Probabilistic/Uncertain Data Management

1. *Dalvi, Suciu. "Efficient query evaluation on probabilistic databases", VLDB'2004.*
2. *Das Sarma et al. "Working models for uncertain data", ICDE'2006.*

Slides based on the Suciu/Dalvi SIGMOD'05 tutorial

What is a Probabilistic Database ?

- "An item belongs to the database" is a probabilistic event
  - Tuple-existence uncertainty
  - Attribute-value uncertainty
- "A tuple is an answer to the query" is a probabilistic event
- Can be extended to all data models; we discuss only probabilistic *relational* data

Possible Worlds Semantics

Attribute domains:

int, char(30), varchar(55), datetime

values: 2^{32}, 2^{120}, 2^{440}, 2^{64}

Relational schema:

Employee(name:varchar(55), dob:datetime, salary:int)

# of tuples: $2^{440} \times 2^{64} \times 2^{23}$

# of instances: $2^{2^{440} \times 2^{64} \times 2^{23}}$

Database schema:

Employee(. . .), Projects(. . .), Groups(. . .), WorksFor(. . .)

# of instances: N (= BIG but finite)

The Definition

The set of all possible database instances:

$$\boxed{\text{INST} = \{I_1, I_2, I_3, \ldots, I_N\}}$$

will use Pr or I^p interchangeably

Definition A *probabilistic database* I^p is a probability distribution on INST

$$\boxed{\Pr : \text{INST} \rightarrow [0,1]} \qquad \text{s.t. } \sum_{i=1,N} \Pr(I_i) = 1$$

Definition A *possible world* is I s.t. $\Pr(I) > 0$

Query Semantics

Given a query Q and a probabilistic database I^p, what is the meaning of $Q(I^p)$?

Query Semantics

Semantics 1: Possible Answers

A probability distribution on *sets of tuples*

$$\forall A.\ \Pr(Q = A) = \sum_{I \in INST.\ Q(I) = A} \Pr(I)$$

Semantics 2: Possible Tuples

A probability function on *tuples*

$$\forall t.\ \Pr(t \in Q) = \sum_{I \in INST.\ t \in Q(I)} \Pr(I)$$

Example: Query Semantics

Purchase^p

Name	City	Product
John	Seattle	Gizmo
John	Seattle	Camera
Sue	Denver	Gizmo
Sue	Denver	Camera

$Pr(I_1) = 1/3$

Name	City	Product
John	Boston	Gizmo
Sue	Denver	Gizmo
Sue	Seattle	Gadget

$Pr(I_2) = 1/12$

Name	City	Product
John	Seattle	Gizmo
John	Seattle	Camera
Sue	Seattle	Camera

$Pr(I_3) = 1/2$

Name	City	Product
John	Boston	Camera
Sue	Seattle	Camera

$Pr(I_4) = 1/12$

```
SELECT DISTINCT x.product
FROM Purchase^p x, Purchase^p y
WHERE x.name = 'John'
      and x.product = y.product
      and y.name = 'Sue'
```

Possible answers semantics:

Answer set	Probability	
Gizmo, Camera	1/3	$Pr(I_1)$
Gizmo	1/12	$Pr(I_2)$
Camera	7/12	$P(I_3) + P(I_4)$

Possible tuples semantics:

Tuple	Probability	
Camera	11/12	$Pr(I_1)+P(I_3) + P(I_4)$
Gizmo	5/12	$Pr(I_1)+Pr(I_2)$

Possible Worlds Query Semantics

Possible answers semantics

- Precise
- Can be used to compose queries
- Difficult user interface

Possible tuples semantics

- Less precise, but simple; sufficient for most apps
- Cannot be used to compose queries
- Simple user interface

Possible Worlds Semantics: Summary

Complete model; Clean formal semantics for SQL queries

Not very useful as a representation or implementation tool

- HUGE number of possible worlds!

Need more effective representation formalisms

- Something that users can understand/explore
- Allow more efficient query execution
  - Avoid "possible worlds explosion"
- *Perhaps giving up completeness*

Representation Formalisms

Problem

Need a good representation formalism

- Will be interpreted as possible worlds
- Several formalisms exists, but no winner

Main open problem in probabilistic db

Evaluation of Formalisms

Completeness?

