An Algebra for Probabilistic Databases

Michael Pittarelli
mike@cs.sunyit.edu

Abstract

An algebra is presented for a simple probabilistic data model that may be regarded as an extension of the standard relational model. The probabilistic algebra is developed in such a way that (restricted to α-acyclic database schemes) the relational algebra is a homomorphic image of it. Strictly probabilistic results are emphasized. Variations on the basic probabilistic data model are discussed. The algebra is used to explicate a commonly used statistical smoothing procedure and is shown to be potentially very useful for decision support with uncertain information.

Index Terms – Bayes and Markov networks, data models, decision support, probability, relational algebra.

I. Introduction

Beginning in the mid–1960s, researchers in systems theory, influenced by Ashby [2], Lewis [22] and others, began work on techniques for *reconstructability analysis* of finite-variable relational and probabilistic systems [20]. These methods are primarily aimed at identifying collections of subsystems into which a system may be (nearly-) losslessly decomposed.

Noting the parallels between reconstructability analysis and database theory, Cavallo and Pittarelli [9] introduced a probabilistic model of data generalizing the relational model (by replacing the characteristic function of a relation with a finite probability distribution function). Since then, models have been proposed allowing probability intervals [34], embedding of multiple distributions within a single "probabilistic relation" and incompletely specified distributions [3].

In what follows, the relative expressive power of these models is discussed. An outline is sketched of a probabilistic algebra analogous to the relational algebra. The utility of this algebra for the construction of probabilistic decision support systems is illustrated. The algebra is used to explicate a commonly applied probability estimation technique. Connections to Bayes and Markov network research are also noted.

II. Probabilistic Data Models

In the standard relational model, a *relational database instance* is informally viewed as a collection of tables. Each column of a table is associated with an *attribute* that can take on any of a finite number of values. Each row is a sequence of these values.

Example II.1: The tables below represent a simple relational database in the customary format. (See Maier [26].)

production	(Plant	Type	Output)	quality	(Plant	Acceptable)
	Lubbock	Chain	Medium		Lubbock	Yes
	Lubbock	Sprocket	Low		Waco	No
	Waco	Chain	High			
	Waco	Sprocket	High			

□

More formally, a relational database instance is a (finite) collection of relations on finite domains; i.e., a set of subsets of Cartesian products of finite sets. Let dom(A) denote the domain of attribute A. Then the relation instance ’quality’, for example, is a subset of the product set dom(Plant)×dom(Acceptable). As observed by Nambiar [30], it is often advantageous to work with the *characteristic function* associated with a relation. In this paper, a relation is identified with its characteristic function. Let V be the set of attributes (*relational scheme*) for relation instance r. Let $\mathrm{dom(V)} = \times_{A \in V} \mathrm{dom(A)}$. Then $\mathrm{r{:}dom(V)} \rightarrow \{0, 1\}$, where, for any tuple $\mathrm{t} \in \mathrm{dom(V)}$, r(t)=1 if and only if t is a member of relation instance r. So,

$$\text{production(Lubbock, Chain, Medium)} = 1,$$

$$\text{production(Lubbock, Chain, High)} = 0, \text{ etc.}$$

(In a tabular representation of the characteristic function, those tuples t of dom(V) for which r(t)=0 are omitted.

A relational database is thus a collection $\mathrm{R} = \{\mathrm{r}_1, \ldots, \mathrm{r}_m\}$, where $\mathrm{r}_i{:}\mathrm{dom(V}_i) \rightarrow \{0, 1\}$. The set $\{\mathrm{V}_1, \ldots, \mathrm{V}_m\}$ is the *database scheme* or *structure* [26] for R. A *probabilistic database* is a collection $\mathrm{P} = \{\mathrm{p}_1, \ldots, \mathrm{p}_m\}$, where $\mathrm{p}_i{:}\mathrm{dom(V}_i) \rightarrow [0, 1]$ and $\sum_{t \in dom(V_i)} \mathrm{p}_i(\mathrm{t}) = 1$.

Example II.2: The database $\{\mathrm{p}_1, \mathrm{p}_2\}$ below represents a (fictional) pooled sample of 100 parts from two different manufacturing plants.

Type	Plant	p_1(t)	Plant	Defective	p_2(t)
Chain	Lubbock	16/100	Lubbock	No	27/100
Chain	Waco	42/100	Lubbock	Yes	2/100
Sprocket	Lubbock	13/100	Waco	No	48/100
Sprocket	Waco	29/100	Waco	Yes	23/100

An entry p_i(t)=x may be interpreted as stating that the relative frequency with which a part from the sample possesses the tuple of attributes t is x. □

The set $\{p_1,p_2\}$ of Example II.2 is a probabilistic database of the type discussed by Cavallo and Pittarelli [9]. Extensions to this model have been proposed. Pittarelli [34] considers *interval-valued* probabilistic databases, in which distribution functions map tuples to closed subintervals of the real interval [0, 1].

Example II.3:

Color	Shape	i(t)	
Black	Sphere	[0.3, 0.6]	
Black	Cube	[0.1, 0.4]	
White	Sphere	[0, 0.3]	
White	Cube	[0, 0.3]	□

The set of intervals $\{i(t) | t \in dom(V)\}$ is regarded as a collection of linear inequality constraints on real-valued distributions over dom(V). For distribution i of Example II.3, the associated set of real-valued distributions is the set of solutions p to the system

$$p(\text{Black, Sphere}) \geq 0.3$$

$$p(\text{Black, Sphere}) \leq 0.6$$

$$\ldots$$

$$p(\text{White, Cube}) \geq 0$$

$$p(\text{White, Cube}) \leq 0.3$$

Values i(t) may be confidence intervals constructed from frequencies (e.g. the data of Example II.2) or imprecisely stated subjective probabilities determined by introspection or elicited from experts. They may also be derived from knowledge of lower-dimensional (real- or interval-valued) distributions, as discussed in Section III.

