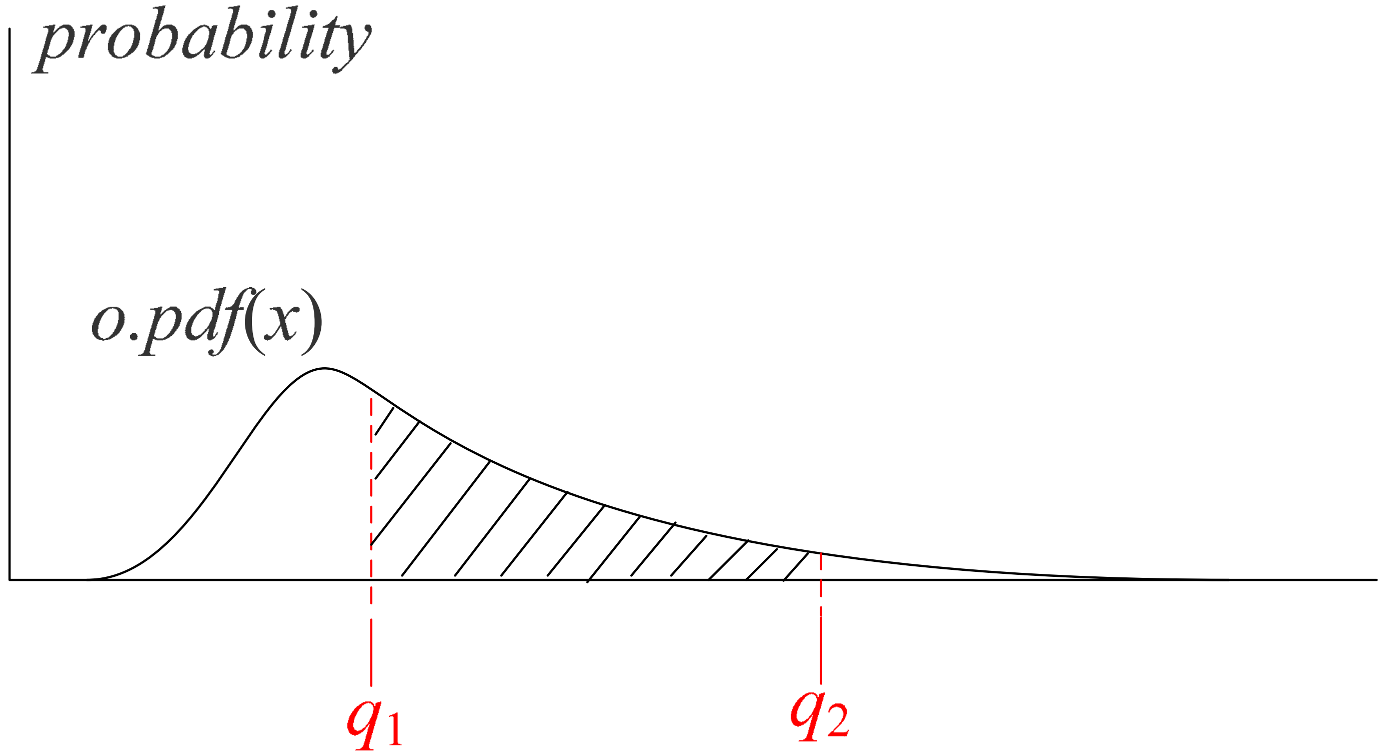
$$\int o.pdf(x)\mathrm{d}x \geq 1-0.3-0.3=0.4$$



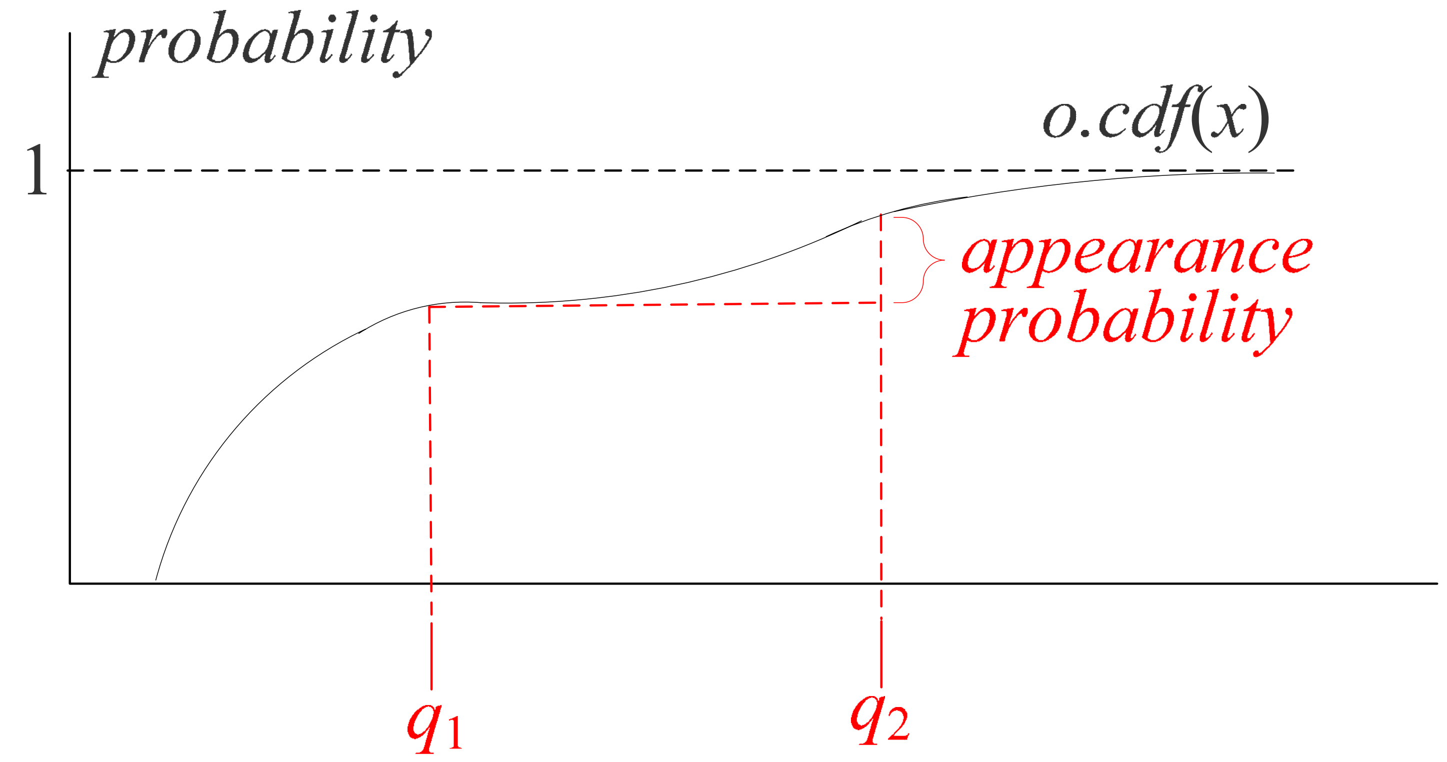
U-tree (Tao et al., VLDB 05, TODS 07)

- More complex reasoning is possible with PCRs.
- PCRs computed at different probabilities are good for pruning/validating for queries with different probability thresholds.
- For each object, prepare its PCRs at several probabilities.
- Index all the PCRs in an R-tree manner.

1D range search

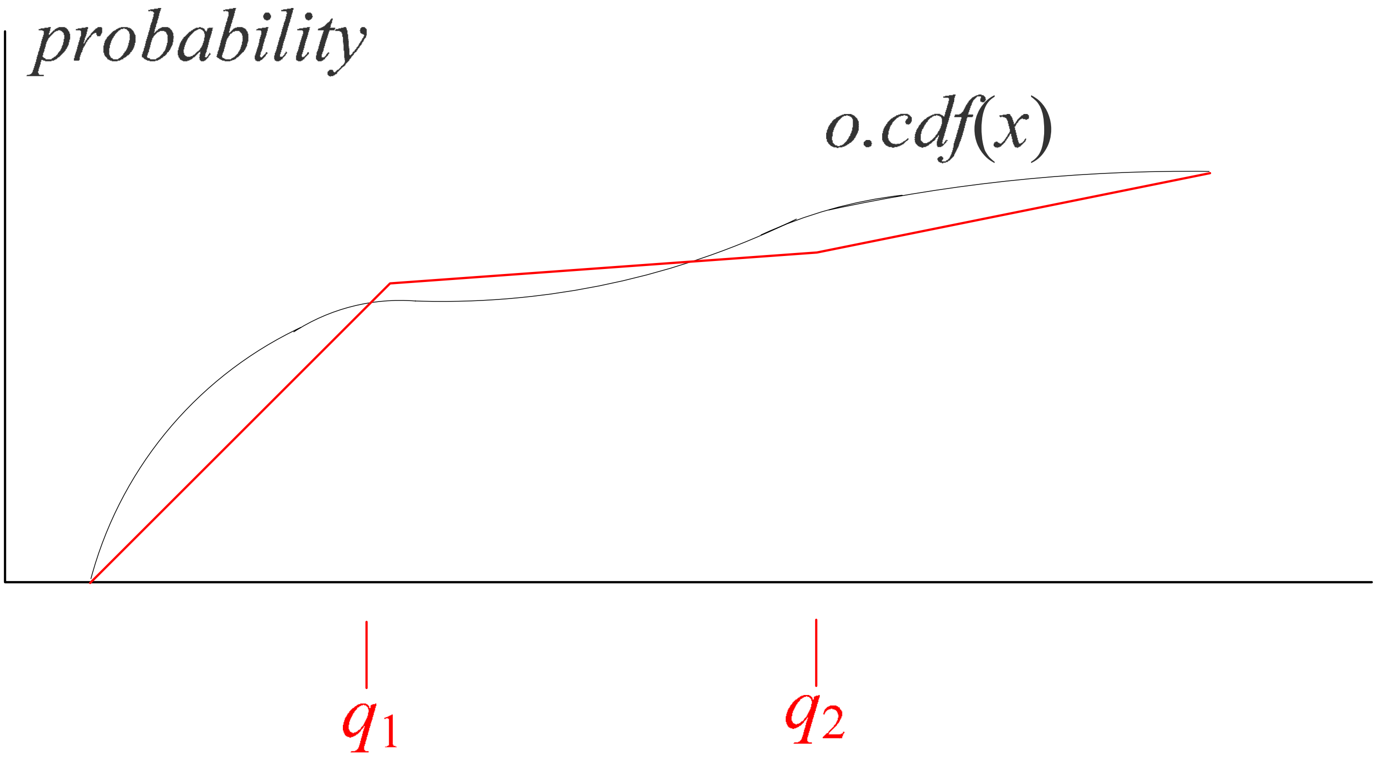


1D range search



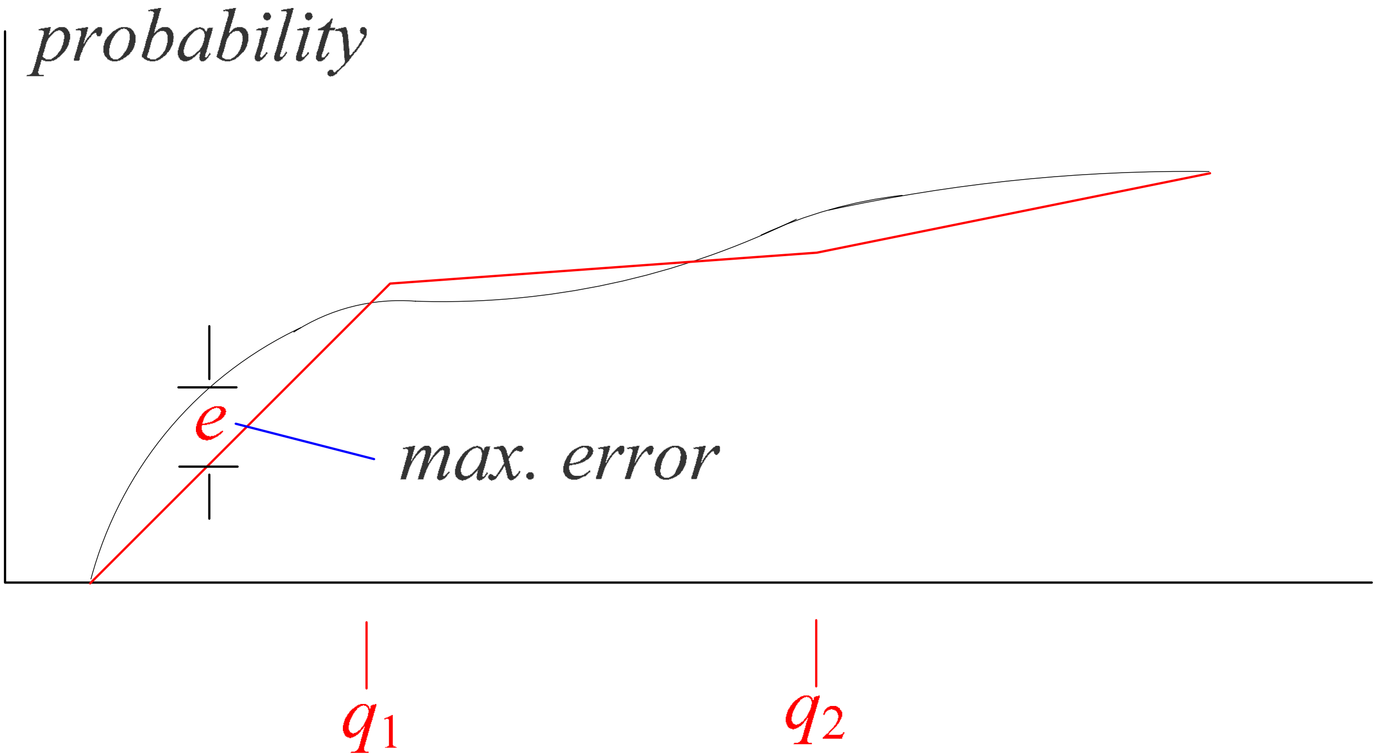
APLA (Ljosa et al. ICDE 07)