- What possible worlds can it represent?
- What probability distributions on worlds?

Closure?

- Is it closed under evaluation of query operators?

Outline

A complete formalism:

- *Intensional Databases*

Incomplete formalisms:

- Various expressibility/complexity tradeoffs
- Focus on *Explicit Independent Tuples*

[Fuhr&Roelleke:1997]

Intensional Database

Atomic event ids	$e_1, e_2, e_3, \ldots$
Probabilities:	$p_1, p_2, p_3, \ldots \in [0,1]$
Event expressions: $\wedge, \vee, \neg$	$e_3 \wedge (e_5 \vee \neg e_2)$

Intensional probabilistic database J:
each tuple t has an event attribute t.E

Intensional DB $\Rightarrow$ Possible Worlds

J =

Name	**Address**	**E**
John	Seattle	$e_1 \wedge (e_2 \vee e_3)$
Sue	Denver	$(e_1 \wedge e_2) \vee (e_2 \wedge e_3)$

$e_1e_2e_3=$ 000 001 010 011 100 101 110 111

I^p

$\emptyset$

$(1-p_1)(1-p_2)(1-p_3)$
$+(1-p_1)(1-p_2)p_3$
$+(1-p_1)p_2(1-p_3)$
$+p_1(1-p_2)(1-p_3$

John	Seattle

$p_1(1-p_2)\ p_3$

Sue	Denver

$(1-p_1)p_2\ p_3$

John	Seattle
Sue	Denver

$p_1p_2(1-p_3)$
$+p_1p_2p_3$

Possible Worlds ⇒ Intensional DB

Name	Address
John	Seattle
John	Boston
Sue	Seattle

p_1

Name	Address
John	Seattle
Sue	Seattle

p_2

Name	Address
Sue	Seattle

p_3

Name	Address
John	Boston

p_4

$=I^p$

$E_1 = e_1$ $\quad Pr(e_1) = p_1$

$E_2 = \neg e_1 \wedge e_2$ $\quad Pr(e_2) = p_2/(1-p_1)$

$E_3 = \neg e_1 \wedge \neg e_2 \wedge e_3$ $\quad Pr(e_3) = p_3/(1-p_1-p_2)$

$E_4 = \neg e_1 \wedge \neg e_2 \wedge \neg e_3 \wedge e_4$ $\quad Pr(e_4) = p_4/(1-p_1-p_2-p_3)$

"Prefix code"