Barbara′, Garcia-Molina and Porter [3] propose a model in which multiple probability distributions may be contained within a single *probabilistic relation*. These relations have deterministic keys. In addition, it is possible to incompletely specify the distributions. Two examples follow.

Example II.4 [3]:

Employee	Department	Quality Bonus	Sales
Jon Smith	Toy	0.4 [Great Yes]	0.3 [30-34K]
		0.5 [Good Yes]	0.7 [35-39K]
		0.1 [Fair No]	
Fred Jones	Houseware	1.0 [Good Yes]	0.5 [20-24K]
			0.5 [25-29K] □

Example II.5 [3]:

Student	GPA	Interest	Accept Evaluation
Adam	3.8	0.7 [theory]	0.6 [Y A]
		0.3 [*]	0.1 [N A]
			0.3 [* *]
Eve	3.9	0.6 [database]	0.5 [Y A]
		0.4 [systems]	0.3 [Y B]
			0.2 [Y C]

The entry 0.3 [*] above is referred to as a *missing probability*. It is considered to be distributed in some unknown fashion among the values of dom(Interest), including ’theory’. □

Any probabilistic relation without missing probabilities may be represented (somewhat awkwardly) by a real-valued Cavallo-Pittarelli database: For each independent attribute or cluster of attributes (over each of which a probability distribution is specified) construct a probability distribution over the key attributes and the cluster by dividing the given probabilities by the number of distinct key values in the active domain of the probabilistic relation. To recover the original probabilistic relation, multiply each probability by the number of active key values (i.e., key values of the tuples assigned positive probability) and collect the resulting distributions into a single table, grouped by key value. (This may be accomplished by means of the probabilistic select operator, Section III.D.) Whether or not such a transformation is applied, the algebra of Section III is applicable to the distributions in a collection of probabilistic relations.

Example II.6: The probabilistic relation of Example II.4 is represented as the database

Employee	Department	$p_1(t)$
Jon Smith	Toy	0.5
Fred Jones	Houseware	0.5

Employee	Quality	Bonus	$p_2(t)$
Jon Smith	Great	Yes	0.2
Jon Smith	Good	Yes	0.25
Jon Smith	Fair	No	0.05
Fred Jones	Good	Yes	0.5

Employee	Sales	$p_3(t)$
Jon Smith	30-34K	0.15
Jon Smith	35-39K	0.35
Fred Jones	20-24K	0.25
Fred Jones	25-29K	0.25

□

Probabilistic relations with missing probabilities (probabilities assigned to tuples containing wildcard values, denoted by asterisks) can be represented by an interval-valued probabilistic database [34]. The probability assigned to a tuple without wildcard components is interpreted as the lower endpoint of the probability interval associated with the tuple. The sum of this value and the probabilities assigned to matching wildcard tuples is the upper endpoint. (Probabilities x of distributions without missing probabilities are represented as intervals [x, x].) Divide each interval endpoint by the number of key tuples in the active domain. The result is an interval probability distribution.

Example II.7: Transformation of Example II.5:

Student	GPA	$i_1(t)$
Adam	3.8	[0.5,0.5]
Eve	3.9	[0.5,0.5]

Student	Interest	$i_2(t)$
Adam	theory	[0.35,0.5]
Adam	t∈dom(Interest)-{theory}	[0,0.15]
Eve	database	[0.3,0.3]
Eve	systems	[0.2,0.2]

Student	Accept	Evaluation	$i_3(t)$
Adam	Y	A	[0.3,0.45]
Adam	N	A	[0.05,0.2]
Adam	t∈(dom(Acc.)×dom(Eval.))-{(Y,A),(N,A)}		[0,0.15]
Eve	Y	A	[0.25,0.25]
Eve	Y	B	[0.15,0.15]
Eve	Y	C	[0.1,0.1]

□

A standard relational database instance may also be transformed via an injective mapping to a Cavallo-Pittarelli probabilistic database: For non-empty relations r,

$$t_{rp}(r)(t) = r(t)/\sum_t r(t).$$

For a database $R = \{r_1, \ldots, r_m\}$, $\{t_{rp}(r_1), \ldots, t_{rp}(r_m)\}$ is abbreviated $t_{rp}(R)$. Let P_V and R_V denote the set of all distributions and the set of all non-empty relations, respectively, over dom(V). Since P_V is infinite and R_V is finite, t_{rp} does not have a two-sided inverse. Further, it has infinitely many left inverses, functions f from distributions to relations with the property

$$f(t_{rp}(r)) = r.$$

A reasonable choice is the (onto, total) function t_{pr} under which a tuple is included in the resulting relation if and only if it is assigned non-zero probability. In Section III, t_{pr} is shown to be a homomorphism from probabilistic to relational systems defined in terms of standard relational operators and their probabilistic analogues. In [9] it is shown that the transformation t_{rp} preserves standard relational data dependencies (functional, join, etc., in probabilistic form (characterized in terms of conditional and relative entropy); the notion of approximate satisfaction of relational and probabilistic dependencies is also discussed there.

Unless stated otherwise, all subsequent uses of the term "probabilistic database" refer to the real-valued Cavallo-Pittarelli model [9].

III. Probabilistic Data Algebra

A fairly small set of probabilistic operators is discussed. Three – *projection*, *selection*, and (maximum entropy) *join* – are shown to be formally analogous to the correspondingly named relational operators. (Linear) *pooling*, a widely used method of reconciling differing expert probability assessments, is shown to be analogous to relational union. *Extension* has a relational counterpart, but the probabilistic version seems to be more useful (e.g. for decision making, as discussed in Section IV.A). Most of the results derived in this section are strictly probabilistic and are shown in Section IV to have practical applications.