- Adaptive piecewise linear approximation (APLA)



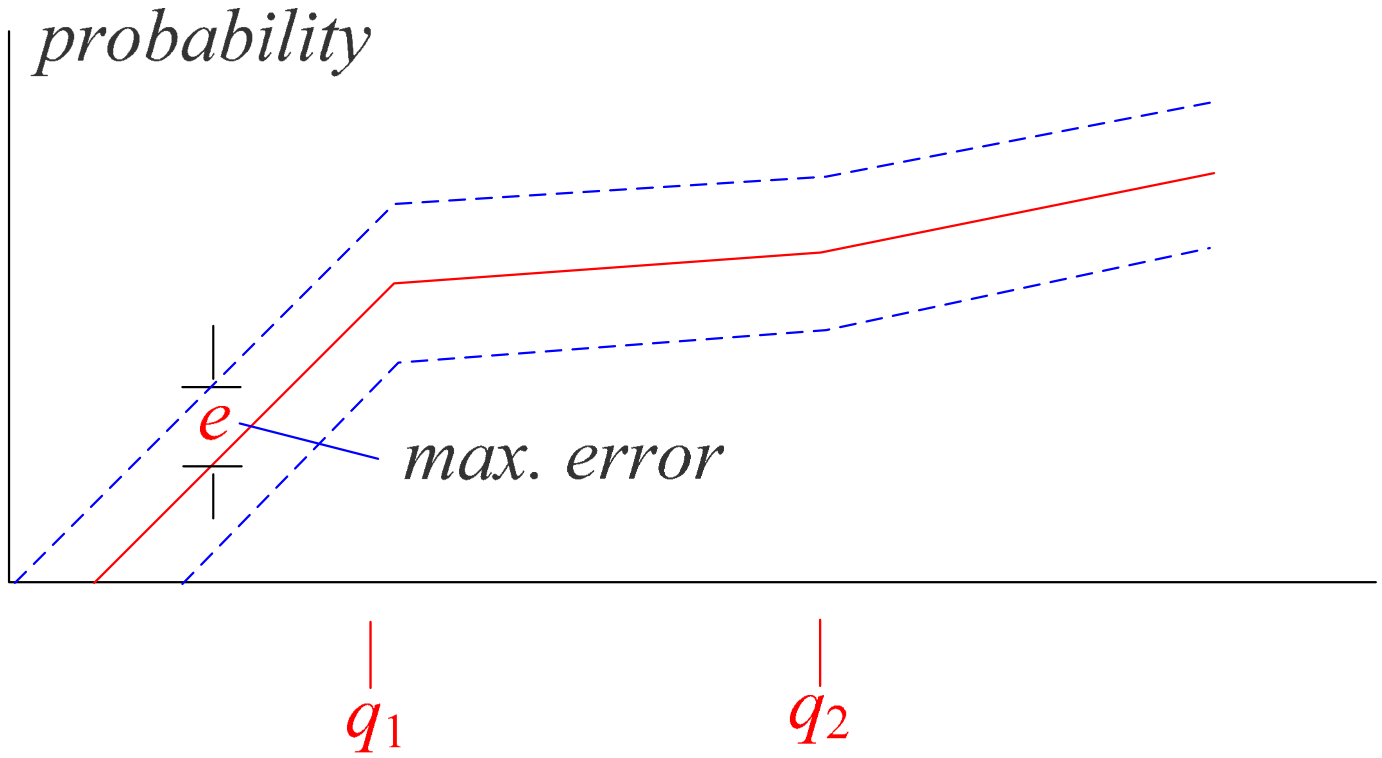
APLA (Ljosa et al. ICDE 07)

- Adaptive piecewise linear approximation (APLA)



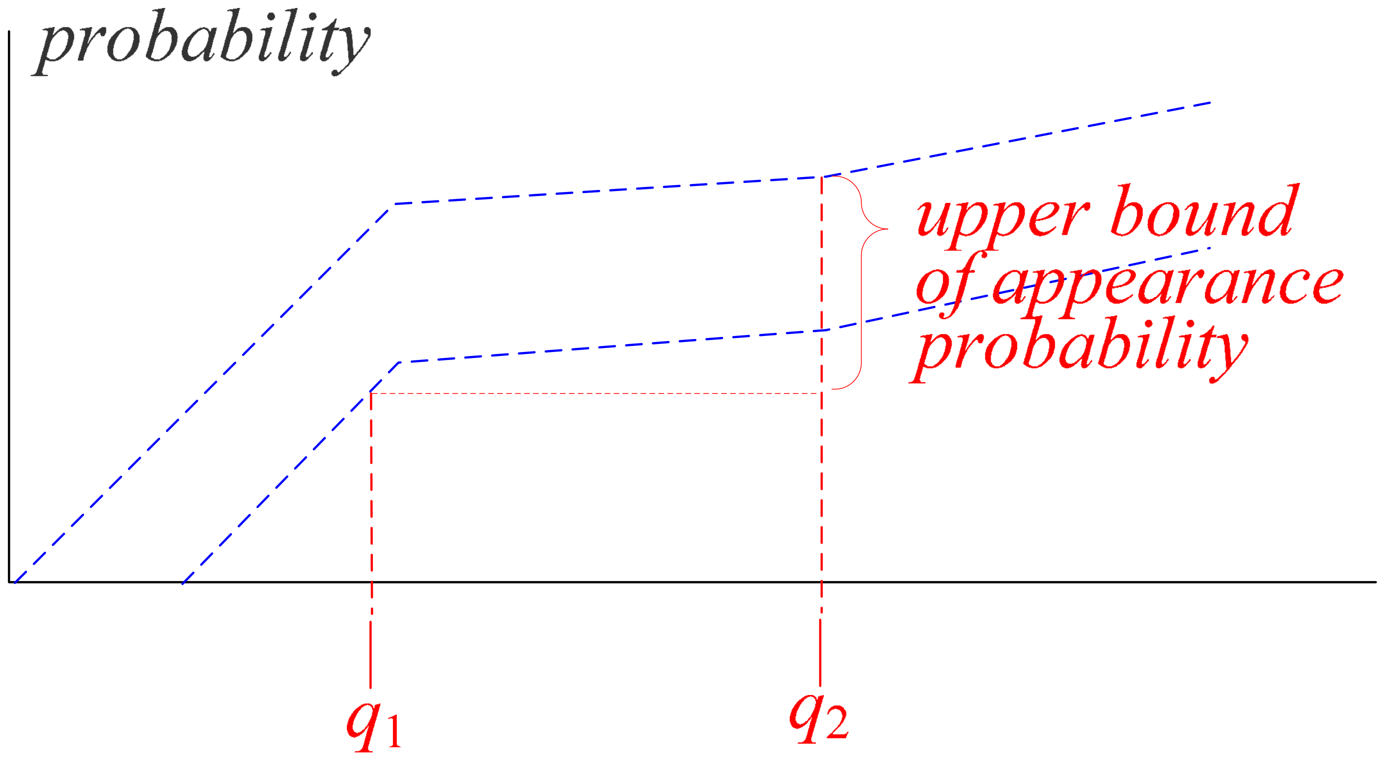
APLA (Ljosa et al. ICDE 07)

- Adaptive piecewise linear approximation (APLA)



APLA (Ljosa et al. ICDE 07)

- Adaptive piecewise linear approximation (APLA)



APLA-tree (Ljosa et al. ICDE 07)

- For each object, compute an APLA.
- Each APLA can be regarded as a time series.
- An APLA-tree organizes these time series in a hierarchical manner.

Other access methods

- Probability thresholding index (PTI)
  - [Cheng et al. VLDB 04]
  - One-dimensional U-tree

- Gauss-tree
  - [Bohm et al. ICDE 06]
  - Each object pdf is a Gaussian function described by (μ,σ), which is regarded as a 2D point.
  - An R-tree on these 2D points.

Categorical pdf

- Data at a vehicle repair center
  - o_1 = (brake, 0.8), (gas, 0.2)
  - o_2 = (engine, 0.5), (brake, 0.4), (gas 0.1)
  - o_3 = (gas, 0.7), (transmission, 0.2), (brake, 0.1)
- The domain of the uncertain attribute has 4 values: engine, brake, gas, transmission.

- In general, let the domain have m values: $v_1, v_2, \ldots, v_m$.
- An object's pdf is an m-dimensional vector:

$$o = (o.pdf(v_1), o.pdf(v_2), \ldots, o.pdf(v_m))$$

Similar query (Singh et al. ICDE 07)

- Given a query pdf q and a similarity threshold t, find all objects o such that $q.pdf(v_1) \cdot o.pdf(v_1) + \ldots + q.pdf(v_m) \cdot o.pdf(v_m) \geq t$.

- o_1 = (brake, 0.8), (gas, 0.2)
- o_2 = (engine, 0.5), (brake, 0.4), (gas 0.1)
- o_3 = (gas, 0.7), (transmission, 0.2), (brake, 0.1)

- q = (brake, 0.7), (gas, 0.3) and $t = 0.5$.
- Answer: o_1 with score 0.56 + 0.06 = 0.62.

R-tree (Singh et al. ICDE 07)

- An object's pdf is an m-dimensional vector:

$$o = (o.pdf(v_1), o.pdf(v_2), \ldots, o.pdf(v_m))$$

- Build an m-dimensional R-tree on all objects.

- Each similarity query is a half-plane search in the m-dimensional space:

$$q \cdot o \geq t$$

Inverted index (Singh et al. ICDE 07)

- Data
  - o_1 = (brake, 0.8), (gas, 0.2)
  - o_2 = (engine, 0.5), (brake, 0.4), (gas 0.1)
  - o_3 = (gas, 0.7), (transmission, 0.2), (brake, 0.1)

- Inverted lists
  - engine: (o_2, 0.5)
  - brake: (o_1, 0.8), (o_2, 0.4), (o_3, 0.1)
  - gas: (o_3, 0.7), (o_1, 0.2), (o_2, 0.1)
  - transmission: (o_3, 0.2)

Inverted index (Singh et al. ICDE 07)

- q = (brake, 0.7), (gas, 0.3) and t = 0.5

- Inverted lists
  - engine: (o_2, 0.5)
  - brake: (o_1, 0.8), (o_2, 0.4), (o_3, 0.1)
  - gas: (o_3, 0.7), (o_1, 0.2), (o_2, 0.1)
  - transmission: (o_3, 0.2)

Inverted index (Singh et al. ICDE 07)

- q = (brake, 0.7), (gas, 0.3) and t = 0.5
- Inverted lists
  - brake: (o_1, 0.8)
  - gas: (o_3, 0.7)
- Partial scores
  - o_1 = 0.56
  - o_3 = 0.21

Inverted index (Singh et al. ICDE 07)

- q = (brake, 0.7), (gas, 0.3) and t = 0.5
- Inverted lists
  - brake: (o_1, 0.8), (o_2, 0.4),
  - gas: (o_3, 0.7), (o_1, 0.2)
- Partial scores
  - o_1 = 0.62
  - o_3 = 0.21 (max possible score = 0.21 + 0.28 = 0.49)
  - o_2 = 0.28 (max possible score = 0.28 + 0.06 = 0.33)
  - any other object's max possible score = 0.33.
- The algorithm stops here.

Outline

- Introduction: motivations, applications and challenges
- Models and possible worlds
- Range search queries
- Ranking queries
- Advanced queries
- Summary: challenges and future directions

Ranking Queries

- Find the top-2 sensors with highest temperature
  - Certain data: answer = {R1, R2}
  - Uncertain data
    - R1 and R2 may not co-exist in a possible world
    - In different possible worlds, the answers are different

RID	Loc.	Time	Sensor-id	Temperature	Conf.
$R1$	A	6/2/06 2:14	$S101$	25	0.3
$R2$	B	7/3/06 4:07	$S206$	21	0.4
$R3$	B	7/3/06 4:09	$S231$	13	0.5
$R4$	A	4/12/06 20:32	$S101$	12	1.0
$R5$	E	3/13/06 22:31	$S063$	17	0.8
$R6$	E	3/13/06 22:28	$S732$	11	0.2