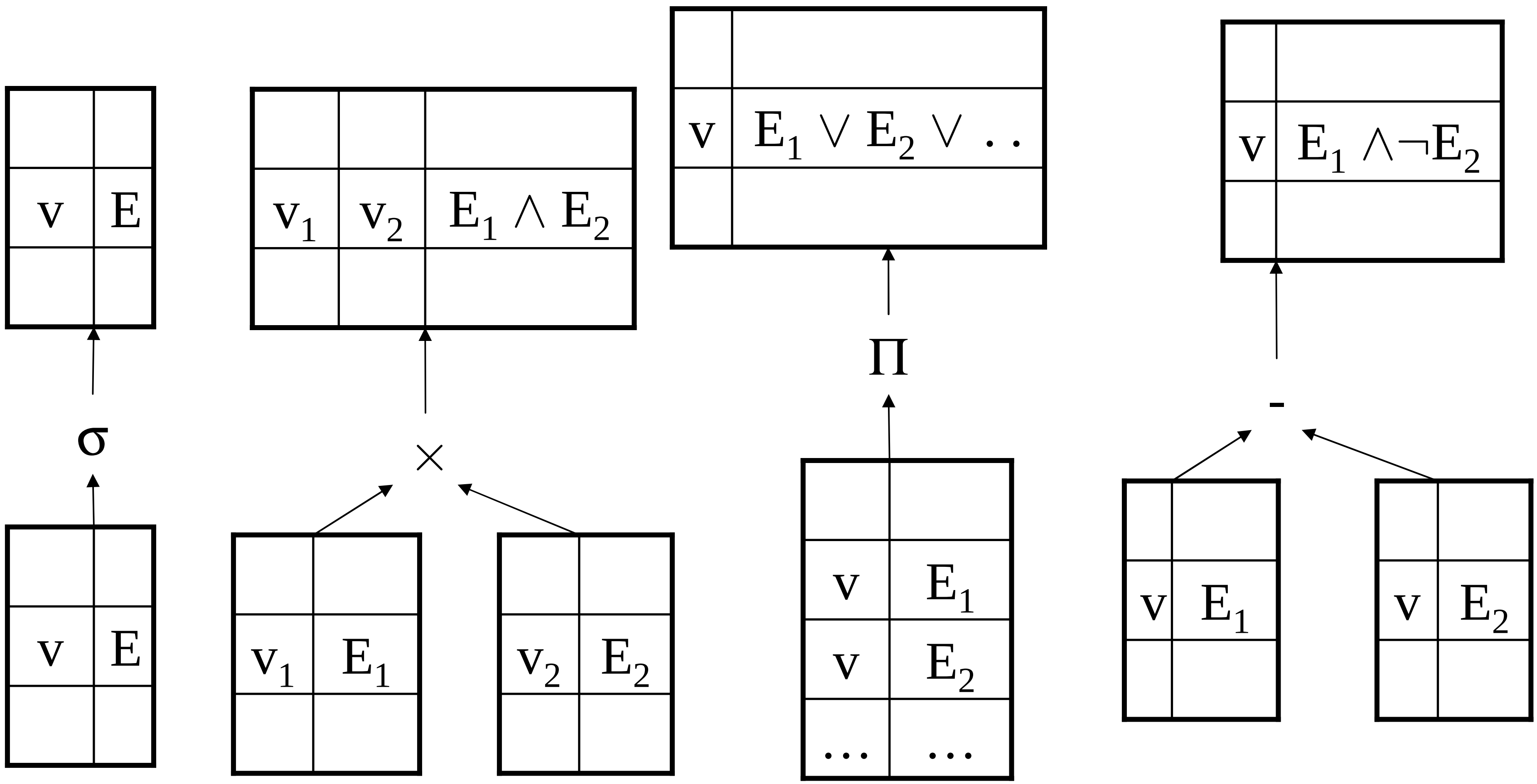
$J =$

Name	Address	E
John	Seattle	$E_1 \vee E_2$
John	Boston	$E_1 \vee E_4$
Sue	Seattle	$E_1 \vee E_2 \vee E_3$

Intensional DBs are complete

[Fuhr&Roelleke:1997]

Closure Under Operators



One still needs to compute probability of event expression

Summary on Intensional Databases

Event expression for each tuple

- Possible worlds: any subset
- Probability distribution: any

Complete… but impractical

- Evaluate the probability of *long* event expressions

Important abstraction: consider restrictions

Related to *c-tables* [Imilelinski&Lipski:1984]

Restricted Formalisms

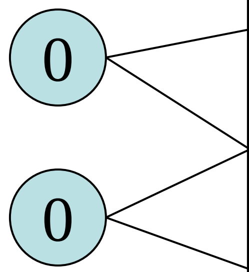
Explicit tuples

- Have a tuple template for every tuple that may appear in a possible world
- Focus on the case of *independent* tuple events

Explicit Independent Tuples

tuple = independent event

Atomic, distinct.
May use TIDs.



independent

Name	**City**	**E**	**pr**
John	Seattle	e_1	0.8
Sue	Boston	e_2	0.6
Fred	Boston	e_3	0.9

Can be easily extended to capture *attribute-value uncertainty*

Explicit Independent Tuples

Tuple independent probabilistic database

INST = $\mathcal{P}$(TUP)
N = 2^M

$\boxed{\text{TUP} = \{t_1, t_2, \ldots, t_M\}}$ = all tuples

$\boxed{\text{pr} : \text{TUP} \rightarrow [0,1]}$

No restrictions

$$\boxed{\Pr(I) = \prod_{t \in I} \text{pr}(t) \times \prod_{t \notin I} (1-\text{pr}(t))}$$

Tuple Prob. $\Rightarrow$ Possible Worlds

J =

Name	City	pr
John	Seattle	$p_1 = 0.8$
Sue	Boston	$p_2 = 0.6$
Fred	Boston	$p_3 = 0.9$