A. Models and Projection

For tuples $w \in dom(W)$ and $b \in dom(B)$, $B \subseteq W$, $w[B]=b$ iff w and b agree on all attributes in scheme B. The *projection* of p with scheme V onto $A \subseteq V$ is the distribution $\pi_A(p)$, where

$$\pi_A(p)(a) = \sum_{t \in dom(V),\, t[A]=a} p(t).$$

Marginal probabilities $p(S)$, $S \subseteq dom(V)$, are computed as

$$p(S) = \textstyle\sum_{t \in S} p(t).$$

With tuples $t \in dom(V)$ viewed as disjoint events, these definitions follow from the additivity of probability.

Relational projection may be defined in terms of the characteristic function as

$$\pi_A(r)(a) = \max_{t \in dom(V),\, t[A]=a} r(t).$$

It follows trivially from these definitions that

$$t_{pr}(\pi_A(p)) = \pi_A(t_{pr}(p)), \qquad \textbf{(Eq. III.1)}$$

i.e., that t_{pr} is a homomorphism from (P_V, P_A, π_A) to (R_V, R_A, π_A), and that

$$\pi_A(r) = t_{pr}(\pi_A(t_{rp}(r))). \qquad \textbf{(Eq. III.2)}$$

Lemma 1. $\pi_V(p) = p$, if V is the scheme for p. □

Lemma 2. $A \subseteq B$ implies $\pi_A(\pi_B(p)) = \pi_A(p)$. □

The corresponding result for relations [26] may be derived from the above:

Theorem 3. $\pi_A(\pi_B(\mathrm{r})) = \pi_A(\mathrm{r})$, if $A \subseteq B$.

Proof: From t_{pr} onto, Eqs III.1 and III.2, Lemma 2, and the observation

$$\pi_A(t_{rp}(t_{pr}(\pi_B(\mathrm{p})))) = 0 \text{ iff } \pi_A(\pi_B(\mathrm{p})) = 0. \ \square$$

A *model* of a scheme V is a structure $X=\{V_1, \ldots, V_m\}$ such that $\bigcup_{j=1}^{m} V_j \subseteq V$ and $V_i \not\subset V_j$ for all $i,j \in \{1,...,m\}$. (X will sometimes be referred to as a model of a distribution with scheme V.) If X is also a cover of V, then X is a *reduced hypergraph* over V [26]. Normally, attention is restricted to reduced hypergraph models of a given scheme V.

A distribution with scheme V may be projected onto a model $X=\{V_1, \ldots, V_m\}$ of V to form a probabilistic database

$$\pi_X(\mathrm{p})=\{\pi_{V_1}(\mathrm{p}), \ldots, \pi_{V_m}(\mathrm{p})\}.$$

Example III.1: The database of Example II.2 is the projection $\pi_X(\mathrm{p})$ of the distribution below, with X = {{Type, Plant}, {Plant, Defective}}.

Type	Plant	Defective	p(t)	
Chain	Lubbock	No	15/100	
Chain	Lubbock	Yes	1/100	
Chain	Waco	No	22/100	
Chain	Waco	Yes	20/100	
Sprocket	Lubbock	No	12/100	
Sprocket	Lubbock	Yes	1/100	
Sprocket	Waco	No	26/100	
Sprocket	Waco	Yes	3/100	□

A useful partial ordering on models is the *refinement relation* [8]. A structure X is a *refinement* of structure Y (and Y is an *aggregate* or *coarsening* of X), denoted $X \leq Y$, iff for each $V_x \in X$ there exists a $V_y \in Y$ such that $V_x \subseteq V_y$. For example, {{A},{B,C}} is a refinement of {{A,B},{B,C},{D}}. The set of all models over V together with the refinement ordering is a lattice. Any pair of models has a greatest lower bound equal to their least refined common refinement and a least upper bound equal to the most refined structure of which they are both refinements. The universal upper bound of the lattice of models over V is {V}; the lower bound is {∅}. For the (sub)lattice of reduced hypergraphs, the universal lower bound is $\{\{v\} | v \in V\}$.

A database P with structure Y may be projected onto a refinement X of Y to form a database $\pi_X(\mathrm{P})$ each element of which is a projection of some element of P.

Example III.2: Projecting P={p_1,p_2} of Example II.2 onto the structure {{Type},{Plant},{Defective}} results in the database

Type	$\pi_{\{Type\}}(p_1)(t)$	Plant	$\pi_{\{Plant\}}(p_1)(t)$	Defective	$\pi_{\{Defective\}}(p_2)(t)$
Chain	58/100	Lubbock	29/100	No	75/100
Sprocket	42/100	Waco	71/100	Yes	25/100

□

For a set of distributions D, π_V(D) denotes the image of D under the mapping π_V. Thus, for any family of sets $(D_i)_{i\in I}$,

$$\pi_V(\cup_{i\in I} D_i) = \cup_{i\in I}\pi_V(D_i)$$

and

$$\pi_V(\cap_{i\in I} D_i) \subseteq \cap_{i\in I}\pi_V(D_i).$$

B. Extension

For a distribution p with scheme A, its *extension* to the scheme V, A⊆V, is the set of all preimages of p under the mapping π_A:

$$E^V(p) = \{p'\in P_V | \pi_A(p') = p\}.$$

The extension of a database P is the intersection of the extensions of its elements:

$$E^V(P) = \cap_{p\in P} E^V(p).$$

Thus, $E^V(\pi_X(p))$ is the set of all preimages of the database π_X(p) under the mapping π_X; any model X of V partitions P_V into classes $E^V(\pi_X(p))$ equivalent with respect to projections onto X. If the structure of P is a cover of V, then E^V(P) may be abbreviated E(P). Any E^V(p) or E^V(P) is a convex polyhedron (set of solutions to the system of linear equations determined by the projection conditions). As discussed in Section IV.A, this makes feasible decision support without assumption or computation of a universal instance [26].