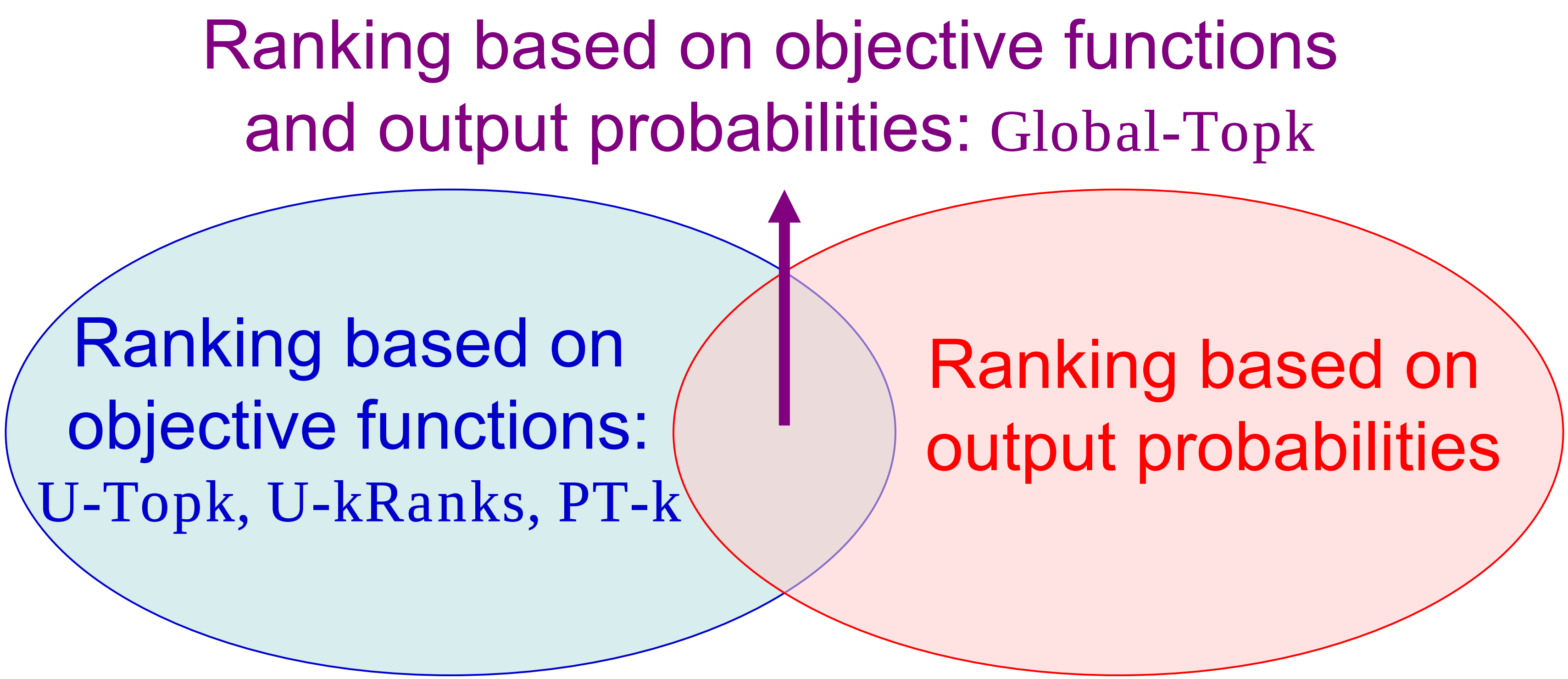
$R2 \oplus R3 \quad R5 \oplus R6$

Challenges

- What does a probabilistic ranking query mean?
  - A ranking query on certain data returns the best k results in the ranking function
  - Ranking queries on uncertain data may be formulated differently to address different application interests
- How can a ranking query be answered efficiently?
  - Answering ranking queries on probabilistic databases can be very costly when the number of possible worlds is huge

Query Types

- How are tuples ranked?



Ranking Based on Objective Functions

- A scoring function is given
  - Rank the sensors in temperature descending order and select the top-2 results

$$R1 \prec R2 \prec R5 \prec R3 \prec R4 \prec R1$$

- How should the top-2 ranking results be captured?

RID	Loc.	Time	Sensor-id	Temperature	Conf.
$R1$	A	6/2/06 2:14	$S101$	25	0.3
$R2$	B	7/3/06 4:07	$S206$	21	0.4
$R3$	B	7/3/06 4:09	$S231$	13	0.5
$R4$	A	4/12/06 20:32	$S101$	12	1.0
$R5$	E	3/13/06 22:31	$S063$	17	0.8
$R6$	E	3/13/06 22:28	$S732$	11	0.2

$R2 \oplus R3 \quad R5 \oplus R6$

U-Topk Queries

- Find the most probable top-2 list in possible worlds
  - ⟨R1,R2⟩: p=0.12
  - ⟨R1,R5⟩: p=0.144
  - ⟨R1,R3⟩: p=0.03
  - ⟨R1,R4⟩: p=0.006
  - ⟨R2,R5⟩: p=0.224
  - ⟨R2,R4⟩: p=0.056
  - ⟨R5,R3⟩: p=0.28
  - ⟨R3,R4⟩: p=0.07
  - ⟨R5,R4⟩: p=0.056
  - ⟨R4,R6⟩: p=0.014
- Answer: ⟨R5,R3⟩

Possible world	Probability	Top-2 on Temperature
$W1 = \{R1, R2, R4, R5\}$	0.096	$R1, R2$
$W2 = \{R1, R2, R4, R6\}$	0.024	$R1, R2$
$W3 = \{R1, R3, R4, R5\}$	0.12	$R1, R5$
$W4 = \{R1, R3, R4, R6\}$	0.03	$R1, R3$
$W5 = \{R1, R4, R5\}$	0.024	$R1, R5$
$W6 = \{R1, R4, R6\}$	0.006	$R1, R4$
$W7 = \{R2, R4, R5\}$	0.224	$R2, R5$
$W8 = \{R2, R4, R6\}$	0.056	$R2, R4$
$W9 = \{R3, R4, R5\}$	0.28	$R5, R3$
$W10 = \{R3, R4, R6\}$	0.07	$R3, R4$
$W11 = \{R4, R5\}$	0.056	$R5, R4$
$W12 = \{R4, R6\}$	0.014	$R4, R6$

U-kRanks Queries

- Find the tuple of the highest probability at each ranking position
  - The 1st position
    - R1: p=0.3
    - R2: p=0.28
    - R5: p=0.336
    - R3: p=0.07
    - R4: p=0.014
  - The 2nd position
    - R5: p=0.368
- Answer: ⟨R5,R5⟩

Possible world	Probability	Top-2 on Temperature	
$W1 = \{R1, R2, R4, R5\}$	0.096	$R1$	$R2$
$W2 = \{R1, R2, R4, R6\}$	0.024	$R1$	$R2$
$W3 = \{R1, R3, R4, R5\}$	0.12	$R1$	$R5$
$W4 = \{R1, R3, R4, R6\}$	0.03	$R1$	$R3$
$W5 = \{R1, R4, R5\}$	0.024	$R1$	$R5$
$W6 = \{R1, R4, R6\}$	0.006	$R1$	$R4$
$W7 = \{R2, R4, R5\}$	0.224	$R2$	$R5$
$W8 = \{R2, R4, R6\}$	0.056	$R2$	$R4$
$W9 = \{R3, R4, R5\}$	0.28	$R5$	$R3$
$W10 = \{R3, R4, R6\}$	0.07	$R3$	$R4$
$W11 = \{R4, R5\}$	0.056	$R5$	$R4$
$W12 = \{R4, R6\}$	0.014	$R4$	$R6$

PT-k Queries

- Find the tuples whose probabilities to be in the top-2 list are at least p (p=0.35)
  - R1: p=0.3
  - R2: p=0.4
  - R3: p=0.38
  - R4: p=0.202
  - R5: p=0.704
  - R6: p=0.014
- Answer: {R2,R3,R5}

Possible world	Probability	Top-2 on Temperature
$W1 = \{R1, R2, R4, R5\}$	0.096	$R1, R2$
$W2 = \{R1, R2, R4, R6\}$	0.024	$R1, R2$
$W3 = \{R1, R3, R4, R5\}$	0.12	$R1, R5$
$W4 = \{R1, R3, R4, R6\}$	0.03	$R1, R3$
$W5 = \{R1, R4, R5\}$	0.024	$R1, R5$
$W6 = \{R1, R4, R6\}$	0.006	$R1, R4$
$W7 = \{R2, R4, R5\}$	0.224	$R2, R5$
$W8 = \{R2, R4, R6\}$	0.056	$R2, R4$
$W9 = \{R3, R4, R5\}$	0.28	$R5, R3$
$W10 = \{R3, R4, R6\}$	0.07	$R3, R4$
$W11 = \{R4, R5\}$	0.056	$R5, R4$
$W12 = \{R4, R6\}$	0.014	$R4, R6$

Query Answering Methods

- The dominant set property
  - For any tuple t, whether t is in the answer set only depends on the tuples ranked higher than t
  - The dominant set of t is the subset of tuples in T that are ranked higher than t
    - E.g. the dominant set of R3 is S_{R3}={R1,R2,R5}
- Framework of Query Answering Methods
  - Retrieve tuples in the ranking order
  - Evaluate each tuple based on its dominant set

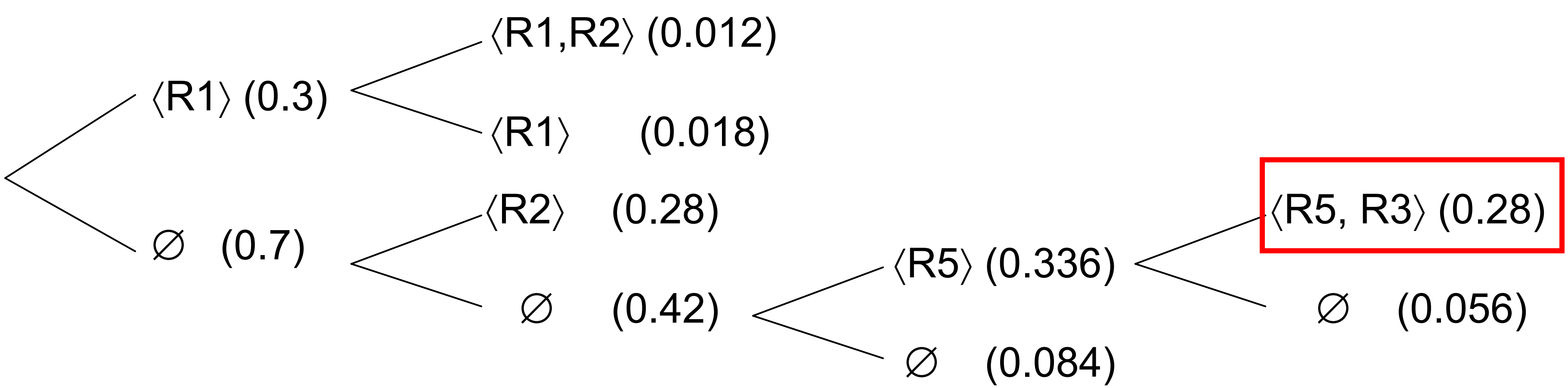
Ranked tuples:

Temperature	25	21	17	13	12	11
RID	$R1$	$R2$	$R5$	$R3$	$R4$	$R6$

Answering U-Topk Queries

- Scan tuples in the ranking order
  - Extend top-k lists based on the scanned tuples
  - Store all top-k lists in a priority queue on their probabilities
  - Stop when a top-k list has a greater probability than that of any top-(k-1) lists

Ranked list: R1 R2 R5 R3 ....



Answering U-kRanks and PT-k Queries

- Position probability $Pr(t_i,j)$
  - The probability that t_i is ranked at the j-th position
  - E.g. $Pr(R3,2)=Pr(R3)\times Pr(S_{R3},1)$

Ranked tuples:

Temperature	25	21	17	13	12	11
RID	$R1$	$R2$	$R5$	$R3$	$R4$	$R6$

R3 is ranked 2nd, if R3 appears, and 1 tuple in S_{R3} appears

- Generally: $\Pr(t_i, j) = \Pr(t_i) \times \Pr(S_{t_i}, j-1)$

Answering U-kRanks and PT-k Queries

- Subset probability $\Pr(S_{ti},j)$
  - The probability that j tuples appear in S_{ti}
  - E.g. $S_{R3}=\{R5\} \cup S_{R5}$
  - $\Pr(S_{R3},2)=\Pr(R5) \times \Pr(S_{R5},1)+(1-\Pr(R5)) \times \Pr(S_{R5},2)$

Temperature	25	21	17	13	12	11
RID	$R1$	$R2$	$R5$	$R3$	$R4$	$R6$

2 tuples appear in S_{R3}, if $\begin{cases} \text{R5 appears, 1 tuple appears in } S_{R5} \\ \text{R5 does not appear, 2 tuples appear in } S_{R5} \end{cases}$

- Generally (Poisson Binomial Recurrence):

$$\Pr(S_{t_i}, j) = \Pr(t_i) \times \Pr(S_{t_{i-1}}, j-1) + (1-\Pr(t_i)) \times \Pr(S_{t_{i-1}}, j)$$

Summary of Query Answering Methods

- Optimal algorithms for U-Topk and U-kRanks queries in terms of the number of accessed tuples (Soliman *et al*. ICDE'07)
- Query answering algorithms for U-Topk and U-kRanks queries based on Poisson binomial recurrence (Yi *et al*. ICDE'08)
- Spatial and probabilistic pruning techniques for U-kRanks queries (Lian and Chen, EDBT'08)
- Efficient query answering algorithms and pruning techniques for PT-k queries (Hua *et al*. ICDE'08, SIGMOD'08)

Ranking based on Output Probabilities

- Query Q: find the average temperature of all sensors
- Ranking: find the top-2 results with the highest probabilities of being the answers to Q (output probabilities)
  - Answer: 14 (p=0.28), 16.67 (p=0.224)

Possible world	Probability	Average temperature
$W1 = \{R1, R2, R4, R5\}$	0.096	18.75
$W2 = \{R1, R2, R4, R6\}$	0.024	17.25
$W3 = \{R1, R3, R4, R5\}$	0.12	16.75
$W4 = \{R1, R3, R4, R6\}$	0.03	15.25
$W5 = \{R1, R4, R5\}$	0.024	18
$W6 = \{R1, R4, R6\}$	0.006	16
$W7 = \{R2, R4, R5\}$	0.224	16.67
$W8 = \{R2, R4, R6\}$	0.056	14.67
$W9 = \{R3, R4, R5\}$	0.28	14
$W10 = \{R3, R4, R6\}$	0.07	12
$W11 = \{R4, R5\}$	0.056	14.5
$W12 = \{R4, R6\}$	0.014	11.5

Query Answering

- Monte Carlo Simulation (1 step)
  - Choose a possible world at random, and evaluate the query
  - Record the answer to the query and its frequency
- E.g. If we run 100 steps of Monte Carlo simulation, and "14" is the answer in 30 steps
  - The output probability of "14" can be approximated by 30/100=0.3, with an error bound ε
  - The output probability of "14" lies in the probability interval $[0.3-\varepsilon, 0.3+\varepsilon]$
  - The more steps of Monte Carlo simulation we run, the smaller probability intervals we can get

Query Answering (cont.)