E[size(I^p)] = 2.3 tuples

I^p =

I_1: $\emptyset$ → $(1-p_1)(1-p_2)(1-p_3)$

I_2:

Name	City
John	Seattl

→ $p_1(1-p_2)(1-p_3)$

I_3:

Name	City
Sue	Bosto

→ $(1-p_1)p_2(1-p_3)$

I_4:

Name	City
Fred	Bosto

→ $(1-p_1)(1-p_2)p_3$

I_5:

Name	City
John	Seattl
Sue	Bosto

→ $p_1p_2(1-p_3)$

I_6:

Name	City
John	Seattl
Fred	Bosto

→ $p_1(1-p_2)p_3$

I_7:

Name	City
Sue	Bosto
Fred	Bosto

→ $(1-p_1)p_2p_3$

I_8:

Name	City
John	Seattl
Sue	Bosto
Fred	Bosto

→ $p_1p_2p_3$

$\sum = 1$

Tuple-Independent DBs are Incomplete

Name	Address	pr
John	Seattle	p_1
Sue	Seattle	p_2

Name	Address
John	Seattle

p_1

Name	Address
John	Seattle
Sue	Seattle

$p_1 p_2$

$= I^p$

$\emptyset$ $1 - p_1 - p_1 p_2$

Very limited – cannot capture correlations across tuples

Not Closed

- Query operators can introduce complex correlations!

Tuple Prob. $\Rightarrow$ Query Evaluation

Name	**City**	**pr**
John	Seattle	p_1
Sue	Boston	p_2
Fred	Boston	p_3

Customer	**Product**	**Date**	**pr**
John	Gizmo	. . .	q_1
John	Gadget	. . .	q_2
John	Gadget	. . .	q_3
Sue	Camera	. . .	q_4
Sue	Gadget	. . .	q_5
Sue	Gadget	. . .	q_6
Fred	Gadget	. . .	q_7

```
SELECT DISTINCT x.city
FROM Person x, Purchase y
WHERE x.Name = y.Customer
      and y.Product = 'Gadget'
```

Tuple	**Probability**
Seattle	$p_1(1-(1-q_2)(1-q_3))$
Boston	$1-(1-p_2(1-(1-q_5)(1-q_6)))$ $\times(1-p_3\,q_7)$

Application 1: Similarity Predicates

Name	City	Profession
John	Seattle	statistician
Sue	Boston	musician
Fred	Boston	physicist

Step 1:
evaluate ~ predicates

Cust	Product	Category
John	Gizmo	dishware
John	Gadget	instrument
John	Gadget	instrument
Sue	Camera	musicware
Sue	Gadget	microphone
Sue	Gadget	instrument
Fred	Gadget	microphone

SELECT DISTINCT x.city
FROM Person x, Purchase y
WHERE x.Name = y.Cust
  and y.Product = ‘Gadget’
  and **x.profession ~ ‘scientist’**
  and **y.category ~ ‘music’**

Application 1: Similarity Predicates

Name	City	Profession	pr
John	Seattle	statistician	$p_1=0.8$
Sue	Boston	musician	$p_2=0.2$
Fred	Boston	physicist	$p_3=0.9$

Step 1:
evaluate ~ predicates

Cust	Product	Category	pr
John	Gizmo	dishware	$q_1=0.2$
John	Gadget	instrument	$q_2=0.6$
John	Gadget	instrument	$q_3=0.6$
Sue	Camera	musicware	$q_4=0.9$
Sue	Gadget	microphone	$q_5=0.7$
Sue	Gadget	instrument	$q_6=0.6$
Fred	Gadget	microphone	$q_7=0.7$

```
SELECT DISTINCT x.city
FROM Person^p x, Purchase^p y
WHERE x.Name = y.Cust
      and y.Product = 'Gadget'
      and x.profession ~ 'scientis
      and y.category ~ 'music'
```

Step 2:
evaluate rest
of query

Tuple	Probability
Seattle	$p_1(1-(1-q_2)(1-q_3))$
Boston	$1-(1-p_2(1-(1-q_5)(1-q_6)))$ $\times(1-p_3q_7)$

Summary on Explicit Independent Tuples

Independent tuples

- Possible worlds: subsets
- Probability distribution: restricted
- Closure: no