Example III.3: The database below represents partial information regarding the contents of a box of wooden blocks.

Color	$p_1(t)$	Shape	$p_2(t)$
Black	0.7	Sphere	0.6
White	0.3	Cube	0.4

Its extension to {Color, Shape} is the set of solutions p to the system

$$p(\text{Black, Sphere}) + p(\text{Black, Cube}) = 0.7$$

$$p(\text{White, Sphere}) + p(\text{White, Cube}) = 0.3$$

$$p(\text{Black, Sphere}) + p(\text{White, Sphere}) = 0.6$$

$$p(\text{Black, Cube}) + p(\text{White, Cube}) = 0.4$$

(The equations imply that $\sum_t$p(t) = 1.) From just the information given, it cannot be determined which of the infinitely many members of E(P) is the actual joint distribution over {Color, Shape} for this box of blocks. □

The extension of a relational database instance may be defined analogously, substituting the maximum operator for addition. For a relational database R, (E(R), ⊆) is a partially ordered set with the natural join of R as maximum element.

Since E(P) is convex, the set of values {p(t)|p∈E(P)} for a given tuple t is an interval. The collection of intervals for each t∈dom(V) is an interval-valued distribution. For the database of Example III.3, the corresponding interval distribution is given as Example II.3. Information is lost when the equations defining E(P) are replaced by the associated intervals. For example, the distribution

Color	Shape	p(t)
Black	Sphere	0.5
Black	Cube	0.1
White	Sphere	0.2
White	Cube	0.2

is consistent with the intervals of Example II.3 but violates each of the equations in Example III.3.

Each element of E(P) is a potential universal instance for P, a distribution p for which π_X(p) = P. Databases whose extensions are nonempty are referred to as *consistent*. As is the case with relational databases, it is not necessary for an otherwise useful probabilistic database to be consistent. Real-valued probabilistic databases that are not constructed by projection of a given distribution onto a model are in fact likely to be inconsistent. (Methods for reconciling inconsistent sets of distributions have been studied [28].)

For databases with α-acyclic structures [12], a sufficient condition for consistency is that each pair of distributions agree on projections onto shared attributes [39]. So, the database of Example II.2 is consistent:

$$\pi_{\{Plant\}}(\mathrm{p}_1) = \pi_{\{Plant\}}(\mathrm{p}_2).$$

Extension is complementary to projection in the weak sense that p∈$\mathrm{E}^V(\pi_A(\mathrm{p}))$, where V is the scheme for p. Several useful results follow easily from the definitions of projection and extension:

Lemma 4. If V is the scheme for p and V⊆W⊆S, then $\pi_W(\mathrm{E}^S(\mathrm{p})) = \mathrm{E}^W(\mathrm{p})$. □

Lemma 5. $\pi_V(\mathrm{p}) \in \mathrm{E}^V(\pi_X(\mathrm{p}))$. □

Lemma 6. $\mathrm{E}^V(\pi_{\{\emptyset\}}(\mathrm{p})) = \mathrm{P}_V$. □

Lemma 7. $\mathrm{E}^V(\pi_{\{V\}}(\mathrm{p})) = \{\pi_V(\mathrm{p})\}$. □

Lemma 8. If the structure for P is a cover of V, then $\mathrm{V} \subseteq \mathrm{S}$ implies $\pi_V(\mathrm{E}^S(\mathrm{P})) \subseteq \mathrm{E}(\mathrm{P})$.

Proof: $\pi_V(\mathrm{E}^S(\mathrm{P})) = \pi_V(\cap_{p \in P} \mathrm{E}^S(\mathrm{p}))$ [Def. Extension]

$\subseteq \cap_{p \in P} \pi_V(\mathrm{E}^S(\mathrm{p}))$

$= \cap_{p \in P} \mathrm{E}^V(\mathrm{p})$ [Lemma 4]

$= \mathrm{E}^V(\mathrm{P})$ [Def. Extension] □

Theorem 9. $\mathrm{X} \le \mathrm{Y}$ implies $\mathrm{E}^V(\pi_Y(\mathrm{p})) \subseteq \mathrm{E}^V(\pi_X(\mathrm{p}))$. □

Proof [9]: $\mathrm{E}^V(\pi_X(\mathrm{p}))$ is the set of all solutions to the linear system determined by the projection of p onto the structure X. If X≤Y, then each equation determined by the projection of p onto X is a linear combination of equations in the system determined by the projection of p onto Y; thus, all solutions to the latter system are also solutions to the first.□

If $\mathrm{E}(\pi_X(\mathrm{p})) = \{\mathrm{p}\}$, then p is said to be *identifiable* from (its projections onto) X. (A distribution is never identifiable from a model that is not a cover of its scheme.) The smaller the set $\mathrm{E}(\pi_X(\mathrm{p}))$ the more information regarding p is contained in the projections $\pi_X(\mathrm{p})$. From Theorem 9, if X≤Y, more information is recoverable from Y than from X for any distribution p for which X and Y are models. (Unfortunately, a randomly selected pair of structures is unlikely to be comparable under ≤; so it usually cannot be determined *a priori* which of two models of a distribution will be more informative in this sense.) From Lemmas 1 and 7, any p with scheme V is identifiable from the structure {V}. By Lemma 6, no information is contained in $\mathrm{E}(\pi_{\{\emptyset\}}(\mathrm{p}))$ that would distinguish p from any other element of P_V.

Ashby and Madden [25] investigate conditions under which relations are identifiable from projections and conclude that they are met extremely rarely. *A fortiori*, this is the case also for probability distributions. Methods for picking a single universal instance from the (almost certainly infinite) set E(P) are discussed next.

C. Join and Decomposition

It may be argued that the sole reason for estimating a probability distribution is to base a decision on it. Given that methods exist (Section IV.A) for basing decisions on sets of distributions, why select a single universal instance from E(P)? There are many situations where it is reasonable to do so.