- The simulation stops when the top-k output probabilities and their relative ranks are clear
  - E.g. There are 5 possible results G1, G2, G3, G4 and G5. After a few steps of Monte Carlo simulation, the output probability interval of each result is shown below
  - G3's output probability is in top-2. The other answer might be one of G1, G2, and G4 . But G5's output probability cannot be in top-2

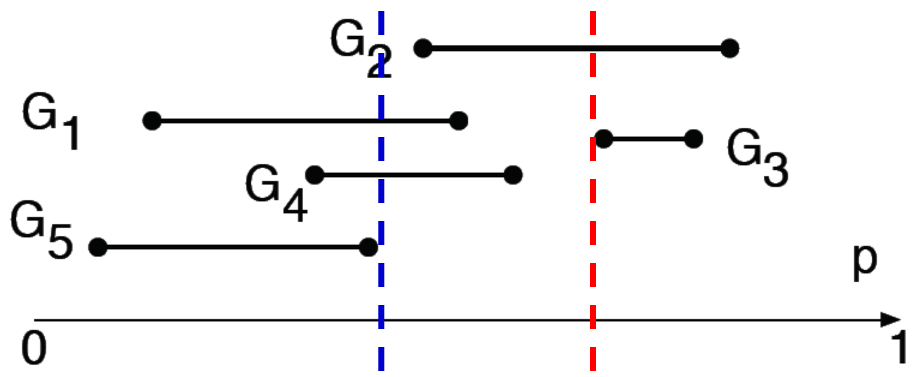


Figure borrowed from C. Re *et al*. Efficient top-k query evaluation on probabilistic data. In ICDE'07.

Global-Topk

- Find the top-2 tuples whose probabilities to be in the top-2 list are the highest
- Ranking based on objective functions and output probabilities
- Example
  - R1: p=0.3
  - R2: p=0.4
  - R3: p=0.38
  - R4: p=0.202
  - R5: p=0.704
  - R6: p=0.014
- Answer={R5,R2}

Possible world	Probability	Top-2 on Temperature
$W1 = \{R1, R2, R4, R5\}$	0.096	$R1, R2$
$W2 = \{R1, R2, R4, R6\}$	0.024	$R1, R2$
$W3 = \{R1, R3, R4, R5\}$	0.12	$R1, R5$
$W4 = \{R1, R3, R4, R6\}$	0.03	$R1, R3$
$W5 = \{R1, R4, R5\}$	0.024	$R1, R5$
$W6 = \{R1, R4, R6\}$	0.006	$R1, R4$
$W7 = \{R2, R4, R5\}$	0.224	$R2, R5$
$W8 = \{R2, R4, R6\}$	0.056	$R2, R4$
$W9 = \{R3, R4, R5\}$	0.28	$R5, R3$
$W10 = \{R3, R4, R6\}$	0.07	$R3, R4$
$W11 = \{R4, R5\}$	0.056	$R5, R4$
$W12 = \{R4, R6\}$	0.014	$R4, R6$

Answering Global-Topk Queries

- Position probability computation for k-Ranks and PT-k queries can be adopted to answer Global-Topk queries
- Threshold Algorithm Optimization
  - Consider *score* and *probability* as two special attributes
  - Apply TA algorithm to Global-Topk computation
  - The algorithm can stop as soon as possible
- A sampling-based method (Silberstein *et al*. ICDE'06)

Properties of Ranking Queries

- Exact-k
  - The cardinality of the answer set is exactly k (|T|>k)
- Faithfulness
  - For any two tuples t1, t2 in T, if both the score and the probability of t1 are higher than those of t2, and t2 is in the answer set, then t1 should also be in the answer set
- Stability
  - If t is an answer, then t will remain in the answer set if its score/probability is increased
  - If t is not an answer, then t cannot be in the answer set if its score/probability is decreased

U-Topk Queries: Exact-k

- U-Topk queries do not satisfy the exact-k property
- Example
  - The most probable top-2 list is ⟨R1⟩
  - The number of tuples in the answer is smaller than 2

An uncertain table

Tuple	Score	Probability
R1	20	0.9
R2	10	0.2

Possible worlds

Possible World	Probability	Top-2 list
W1={R1,R2}	0.18	⟨R1,R2⟩
W2={R1}	0.72	⟨R1⟩
W3={R2}	0.02	⟨R2⟩
W4= ∅	0.08	∅

U-Topk Queries: Faithfulness

- U-Topk queries do not satisfy the faithfulness property
- Example:
  - The most probable top-2 list is ⟨R1,R3⟩
  - The score and probability of R2 are larger than those of R3, but R2 is not in the answer

An uncertain table

Tuple	Score	Probability
R1	50	0.6
R2	40	0.4
R3	30	0.35
R4	20	0.25
R5	10	0.2

Rules: R1⊕R2, R3⊕R4⊕R5

Possible worlds

Possible World	Probability	Top-2 list
W1={R1,R3}	0.21	⟨R1,R3⟩
W2={R1,R4}	0.15	⟨R1,R4⟩
W3={R1,R5}	0.12	⟨R1,R5⟩
W4= {R1}	0.12	⟨R1⟩
W5={R2,R3}	0.14	⟨R2,R3⟩
W6={R2,R4}	0.1	⟨R2,R4⟩
W7={R2,R5}	0.08	⟨R2,R5⟩
W8={R2}	0.08	⟨R2⟩

U-kRanks Queries: Exact-k

- U-kRanks queries do not satisfy the exact-k property
- Example:
  - The 1st position
    - R5: p=0.336
  - The 2nd position
    - R5: p=0.368
  - Answer: ⟨R5,R5⟩

Possible world	Probability	Top-2 on Temperature	
$W1 = \{R1, R2, R4, R5\}$	0.096	$R1$	$R2$
$W2 = \{R1, R2, R4, R6\}$	0.024	$R1$	$R2$
$W3 = \{R1, R3, R4, R5\}$	0.12	$R1$	$R5$
$W4 = \{R1, R3, R4, R6\}$	0.03	$R1$	$R3$
$W5 = \{R1, R4, R5\}$	0.024	$R1$	$R5$
$W6 = \{R1, R4, R6\}$	0.006	$R1$	$R4$
$W7 = \{R2, R4, R5\}$	0.224	$R2$	$R5$
$W8 = \{R2, R4, R6\}$	0.056	$R2$	$R4$
$W9 = \{R3, R4, R5\}$	0.28	$R5$	$R3$
$W10 = \{R3, R4, R6\}$	0.07	$R3$	$R4$
$W11 = \{R4, R5\}$	0.056	$R5$	$R4$
$W12 = \{R4, R6\}$	0.014	$R4$	$R6$

  - The number of tuples in the answer set is smaller than 2

U-kRanks Queries: Faithfulness

- U-kRanks queries do not satisfy the faithfulness property
- Example:
  - The score and probability of R2 are higher than those of R3, but R2 is not in the answer set

An uncertain table

Tuple	Score	Probability
R1	50	0.6
R2	40	0.4
R3	30	0.35
R4	20	0.25
R5	10	0.2

Rules: R1⊕R2, R3⊕R4⊕R5

Answer: Rank 1: R1(p=0.6)
Rank 2: R3 (p=0.35)

Possible worlds

Possible World	Probability	Rank 1	Rank 2
W1={R1,R3}	0.21	R1	R3
W2={R1,R4}	0.15	R1	R4
W3={R1,R5}	0.12	R1	R5
W4= {R1}	0.12	R1	∅
W5={R2,R3}	0.14	R2	R3
W6={R2,R4}	0.1	R2	R4
W7={R2,R5}	0.08	R2	R5
W8={R2}	0.08	R2	∅

U-kRanks Queries: Stability

- U-kRanks queries do not satisfy the stability property
- Example:
  - When the score of R2 is 40, R2 is in the answer set
  - When the score of R2 is increased to 60, R2 is not in the answer set anymore

An uncertain table

Tuple	Score	Probability
R1	50	0.6
R2	40	0.3
R3	30	0.1

Answer: Rank 1: R1(p=0.6)
Rank 2: R2 (p=0.18)

Increase score(R2) →

An uncertain table

Tuple	Score	Probability
R2	60	0.3
R1	50	0.6
R3	30	0.1

Answer: Rank 1: R1(p=0.42)
Rank 2: R1 (p=0.18)

PT-k Queries: Exact-k

- PT-k queries do not satisfy the exact-k property
- Example
  - R1: p=0.3
  - R2: p=0.4
  - R3: p=0.38
  - R4: p=0.202
  - R5: p=0.704
  - R6: p=0.014

Possible world	Probability	Top-2 on Temperature
$W1 = \{R1, R2, R4, R5\}$	0.096	$R1, R2$
$W2 = \{R1, R2, R4, R6\}$	0.024	$R1, R2$
$W3 = \{R1, R3, R4, R5\}$	0.12	$R1, R5$
$W4 = \{R1, R3, R4, R6\}$	0.03	$R1, R3$
$W5 = \{R1, R4, R5\}$	0.024	$R1, R5$
$W6 = \{R1, R4, R6\}$	0.006	$R1, R4$
$W7 = \{R2, R4, R5\}$	0.224	$R2, R5$
$W8 = \{R2, R4, R6\}$	0.056	$R2, R4$
$W9 = \{R3, R4, R5\}$	0.28	$R5, R3$
$W10 = \{R3, R4, R6\}$	0.07	$R3, R4$
$W11 = \{R4, R5\}$	0.056	$R5, R4$
$W12 = \{R4, R6\}$	0.014	$R4, R6$

- Answer: {R2,R3,R5} (k=2, p=0.35)
- The number of tuples in the answer set is greater than 2

Comparison

Queries	Exact k	Faithfulness	Stability
U-Topk	×	×	√
U-kRanks	×	×	×
PT-k	×	√	√
Global-Topk	√	√	√

A part of the table is borrowed from X. Zhang and J. Chomichi. On the Semantics and Evaluation of Top-k Queries in Probabilistic Databases. In ICDE Workshops 2008.

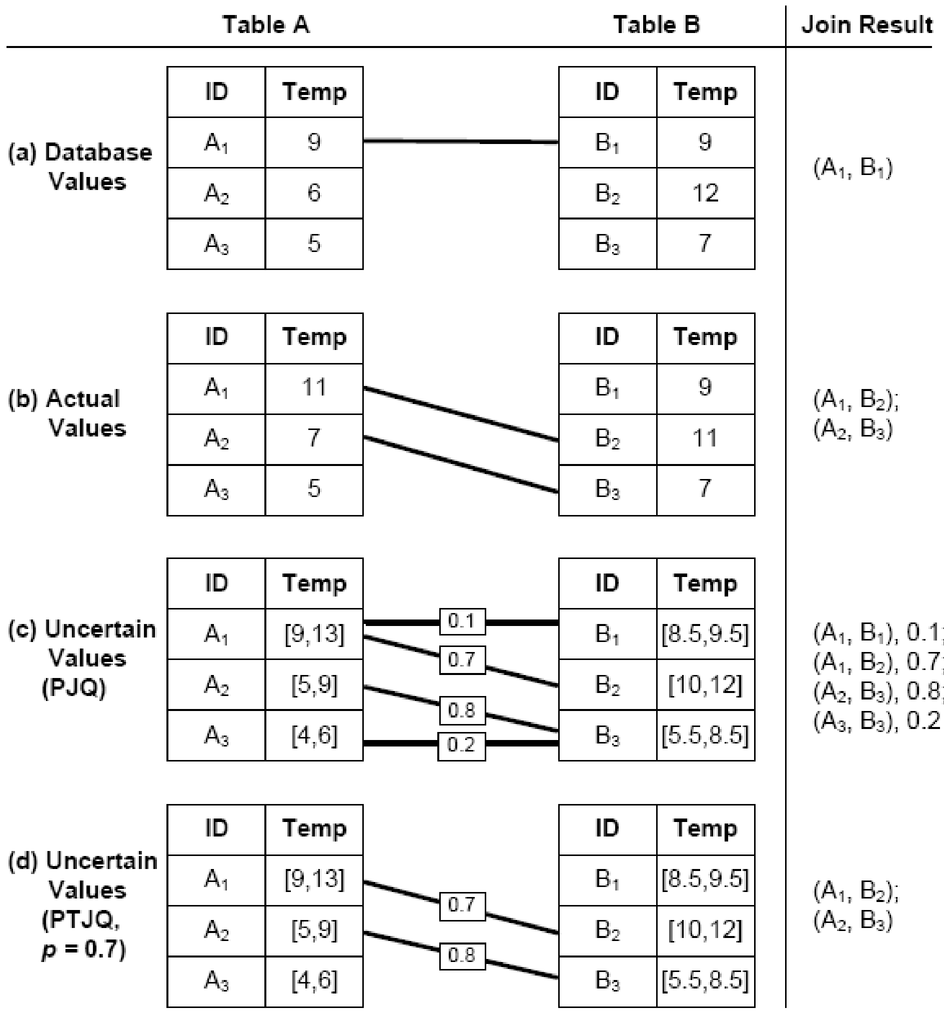
Outline

- Introduction: motivations, applications and challenges
- Models and possible worlds
- Range search queries
- Ranking queries
- Advanced queries
- Summary: challenges and future directions

Joining Uncertain Data

- The join operator is essential in relational databases of certain data
- How to join uncertain and probabilistic data?
  - Attribute-uncertainty: probabilistic join queries
  - Tuple-uncertainty: confidence-aware join queries