A consistent database may be more compactly represented as a universal instance if its structure is relatively unrefined. For any database P with structure X, P = π_X(p), for any p$\in$E(P). So P is recoverable from any of its universal instances without loss of information. Suppose X consists of all (n–1)-element subsets of a set of n binary attributes. Then n/2 times as many numbers are required to represent P as are needed for any p$\in$E(P). (Of course, this works the other way, too. If the structure X is relatively refined, then storage or transmission of π_X(p) is cheaper than that of p. This motivated the earliest published research in what could be considered probabilistic database theory [5, 22]. However, as discussed below, there usually does not exist a non-trivial model from projections onto which a distribution may be recovered.)

It may be known that certain relations of (conditional) probabilistic independence hold among the attributes of the database. If these relations and the marginal probabilities p_i(t) are taken as exact, then a unique p$\in$E(P) may be inferred.

Let P = $\{p_1, p_2\}$, with structure $\{V_1, V_2\}$. The (*pairwise*) *join* of P is the probability distribution J(P) $\in$ E(P) whose components are calculated as

$$J(P)(t) = p_1(a)\times p_2(b) / \sum_{c|c[V_1\cap V_2]=b[V_1\cap V_2]} p_2(c),$$

where a=t[V_1] and b=t[V_2]. (The denominator of the above expression equals 1 if $V_1\cap V_2 = \varnothing$.)

Example III.4: The join of the database in Example II.2 is the distribution:

Type	Plant	Defective	J($\{p_1, p_2\}$)(t)	
Chain	Lubbock	No	0.149	= (16/100×27/100)/(27/100+2/100)
Chain	Lubbock	Yes	0.011	
Chain	Waco	No	0.284	
Chain	Waco	Yes	0.136	
Sprocket	Lubbock	No	0.121	
Sprocket	Lubbock	Yes	0.009	
Sprocket	Waco	No	0.196	
Sprocket	Waco	Yes	0.094	□

For sets of variables V_1, V_2, and V_3, V_1 is *conditionally independent* of V_2, given V_3, iff $p(t_{12}|t_3) = p(t_1|t_3)\times p(t_2|t_3)$, for all $t_{12}\in dom(V_1\cup V_2)$ and $t_3\in dom(V_3)$, where $t_1 = t_{12}[V_1]$ and $t_2 = t_{12}[V_2]$. It follows immediately that:

Theorem 10. For a model $\{V_1, V_2\}$ of p with scheme $V_1 \cup V_2$, $p = J(\pi_{\{V_1,V_2\}}(p))$ iff $V_1-(V_1\cap V_2)$ and $V_2-(V_1\cap V_2)$ are conditionally independent, given $V_1\cap V_2$. □
For the industrial parts example, since the distributions of Examples III.1 and III.4 are not equal, the attributes Type and Defective are not conditionally independent given Plant.

More generally, the *join* of a database $P = \{p_1, \ldots, p_m\}$ is the element $J^V(P)$ of $E^V(P)$ with *maximum entropy*:

$$H(J^V(P)) = \max_{p \in E^V(P)} H(p),$$

where

$$H(p) = -\sum_t p(t) \times \log(p(t)).$$

(Normally, the structure of P is a cover of V. In this case, or when the context makes clear what set of variables is intended, the superscript is dropped.) Recall that for any $p' \in E(\pi_X(p))$, $\pi_X(p') = \pi_X(p)$. Thus:

Lemma 11. $\pi_X(J(\pi_X(p))) = \pi_X(p)$. □

Since the maximum entropy element of an extension E(P) is unique [19], it follows that:

Theorem 12. $J(\pi_X(p))$ is the unique fixed point of the *project-join mapping* $J \circ \pi_X \colon E(\pi_X(p)) \to E(\pi_X(p))$. □

If $p = J(\pi_X(p))$, then p is said to be *reconstructable* from X. (From Theorem 12, only one of the infinitely many distributions in a non-unit equivalence class of distributions $E(\pi_X(p))$ is reconstructable from X.) If p is identifiable from X, then it is reconstructable from X, but not conversely. (None of the elements of a non-unit $E(\pi_X(p))$ is identifiable from X.)

For any set K with a unique maximum entropy element, for example, $\pi_A(E(P))$, let J(K) denote that element. When K is E(P) for a given P, computation of J(K) is more efficient than in the general case.

Let $P = \{p_1, \ldots, p_m\}$. The result of a sequence of applications

$$J(\cdots(J(J(p_{\sigma(1)}, p_{\sigma(2)}), p_{\sigma(3)})\cdots), p_{\sigma(m)})$$

of the pairwise join procedure, where σ is a permutation of $\{1, \ldots, m\}$, is a *product extension* of P iff it is an element of E(P).

Theorem 13 [22]. If p is a product extension of P, then p = J(P). □

Theorem 14 [39]. If the structure of P is α-acyclic, then a product extension of P may be computed with σ corresponding to the reverse of any order in which elements of the structure of P are eliminated by Graham's algorithm [12]. □

(For α-cyclic structures, an iterative proportional fitting algorithm converges to J(P) [5, 20, 38].)

Thus, if X is α-acyclic,

$$t_{pr}(J(\pi_X(p))) = J(\pi_X(t_{pr}(p))), \qquad \textbf{(Eq. III.3)}$$

i.e., t_{pr} is a homomorphism from $(P_V, J\circ\pi_X)$ to $(R_V, J\circ\pi_X)$, and

$$r = J(\pi_X(r)) \text{ iff } t_{rp}(r) = J(\pi_X(t_{rp}(r))), \qquad \textbf{(Eq. III.4)}$$

i.e., r satisfies the *join dependency* |><|[X] iff $t_{rp}(r)$ is reconstructable from X [9].

For α-cyclic X, there exist relations r for which

$$r \neq J(\pi_X(r))$$

but

$$r = t_{pr}(J(\pi_X(t_{rp}(r)))).$$

On the other hand [9], for no structure X (cyclic or otherwise) is it the case that

$$r = J(\pi_X(r))$$

and

$$r \neq t_{pr}(J(\pi_X(t_{rp}(r)))).$$

So, embedding the relational algebra in the probabilistic via the mapping t_{pr} has the advantage of allowing non-trivial lossless decomposition of more relations.

In reconstructability analysis, two complementary problems are studied. The *identification problem* is to determine from a consistent database $\pi_X(p)$ as much as possible regarding p. Usually, the system of projection equations (with unknowns p(t)) is underdetermined. So all that can be inferred deductively is that $p\in E(\pi_X(p))$. This may be sufficient for decision making (Section IV.A) or if determination of bounds on particular p(t) is all that is required.

The identification problem is a type of *inverse problem* in which data are generated via some non-injective mapping from a set of sources. The problem is to identify, using some reasonable criterion, a best representative element from the usually infinite set of preimages for the given data (in this case, a consistent probabilistic database instance). In all published applications of reconstructability analysis [20], the solution has been to *maximize entropy* within E(P); i.e., to select J(P). The primary reason given is that this is the information-theoretically least bold inference that can be made from the data. Appeal is also made to Jaynes' *concentration theorem*, which has been interpreted as stating that the (relative frequency) distribution J(P) is the most likely to arise from observations satisfying the marginal constraints P and that this likelihood decreases with increasing distance from J(P) [18].

The maximum entropy approach is criticized in [15, 24, 37]. Interestingly, selection of the *centroid* of a set of distributions is advocated in [24]. The centroid, C(P), minimizes the expected squared-error when it is selected as a solution to the identification problem. But C(P) is more difficult to calculate than J(P) [32]. When X = {V} or X = {∅}, J(P) = C(P). In experiments involving approximately 8,000 randomly generated databases with non-trivial structures, the ratio of the squared-error distance between J(P) and C(P) to the squared-error diameter of E(P) was found to be approximately 0.09 [33]. So the join of P, when selection of a single representative element of E(P) is called for, is, all things considered, not an unreasonable choice.

If relations of conditional independence are known to hold for some subset of attributes W, the preimage set may be reduced by calculating joint probabilities p(w) and adding the linear equation

$$\sum_{t|t[W]=w} \mathrm{p(t) = p(w)}$$

to the system for each w∈dom(W). (This corresponds to the embedded join dependency concept of relational database theory [13].) If the independence relations correspond exactly to the (α-acyclic) structure of P, then a unique solution, J(P), is determined.

This leads to consideration of the connections between probabilistic database theory and Bayes/Markov network research [31]. A probabilistic database may be used in conjunction with such networks. The conditional probabilities necessary for propagation may be calculated from the marginal tables. At the same time, the structure of a database needn't reflect the dependency structure (if any) of its attributes. This is the case when, for whatever reason (e.g., constraints on data collection over large groups of attributes simultaneously) data are obtainable only in the form of certain marginal distributions. As discussed in Section IV.A, for decision making it is not necessary to work with a single, numerically determinate probability distribution, as in the standard Bayes/Markov network methodology. Determination of a set (not necessarily the smallest determinable set) of distributions compatible with the data P sometimes suffices. Thus, it is possible to avoid the potential for error incurred by calculating a single distribution, e.g., J(P), when it is not certain that the dependencies implying a single solution actually hold.

The concept of approximate reconstructability is more useful than the corresponding relational concept of approximate join dependency. A probability distribution which it is desired to decompose into marginals is far more likely to be an approximation in the first place than is a relation. Further, if a decision is to be based on the information in a probabilistic database, it is only the ordering of actions by expected utility that matters, which is likely to be insensitive to small variations in the probabilities.

In reconstructability analysis, the degree to which p is reconstructable from X is quantified as $\mathrm{d}(\mathrm{p}, \mathrm{J}(\pi_X(\mathrm{p})))$, where d is *directed divergence* (relative entropy, cross-entropy) [1]:

$$\mathrm{d}(\mathrm{p}, \mathrm{p}') = \textstyle\sum_t \mathrm{p}(\mathrm{t}) \times \log(\mathrm{p}(\mathrm{t})/\mathrm{p}'(\mathrm{t})).$$

The *reconstruction problem* is to search for structures that minimize this quantity and are maximally refined. However, since [17]

$$\mathrm{d}(\mathrm{p}, \mathrm{J}(\pi_X(\mathrm{p})) = \mathrm{H}(\mathrm{J}(\pi_X(\mathrm{p}))) - \mathrm{H}(\mathrm{p}),$$

$\mathrm{X} \le \mathrm{Y}$ implies $\mathrm{d}(\mathrm{p}, \mathrm{J}(\pi_X(\mathrm{p}))) \ge \mathrm{d}(\mathrm{p}, \mathrm{J}(\pi_Y(\mathrm{p})))$. Thus, these two criteria are in conflict. (Search procedures are discussed in [7, 8, 20].)

Although these procedures are used mostly for data analysis, significant storage savings may also be achieved. For n k-ary attributes, storage of p requires k^n numbers, vs. kn for $\{\pi_{\{v_1\}}(\mathrm{p}), \ldots, \pi_{\{v_n\}}(\mathrm{p})\}$. Such dramatic compression might compensate for the resulting information loss.

D. Select and Threshold

Select is a unary operation on P_V: for $\mathrm{S} \subseteq \mathrm{dom}(\mathrm{V})$,

$$\sigma_{\mathrm{S}}(\mathrm{p})(\mathrm{t}) = \begin{cases} 0, & \text{if } \mathrm{t} \notin \mathrm{S} \\ \mathrm{p}(\mathrm{t}) / \sum_{\mathrm{t} \in \mathrm{S}} \mathrm{p}(\mathrm{t}), & \text{otherwise.} \end{cases}$$

$\sigma_X(\mathrm{p})$ is undefined when $\sum_{t \in S} \mathrm{p}(\mathrm{t}) = 0$.