Probabilistic Join Queries

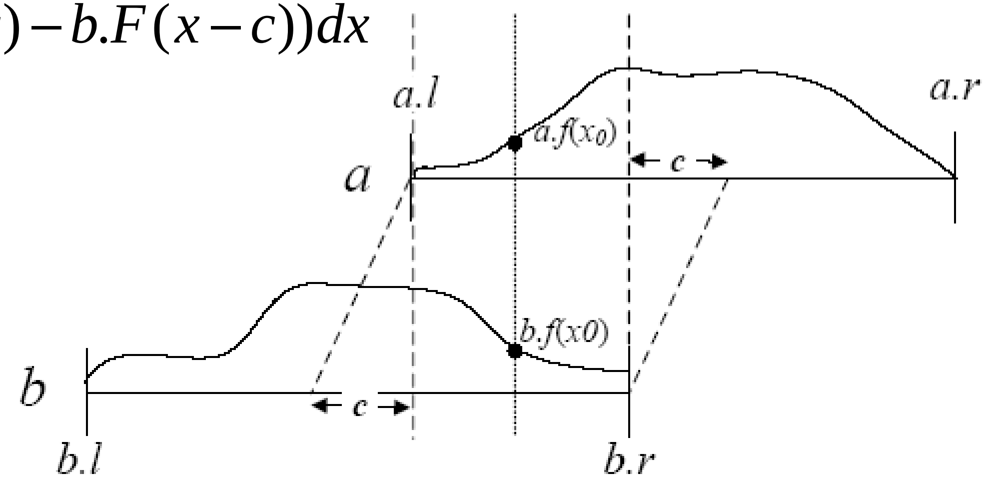


Comparing Uncertain Values

- By comparing uncertain values, we can obtain the probability where an equality/inequality (in some resolution) may hold

$$P(a =_c b) = \int_{-\infty}^{\infty} a.f(x) \cdot (b.F(x+c) - b.F(x-c))dx$$

$$P(a \neq_c b) = \int_{-\infty}^{\infty} a.f(x) \cdot (b.F(x+c) - b.F(x-c))dx$$



Pruning Techniques

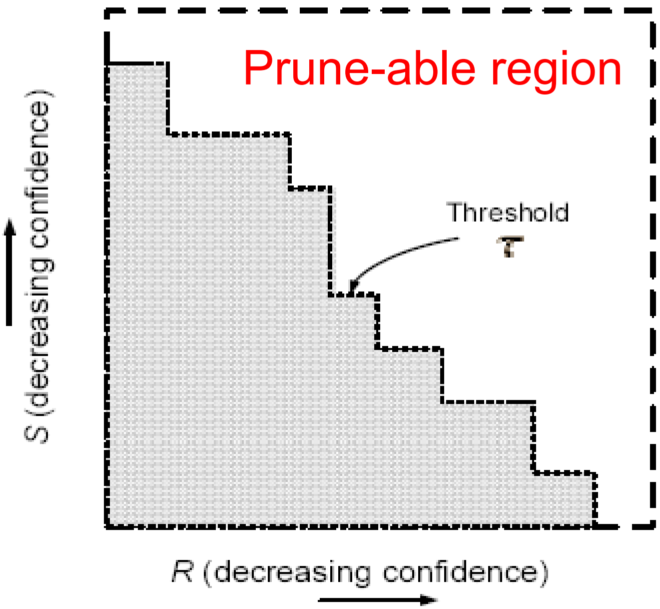
- Item-level pruning
  - If $Pr(r_i, s_j) < p$ (the probability threshold), then (r_i, s_j) can be pruned
  - Method: obtaining the upper bound of $Pr(r_i, s_j)$
- Page-level pruning
  - If each interval value on a page has a probability less than p to join the interval in the other table, the page can be pruned
- Index-level pruning
  - To reduce I/O cost, extend the idea of page-level pruning by organizing the pages in a tree structure

Confidence-Aware Joins

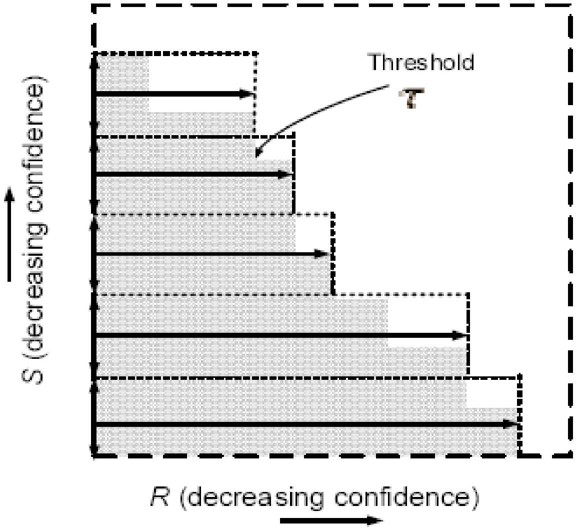
- Each tuple carries a confidence
  - Only the joining results with high confidence should be returned
- Four types of queries
  - Threshold: return only result tuples with confidence passing a threshold
  - Top-k: return k tuples with the highest confidence values
  - Sorted: return result tuples sorted by confidence
  - Sorted-threshold: return result tuples with confidence above a threshold, sorted by confidence

Confidence-Descending Processing

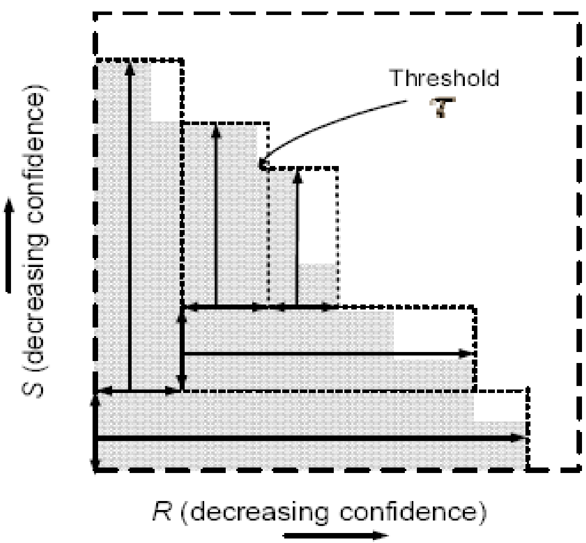
Essential idea



Nested loop method



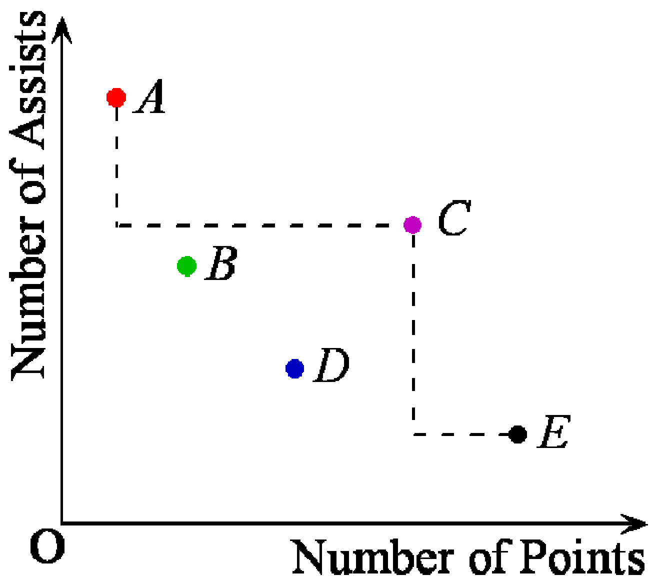
Adaptive nested loop: exploring "longer" rectangles



The idea can be extended to handle top-k, sorted, and sorted-threshold queries

Skyline Queries

- Numeric space $D = (D_1, \ldots, D_n)$, larger values are more preferable
- Two points, u dominates v ($u \succ v$), if
  - $\forall D_i$ ($1 \leqslant i \leqslant n$), $u.D_i \geqslant v.D_i$
  - $\exists D_j$ ($1 \leqslant j \leqslant n$), $u.D_j > v.D_j$
- Given a set of points S, skyline = {u | u∈S and u is not dominated by any other point}
  - Example: $C \succ B$, $C \succ D$
    skyline = {A, C, E}
- A well studied problem with many applications

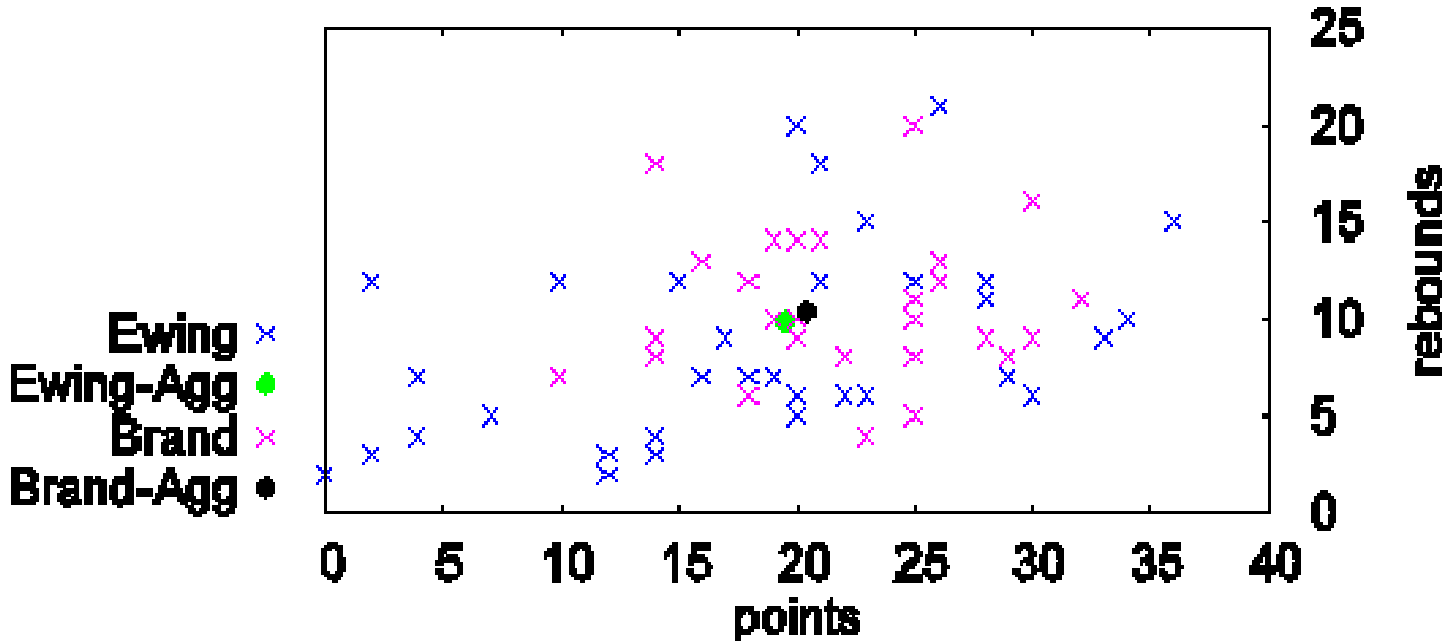


Skylines on Uncertain Data

- Conventional methods compute the skyline on
  - Individual game records
  - Aggregate: mean or median
- Limitations
  - Aggregates may be misled by outliers
  - Data distribution is not captured
- Probabilistic skylines
  - An instance has a probability to represent the object
  - An object has a probability to be in the skyline

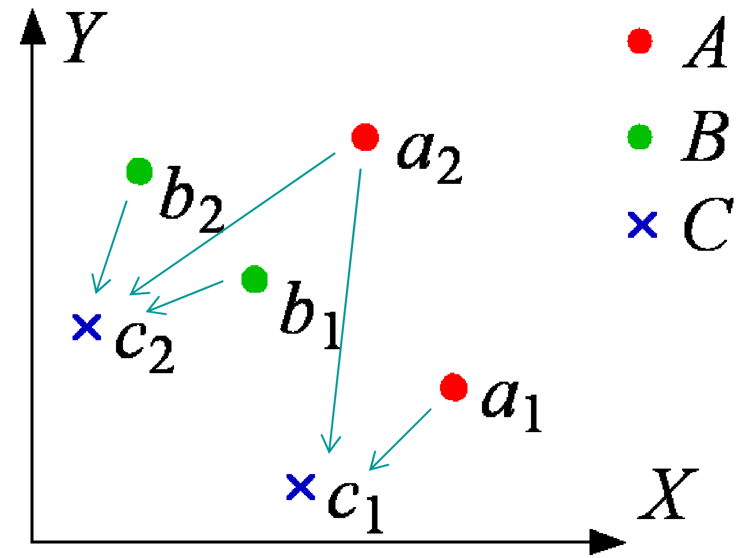
Example

Brand-Agg (20.39, 2.67, 10.37)
Ewing-Agg (19.48, 1.71, 9.91)



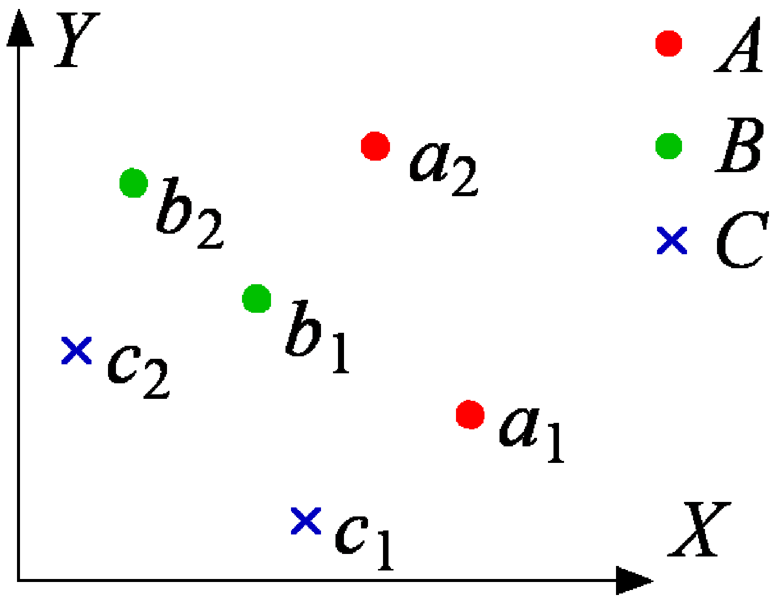
A Probabilistic Skyline Model

- A set of object S = {A, B, C}, each instance takes a probability (0.5) to appear
- Probabilistic Dominance
  - $\Pr(A \succ C) = 3/4$
  - $\Pr(B \succ C) = 1/2$
  - $\Pr((A \succ C) \vee (B \succ C)) = 1$
  - $\Pr(C \text{ is in the skyline}) \neq (1 - \Pr(A \succ C)) \times (1 - \Pr(B \succ C))$
  - Probabilistic dominance $\not\Rightarrow$ Probabilistic skyline



Skyline Probabilities

- Possible world: $W = \{a_i, b_j, c_k\}$ (i, j, k = 1 or 2)
  - $Pr(W) = 0.5 \times 0.5 \times 0.5 = 0.125$, $\sum W \in \Omega\ Pr(W) = 1$
- $SKY(\{a_1, b_1, c_1\}) = \{a_1, b_1\}$
  - A and B are in $SKY(\{a_1, b_1, c_1\})$
- B is in the skyline of possible worlds $\{a_1, b_1, c_1\}$, $\{a_1, b_1, c_2\}$, $\{a_1, b_2, c_1\}$, and $\{a_1, b_2, c_2\}$
  - $Pr(B) = 4 \times 0.125 = 0.5$
- $Pr(A) = 1$, $Pr(C) = 0$



Problem Statement

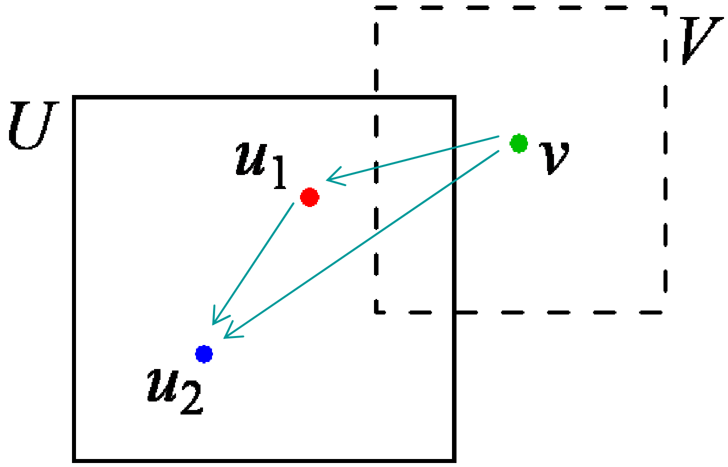
- Skyline probability: $Pr(U) = \sum_{U \in SKY(W)} Pr(W)$
- For object: $Pr(U) = \frac{1}{|U|} \sum_{u \in U} \prod_{V \neq U} (1 - \frac{|\{v \in V \mid v \succ U\}|}{|V|})$
- For instance: $Pr(u) = \prod_{V \neq U} (1 - \frac{|\{v \in V \mid v \succ u\}|}{|V|})$
- $Pr(U) = \frac{1}{|U|} \sum_{u \in U} Pr(u)$
- p-skyline = {U | Pr(U) $\geqslant$ p} for a given threshold p

Probabilistic Skyline Computation

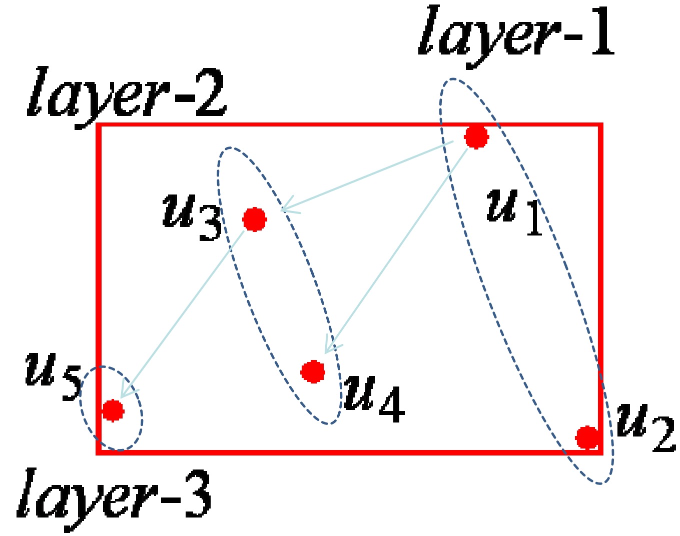
- Iteration: Bounding-Pruning-Refining
- Bounding
  - Bound $Pr(u)$: lower bound $Pr^{-}(u)$ and upper bound $Pr^{+}(u)$
  - Bound $Pr(U)$: $Pr(U) = \frac{1}{|U|} \sum_{u \in U} Pr(u)$
- Pruning
  - In p-skyline if lower bound $Pr^{-}(U) \geq p$
  - Not in p-skyline if upper bound $Pr^{+}(U) < p$
- Refining
  - Bottom-up method
  - Top-down method

The Bottom-Up Method

- Key idea: sort the instances of an object according to the dominance relation such that their skyline probabilities are in descending order
- Two instances u_1 and $u_2 \in U$, if $u_1 \succ u_2$, then $Pr(u_1) \geq Pr(u_2)$



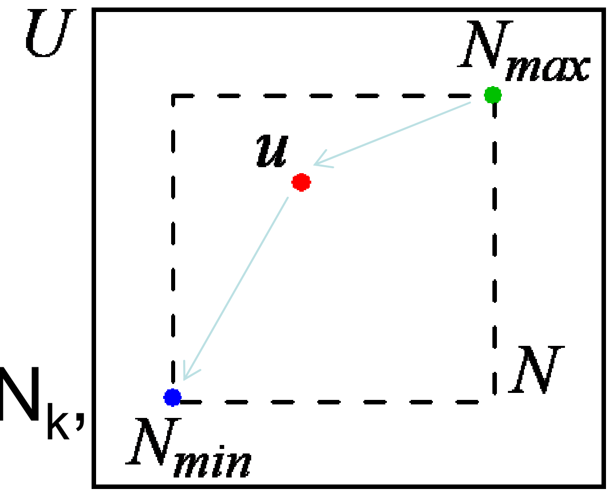
The Layer Structure



- layer-1: the skyline of all instances
- layer-k (k > 1): the skyline of instances except those at layer-1, …, layer-(k-1)
- $\forall$ u at layer-k, $\exists$ u' at layer-(k-1) such that u' $\succ$ u and Pr(u') ≥ Pr(u)
- max{Pr(u) | u is at layer-(k-1)} ≥ max{Pr(u) | u is at layer-k}
- Bounding
  - max{Pr(u1), Pr(u2)} ≥ max{Pr(u3), Pr(u4)} ≥ Pr(u5)

The Top-Down Method

- For instances u_1 and $u_2 \in U$, if $u_1 \succ u_2$, then $Pr(u_1) \geq Pr(u_2)$
  – N is a subset of instances of U, $\forall u \in N$, $Pr(N_{max}) \geq Pr(u) \geq Pr(N_{min})$



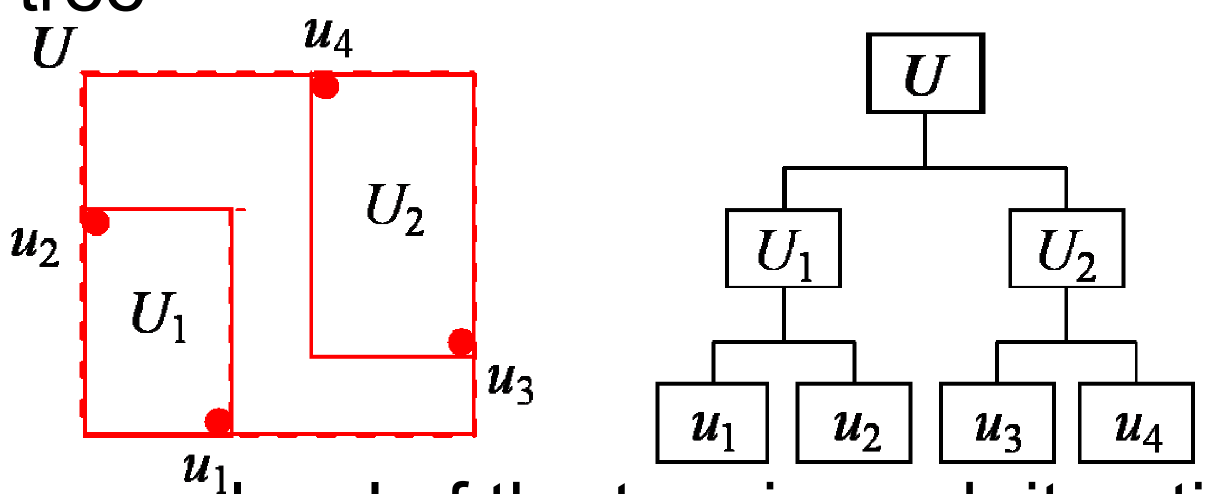
- Object U has k partitions $N_1, \ldots, N_k$,

$$\frac{1}{|U|}\sum_{i=1}^{k}|N_i| \cdot Pr(N_{i,max}) \geq Pr(U) \geq \frac{1}{|U|}\sum_{i=1}^{k}|N_i| \cdot Pr(N_{i,min})$$

- Build a partition tree for each object to organize partitions

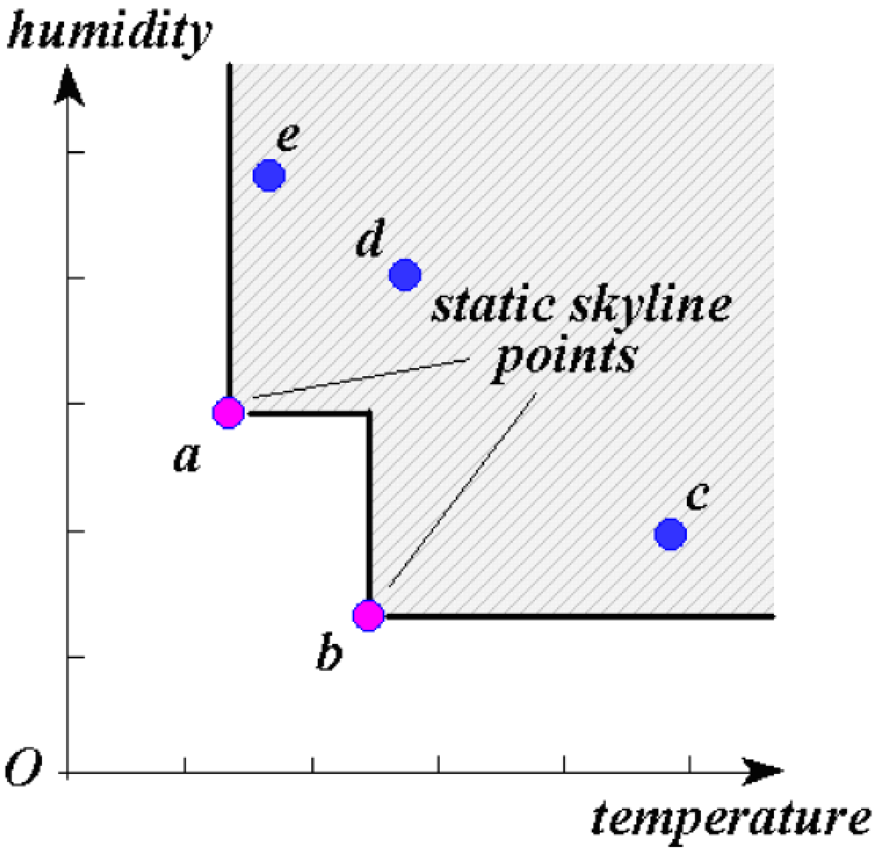
Partition Tree

- Binary tree

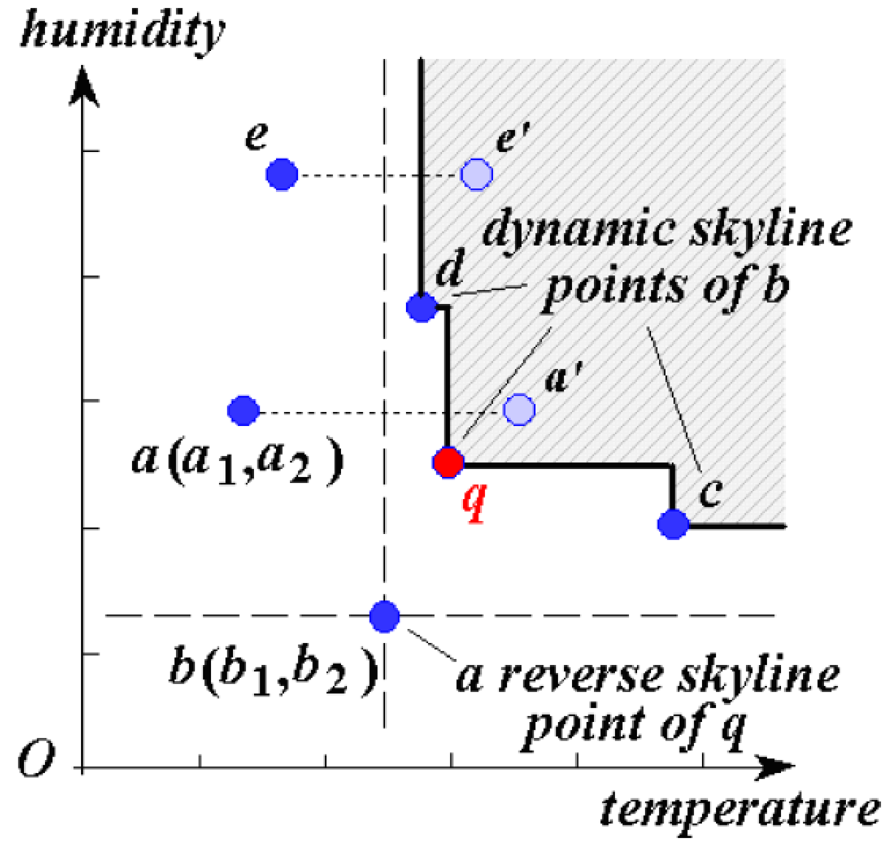


- Growing one level of the tree in each iteration
  - Choose one dimension in a round-robin fashion
  - Each leaf node is partitioned into two children nodes, each of which has half of instances
- Bound $Pr(N_{max})$ and $Pr(N_{min})$ of a partition N