Example III.5: In Example II.6, $\sigma_S(\mathrm{p}_2)$, where $\mathrm{S} = \{\mathrm{t} \mid \mathrm{t}[\mathrm{Employee}] = \mathrm{Jon\ Smith}\}$ is

Employee	Quality	Bonus	$\sigma_S(\mathrm{p}_2)(\mathrm{t})$
Jon Smith	Great	Yes	0.4
Jon Smith	Good	Yes	0.5
Jon Smith	Fair	No	0.1

which corresponds to one of the distributions of the probabilistic relation in Example II.4. □

The mapping t_{pr} is a homomorphism with respect to select also:

$$\sigma_S(\mathrm{t}_{pr}(\mathrm{p})) = \mathrm{t}_{pr}(\sigma_S(\mathrm{p})).$$

The *threshold* operator is unary, and renormalizes a probability distribution after eliminating components failing to exceed a specified value:

$$T_x(p)(t) = \begin{cases} 0, & \text{if } p(t) \leq x \\ p(t)/\sum_{p(t)>x} p(t), & \text{otherwise.} \end{cases}$$

($T_x(p)$ is undefined when $\sum_{p(t)>x} p(t) = 0$.)

The composite mapping $t_{pr} \circ T_\alpha : P_V \rightarrow R_V$ is analogous to the (strong) α-cut operator for fuzzy relations [11].

E.Pooling

Probability distributions are sometimes assessed subjectively; and multiple subjective assessments of a single distribution are sometimes solicited from independent experts. One may wish to combine the estimates into a single distribution. This distribution may in turn be decomposed to form a probabilistic database.

These estimates are usually solicited for the purpose of decision making. Although one could, using the techniques of Section IV.A, work with the entire convex hull of the individual estimates, the standard practice is to select a single distribution from this set, as in the identification problem of reconstructability analysis (Section III.C). A common method (*linear pooling* [29]) is to compute a weighted average of the estimates:

$$p = w_1 p_1 + \ldots + w_k p_k,$$

where $w_i \geq 0$, $\sum_i w_i = 1$, and $w_i > w_j$ iff estimate p_i is judged more trustworthy than estimate p_j.

Let $Lp_{a,b}(p, p') = ap + bp'$, $a, b \geq 0$, $a + b = 1$.

Theorem 15. $Lp_{a,b}(\pi_A(p), \pi_A(p')) = \pi_A(Lp_{a,b}(p, p'))$. □

Theorem 16. For P, P' with structure $\{V_1, \ldots, V_m\}$, let

$$Lp_{a,b}(P, P') = \{Lp_{a,b}(p_1, p'_1), \ldots, Lp_{a,b}(p_m, p'_m)\}.$$

If P and P' are locally consistent, then so is $Lp_{a,b}(P, P')$.□

Theorem 17. For any $p, p' \in P_V$ and $a, b > 0$, $t_{pr}(Lp_{a,b}(p, p')) = t_{pr}(p) \cup t_{pr}(p')$. □

F. Updates

Arbitrary changes to a relation instance may be effected by a sequence of deletions and insertions [26]. These in turn may be characterized algebraically as applications of set difference and union, respectively.

Similarly, insertion and deletion applied to a relative frequency distribution (e.g., distribution p of Example III.1) may be characterized in terms of linear pooling. If c is the number of observations recorded in distribution p, the result of incrementing the relative frequency of tuple t is the distribution

$$\mathrm{Lp}_{c/c+1,\,1/c+1}(\mathrm{p},\mathrm{q}),$$

where $\mathrm{q(t)} = 1$. Allowing pooling with negative parameters, the result of retracting observation t is

$$\mathrm{Lp}_{c/c-1,\,-1/c-1}(\mathrm{p},\mathrm{q}).$$

Besides noting that updating elements of an existing database whose structure is not a partition is likely to generate inconsistencies, the topic of probability updating in a more general sense is beyond the scope of this paper. See the discussion and references in [10, 31].

IV. Applications

A. Decision Support

Techniques for decision making from the information in a probabilistic database may be devised by means of the probabilistic algebra. For a database P with structure X, attention is restricted to decision problems with an event space S constructible from elements of dom(V), $\mathrm{V} = \cup_{V_i \in X} \mathrm{V}_i$, a set of mutually exclusive and exhaustive possible actions A, and a utility function $\mathrm{u}: \mathrm{S}\times\mathrm{A} \to \mathbf{R}$.

When S is a partition of $\mathrm{dom}(\mathrm{V}_o)$ for some $\mathrm{V}_o \subseteq \mathrm{V}_i \in \mathrm{X}$, expected utilities are calculated straightforwardly. In the most complicated case, the required distribution is obtained as

$$\mathrm{p(s)} = \textstyle\sum_{a\in s} \sum_{t[V_o]=a} \mathrm{p}_i(\mathrm{t}),$$

for all $\mathrm{s}\in\mathrm{S}$, by Lemma 2. (For convenience, if $\mathrm{S} = \{\{\mathrm{t}\}\,|\,\mathrm{t}\in\mathrm{dom}(\mathrm{V}_o)\}$, then S is identified with $\mathrm{dom}(\mathrm{V}_o)$.) Similarly, if dependencies permit calculation of a unique distribution p* over V*, $\mathrm{V}_o \subseteq \mathrm{V}^* \subseteq \mathrm{V}$, then

$$\mathrm{p(s)} = \textstyle\sum_{a\in s} \sum_{t[V_o]=a} \mathrm{p}^*(\mathrm{t}).$$

When $\mathrm{V}_o \not\subseteq \mathrm{V}_i \in \mathrm{X}$, the elements p of E(P), the potential universal instances, do not necessarily agree on $\pi_{V_o}(\mathrm{p})$. The strongest inference that can be made is that $\pi_{V_o}(\mathrm{p}) \in \pi_{V_o}(\mathrm{E(P)})$.