Skyline and Dynamic Skyline



(a) Static Skyline



(b) Dynamic/Reverse Skyline

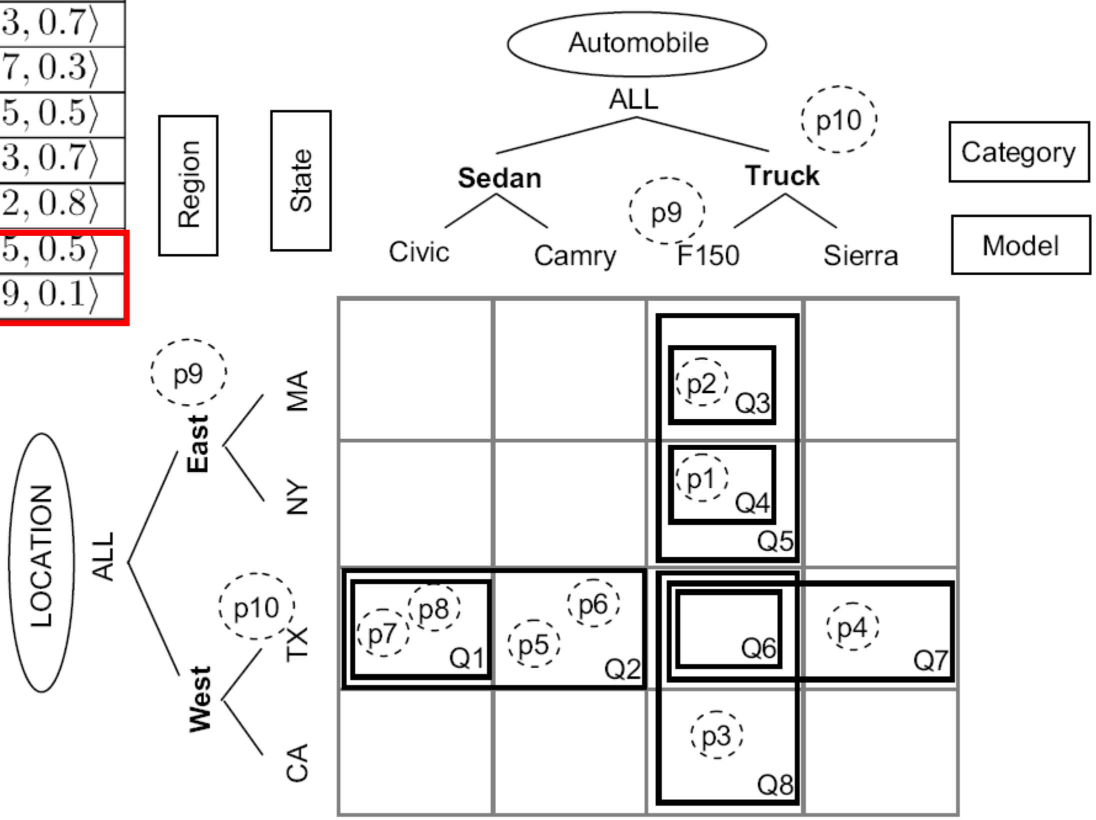
Reverse Dynamic Skyline Queries

- Given a query point q, find the set of objects whose dynamic skyline contains q
- Monochromatic probabilistic reverse skyline queries: find the uncertain objects whose dynamic skylines contain a query object with a probability passing a threshold
- Bichromatic probabilistic reverse skyline queries: given two distinct uncertain objects A and B and a query point q, find points o in A such that the dynamic skyline of o in B contains q
- Details in [Lian and Chen, SIGMOD'08]

OLAP Query

What are the total repair cost for F150's in the East?

	Auto	*Loc*	*Repair*	*Text*	*Brake*
p1	F-150	NY	\$200	...	$\langle 0.8, 0.2 \rangle$
p2	F-150	MA	\$250	...	$\langle 0.9, 0.1 \rangle$
p3	F-150	CA	\$150	...	$\langle 0.7, 0.3 \rangle$
p4	Sierra	TX	\$300	...	$\langle 0.3, 0.7 \rangle$
p5	Camry	TX	\$325	...	$\langle 0.7, 0.3 \rangle$
p6	Camry	TX	\$175	...	$\langle 0.5, 0.5 \rangle$
p7	Civic	TX	\$225	...	$\langle 0.3, 0.7 \rangle$
p8	Civic	TX	\$120	...	$\langle 0.2, 0.8 \rangle$
p9	F150	East	\$140	...	$\langle 0.5, 0.5 \rangle$
p10	Truck	TX	\$500	...	$\langle 0.9, 0.1 \rangle$

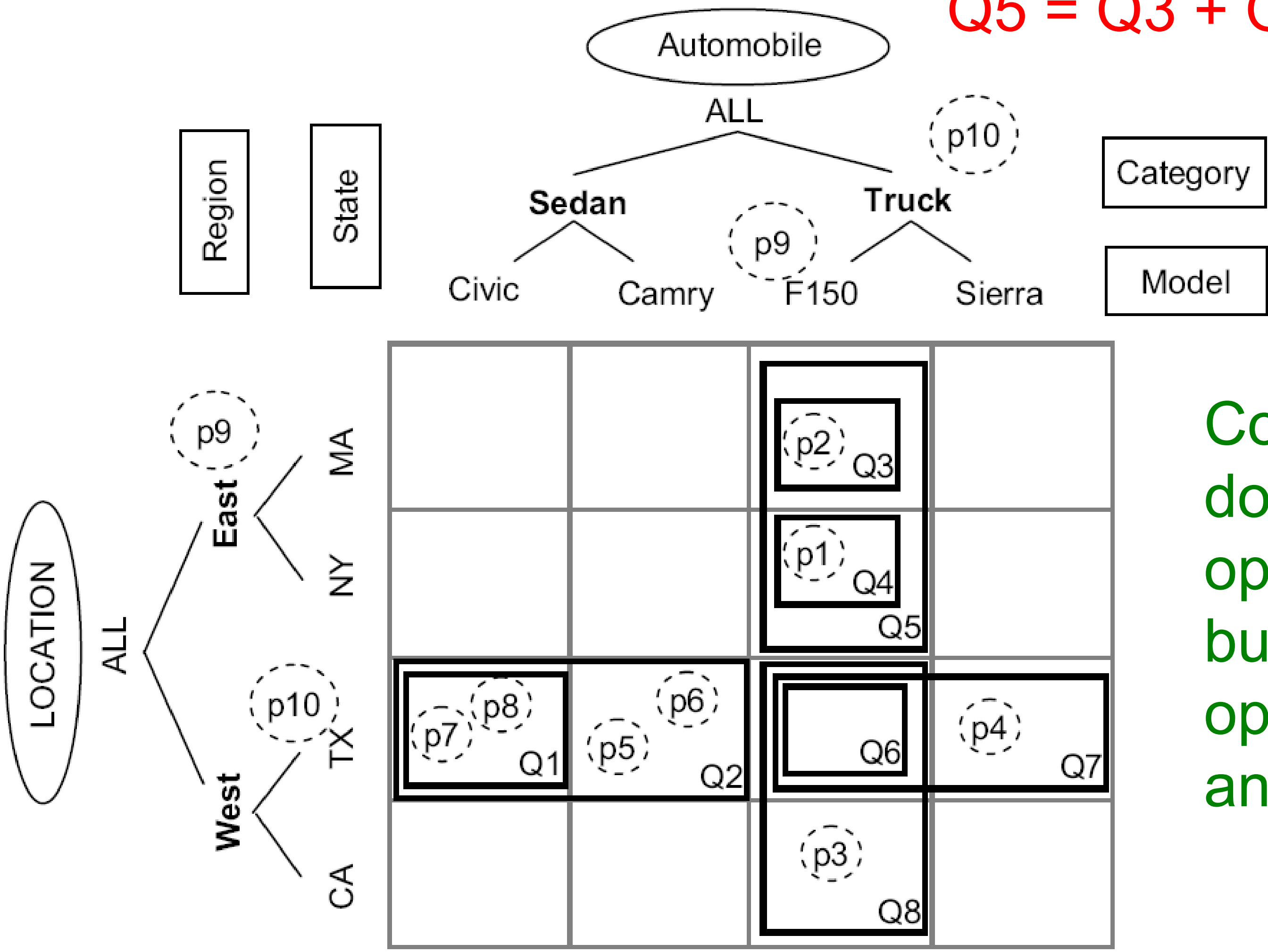


Three Options

- None: ignore all imprecise facts
  - Answer: p1, p2
- Contains: include only those contained in the query region
  - Answer: p1, p2, p9
- Overlaps: include all imprecise facts whose region overlaps the query region
  - Answer: p1, p2, p9, p10

Consistency among OLAP Queries

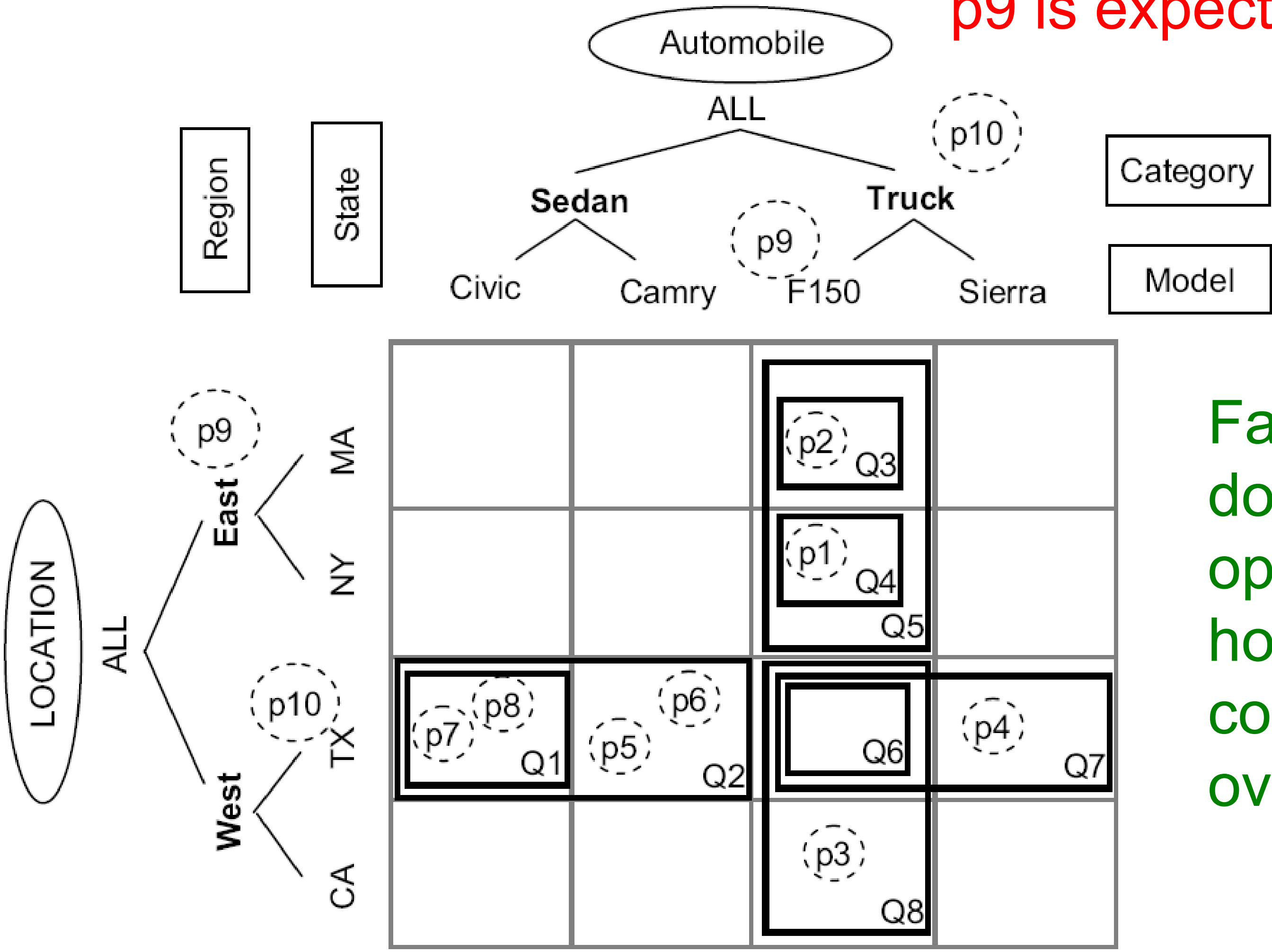
Q5 = Q3 + Q4 is expected!



Consistency does not hold for option contains, but holds for options none and overlaps!

Faithfulness of OLAP Queries

p9 is expected in Q5!

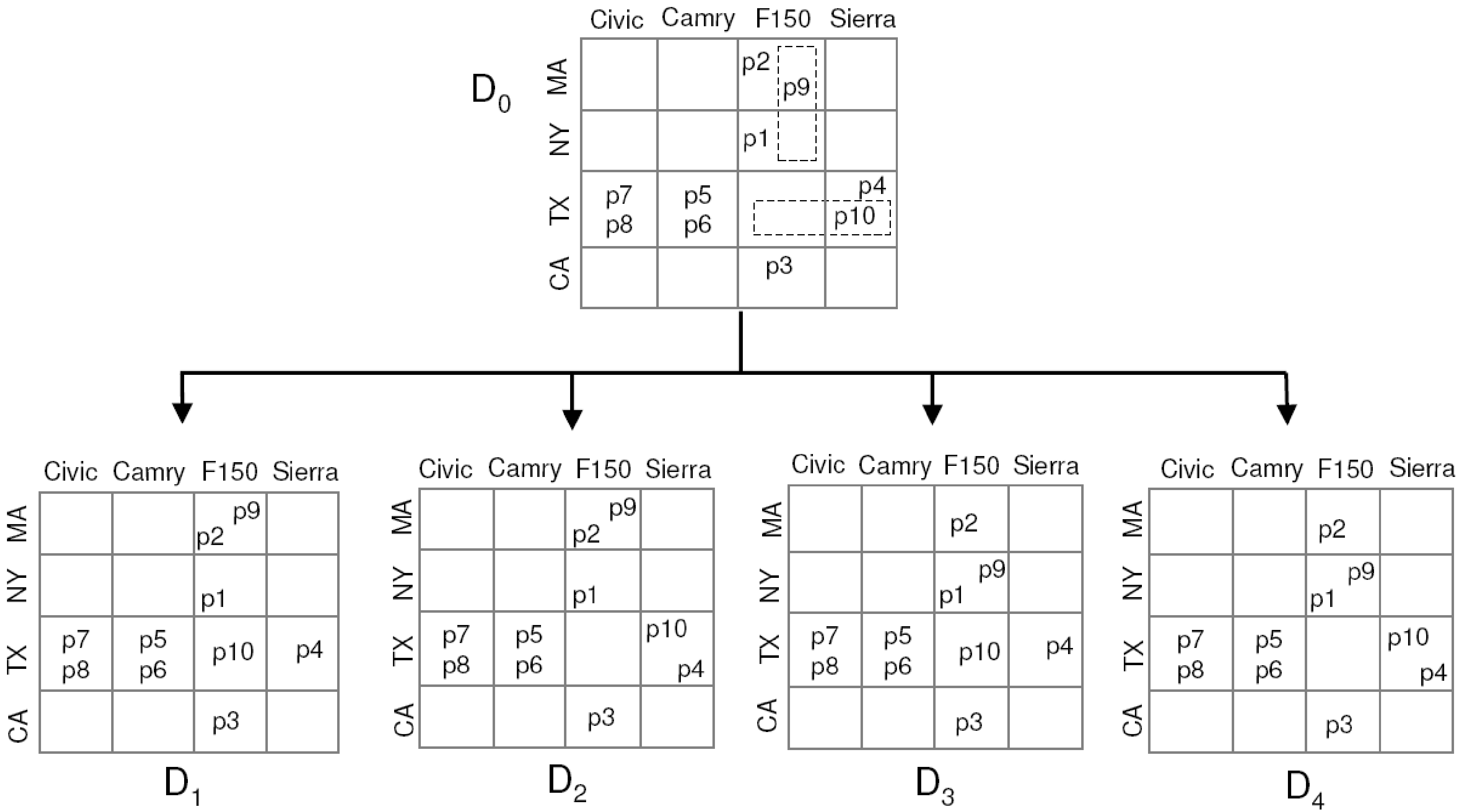


Faithfulness does not hold for option none, but holds for options contains and overlaps!

OLAP Requirements

- Consistency (summarizability): some natural relationships hold between answers to aggregation queries associated with different (connected) regions in a hierarchy
- Faithfulness: imprecise data should be considered properly in query answering

Possible Worlds



Allocation and Query Answering

- The allocation weights encode a set of possible worlds $D_1, \ldots, D_m$ with associated weights $w_1, \ldots, w_m$
- The answer to a query is a multiset $\{v_1, \ldots, v_m\}$
- Problem: how to summarize $\{v_1, \ldots, v_m\}$ properly?

Answer Variable

- Consider multiset $\{v_1, \ldots, v_m\}$ of possible answers to a query Q
- Define the answer variable Z associated with Q to be a random variable with probability density function

$$Pr[Z=v_i]=\Sigma_{j \text{ s.t. } vi=vj} w_j,\ 1 \leq i, j \leq m$$

Answer Variable

- The answer to a query can be summarized as the first and the second moments (expected value and variance) of the answer variable Z
- Basic faithfulness is satisfied if answers to queries are computed using the expected value of the answer variable

Query Answering

- Identify the set of candidate facts and compute the corresponding allocations to Q
  - Identifying candidate facts: using a filter for the query region
  - Computing the corresponding allocations: identifying groups of facts that share the same identifier in the ID column, then summing up the allocations within each group
- Identify the information necessary to compute the summarization while circumventing the enumeration of possible worlds

Allocation Policies

- Dimension-independent allocation such as uniform allocation
- Measure-oblivious allocation such as count-based allocation
  - If Vancouver and Victoria have 100 and 50 F150’s, respectively, and there are another 30 in BC as imprecise records, then allocate 20 and 10 to Vancouver and Victoria, respectively

Outline

- Introduction: motivations, applications and challenges
- Models and possible worlds
- Range search queries
- Ranking queries
- Advanced queries
- Summary: challenges and future directions

Uncertain and Probabilistic Data

- Uncertainty is inherent in many applications
  - Sensor networks, mobile equipment, social data
- Modeling uncertain and probabilistic data
  - Individual objects: probability distribution function (PDF) or a set of sampled instances
  - Distribution/configuration of a set of objects: possible worlds
  - Enumerating all possible worlds is exponential

Query Answering on Uncertain Data

- Range queries
- Ranking queries
- Advanced queries
  - Joins
  - Skyline queries
  - OLAP queries
- We apologize that many recent studies cannot be covered in this 2 hour talk

Future Directions

- Uncertain and probabilistic data processing is a fast-growing track
- How to extend well accepted queries on certain data to undertain and probabilistic data
  - K-nearest neighbor search, reverse nearest neighbor search, continuous nearest neighbor search, …
- Novel types of queries
  - U-kRank queries, queries about probability information
- Efficient/fast/scalable query answering algorithms
  - Extending heuristics on certain data to uncertain data
  - Theoretical analysis

References(1)

- Parag Agrawal, Omar Benjelloun, Anish Das Sarma, Chris Hayworth, Shubha U. Nabar, Tomoe Sugihara, and Jennifer Widom. Trio: A system for data, uncertainty, and lineage. In VLDB, pages 1151–1154, 2006.
- Parag Agrawal and Jennifer Widom. Confidence-aware joins in large uncertain databases. Technical report, Stanford University CA, USA.
- Lyublena Antova, Christoph Koch, and Dan Olteanu. 10^10^6 worlds and beyond: Efficient representation and processing of incomplete information. In ICDE, pages 606–615, 2007.
- Daniel Barbará, Hector Garcia-Molina, and Daryl Porter. The management of probabilistic data. IEEE Trans. Knowl. Data Eng., 4(5):487–502, 1992.
- Omar Benjelloun, Anish Das Sarma, Chris Hayworth, and Jennifer Widom. An introduction to uldbs and the trio system. IEEE Data Eng. Bull., 29(1):5–16, 2006.

References(2)

- Christian Böhm, Alexey Pryakhin, and Matthias Schubert. The gauss-tree: Efficient object identification in databases of probabilistic feature vectors. In ICDE, page 9, 2006.
- Douglas Burdick, Prasad Deshpande, T. S. Jayram, Raghu Ramakrishnan, and Shivakumar Vaithyanathan. Olap over uncertain and imprecise data. In VLDB, pages 970–981, 2005.
- Douglas Burdick, Prasad M. Deshpande, T. S. Jayram, Raghu Ramakrishnan, and Shivakumar Vaithyanathan. Efficient allocation algorithms for olap over imprecise data. In VLDB, pages 391–402, 2006.
- Doug Burdick, AnHai Doan, Raghu Ramakrishnan, and Shivakumar Vaithyanathan. Olap over imprecise data with domain constraints. In VLDB, pages 39–50, 2007.
- Roger Cavallo and Michael Pittarelli. The theory of probabilistic databases. In VLDB, pages 71–81, 1987.

References(3)

- Reynold Cheng, Dmitri V. Kalashnikov, and Sunil Prabhakar. Evaluating probabilistic queries over imprecise data. In SIGMOD Conference, pages 551–562, 2003.
- Reynold Cheng, Dmitri V. Kalashnikov, and Sunil Prabhakar. Querying imprecise data in moving object environments. IEEE Trans. Knowl. Data Eng., 16(9):1112–1127, 2004.
- Reynold Cheng, Sarvjeet Singh, Sunil Prabhakar, Rahul Shah, Jeffrey Scott Vitter, and Yuni Xia. Efficient join processing over uncertain data. In CIKM, pages 738–747, 2006.
- Reynold Cheng, Yuni Xia, Sunil Prabhakar, Rahul Shah, and Jeffrey Scott Vitter. Efficient indexing methods for probabilistic threshold queries over uncertain data. In VLDB, pages 876–887, 2004.
- Nilesh N. Dalvi and Dan Suciu. Efficient query evaluation on probabilistic databases. In VLDB, pages 864–875, 2004.

References(4)

- Nilesh N. Dalvi and Dan Suciu. Answering queries from statistics and probabilistic views. In VLDB, pages 805–816, 2005.
- Nilesh N. Dalvi and Dan Suciu. The dichotomy of conjunctive queries on probabilistic structures. In PODS, pages 293–302, 2007.
- Anish Das Sarma, Martin Theobald, and Jennifer Widom. Exploiting lineage for confidence computation in uncertain and probabilistic databases. Technical report, Stanford University CA, USA.
- Lise Getoor. An introduction to probabilistic graphical models for relational data. IEEE Data Eng. Bull., 29(1):32–39, 2006.
- Todd J. Green and Val Tannen. Models for incomplete and probabilistic information. IEEE Data Eng. Bull., 29(1):17–24, 2006.

References(5)

- Ming Hua, Jian Pei, Wenjie Zhang, and Xuemin Lin. Efficiently answering probabilistic threshold top-k queries on uncertain data (extended abstract). In ICDE, pages 1403–1405, 2008.
- Ming Hua, Jian Pei, Wenjie Zhang, and Xuemin Lin. Ranking queries on uncertain data: A probabilistic threshold approach. In SIGMOD, Vancouver, Canada, 2008.
- Benny Kimelfeld and Yehoshua Sagiv. Maximally joining probabilistic data. In PODS, pages 303–312, 2007.
- Hans-Peter Kriegel, Peter Kunath, Martin Pfeifle, Matthias Renz. Probabilistic Similarity Join on Uncertain Data. In DASFAA 2006, pages 295-309.
- Xiang Lian and Lei Chen. Monochromatic and bichromatic reverse skyline search over uncertain databases. In SIGMOD, Vancouver, Canada, 2008.
- Xiang Lian and Lei Chen. Probabilistic ranked queries in uncertain databases. In EDBT, pages 511-522, 2008.

References(6)

- Vebjorn Ljosa and Ambuj K. Singh. Apla: Indexing arbitrary probability distributions. In ICDE, pages 946–955, 2007.
- Michi Mutsuzaki, Martin Theobald, Ander de Keijzer, Jennifer Widom, Parag Agrawal, Omar Benjelloun, Anish Das Sarma, Raghotham Murthy, and Tomoe Sugihara. Trio-one: Layering uncertainty and lineage on a conventional dbms (demo). In CIDR, pages 269–274, 2007.
- Jian Pei, Bin Jiang, Xuemin Lin, and Yidong Yuan. Probabilistic skylines on uncertain data. In VLDB, pages 15–26, 2007.
- Christopher Re, Nilesh N. Dalvi, and Dan Suciu. Query evaluation on probabilistic databases. IEEE Data Eng. Bull., 29(1):25–31, 2006.
- Christopher Re, Nilesh N. Dalvi, and Dan Suciu. Efficient top-k query evaluation on probabilistic data. In ICDE, pages 886–895, 2007.

References(7)

- Christopher Re and Dan Suciu. Materialized views in probabilistic databases for information exchange and query optimization. In VLDB, pages 51–62, 2007.
- A. Das Sarma, O. Benjelloun, A. Halevy, S.U. Nabar, and J. Widom. Representing uncertain data: Models, properties, and algorithms. Technical report, Stanford University CA, USA.
- Adam Silberstein Silberstein, Rebecca Braynard, Carla Ellis, Kamesh Munagala, and Jun Yang. A sampling-based approach to optimizing top-k queries in sensor networks. In ICDE, page 68, 2006.
- Sarvjeet Singh, Chris Mayfield, Sunil Prabhakar, Rahul Shah, and Susanne E. Hambrusch. Indexing uncertain categorical data. In ICDE, pages 616–625, 2007.
- Mohamed A. Soliman, Ihab F. Ilyas, and Kevin Chen-Chuan Chang. Top-k query processing in uncertain databases. In ICDE, pages 896–905, 2007.

References(8)

- Yufei Tao, Reynold Cheng, Xiaokui Xiao, Wang Kay Ngai, Ben Kao, and Sunil Prabhakar. Indexing multi-dimensional uncertain data with arbitrary probability density functions. In VLDB, pages 922–933, 2005.
- Yufei Tao, Xiaokui Xiao, and Reynold Cheng. Range search on multidimensional uncertain data. ACM Trans. Database Syst., 32(3):15, 2007.
- Jennifer Widom. Trio: A system for integrated management of data, accuracy, and lineage. In CIDR, pages 262–276, 2005.
- Ke Yi, Feifei Li, Divesh Srivastava, and George Kollios. Efficient processing of top-k queries in uncertain databases. In ICDE, pages 1406–1408, 2008.
- Xi Zhang and Jan Chomicki. On the semantics and evaluation of top-k queries in probabilistic databases. In ICDE Workshops, pages 556–563, 2008.