Lemma 18. If $\mathrm{V}_o \subseteq \bigcup_{V_i\in X} \mathrm{V}_i$, where X is the structure for P, then $\pi_{V_o}(\mathrm{p}) \in \pi_{V_o}(\mathrm{E(P)})$, for any $\mathrm{p} \in \mathrm{E(P)}$. □

It is not guaranteed, for arbitrary $\mathrm{p}\in\mathrm{E(P)}$, that $\pi_{V_o}(\mathrm{p})$ is contained in any smaller sets that can be constructed by means of the algebra, for example, $\pi_{V_o}(\mathrm{E}(\pi_Y(\mathrm{p}')))$, for $\mathrm{Y} > \mathrm{X}$

and $p'\in E(P)$. When the required distribution is known only to the extent that it is a member of some (non-unit) set K, criteria for decision making with partial information may be applied to identify admissible actions [36]. For any of these criteria, the smaller K is, the more likely it is that a single optimal action will emerge. Let $e_p(a)$ denote the expected utility of action a relative to distribution p:

$$e_p(a) = \textstyle\sum_{s\in S} p(s)\times u(s, a).$$

Suppose it is known only that $p\in K$. The set of expected utilities for a as p ranges over K is

$$U_K(a) = \{e_p(a) \,|\, p\in K\}.$$

When K is convex, $U_K(a)$ is an interval. Further, when K is the solution set of a system of linear equations or inequalities, for example, $\pi_{V_o}(E(P))$, the endpoints may be computed by linear programming.

One criterion for decision making with such information orders actions as

$$a_i > a_j \text{ iff } \min U_K(a_i) > \max U_K(a_j)$$

and eliminates all but the maximal elements of A under this ordering as inadmissible [23]. This criterion will be applied to the problem of Example IV.1. (The same example is analyzed according to a more stringent criterion in [36], with the same result.)

Example IV.1: Suppose P is

I	B	$p_1(t)$	B	C	$p_2(t)$	G	D	$p_3(t)$	E	F	$p_4(t)$
i_1	b_1	0.2	b_1	c_1	0.2	g_1	d_1	0.3	e_1	f_1	0.2
i_1	b_2	0.3	b_1	c_2	0.4	g_1	d_2	0.3	e_1	f_2	0.5
i_2	b_1	0.4	b_2	c_1	0.3	g_2	d_1	0.3	e_2	f_1	0
i_2	b_2	0.1	b_2	c_2	0.1	g_2	d_2	0.1	e_2	f_2	0.3

Let $S = \mathrm{dom}(V_o)$ with $V_o = \{I,C\}$, and let A and u be given as:

	$s_{IC=i_1c_1}$	$s_{i_1c_2}$	$s_{i_2c_1}$	$s_{i_2c_2}$
a_1	50	0	-5	1000
a_2	0	10	20	0
a_3	400	0	0	10

Calculating expected utility endpoints from the system of equations corresponding to $\pi_{V_o}(E(P))$ is unnecessarily expensive. A larger set, corresponding to a system with fewer unknowns (8 vs. 128), suffices for this problem.□

Theorem 19. Suppose $Y\le X$, the structure for P, and Y is a cover of V', where $V_o \subseteq V' \subseteq V = \bigcup_{V_i\in X} V_i$. Then $\pi_{V_o}(E(P)) \subseteq \pi_{V_o}(E(\pi_Y(P)))$.

Proof: If $E(P)=\varnothing$, then $\pi_{V_o}(E(P))=\varnothing$. If not, then $P=\pi_X(p)$ for any $p\in E(P)$, and

$$\begin{aligned}\pi_{V_o}(\mathrm{E}^V(\pi_X(\mathrm{p}))) &= \pi_{V_o}(\pi_{V'}(\mathrm{E}^V(\pi_X(\mathrm{p})))) \quad \text{[Lemma 2]}\\ &\subseteq \pi_{V_o}(\pi_{V'}(\mathrm{E}^V(\pi_Y(\mathrm{p})))) \quad \text{[Theorem 9]}\\ &\subseteq \pi_{V_o}(\mathrm{E}^{V'}(\pi_Y(\mathrm{p}))) \quad \text{[Lemma 8]}\\ &= \pi_{V_o}(\mathrm{E}^{V'}(\pi_Y(\pi_X(\mathrm{p})))) \quad \text{[Lemma 2]}\\ &= \pi_{V_o}(\mathrm{E}(\pi_Y(\mathrm{P}))). \quad \square\end{aligned}$$

Corollary 20. For P and X as above, Z a cover of $\mathrm{V}'' \supseteq \mathrm{V}_o$, and $\mathrm{Z} \leq \mathrm{Y} \leq \mathrm{X}$,

$$\pi_{V_o}(\mathrm{E}(\mathrm{P})) \subseteq \pi_{V_o}(\mathrm{E}(\pi_Y(\mathrm{P}))) \subseteq \pi_{V_o}(\mathrm{E}(\pi_Z(\mathrm{P}))). \square$$

If a_i is uniquely maximal under '>' relative to a set K of distributions, then it is uniquely maximal relative to any $\mathrm{K}' \subseteq \mathrm{K}$. This fact and Corollary 20 suggest the following strategy: Starting with the structure $\mathrm{W} = \{\{\mathrm{v}\} | \mathrm{v} \in \mathrm{V}_o\}$, the most refined structure that is a cover of some $\mathrm{V}' \supseteq \mathrm{V}_o$, repeatedly aggregate W until there is a unique maximal element under '>' relative to $\pi_{V_o}(\mathrm{E}(\pi_W(\mathrm{P})))$, or $\pi_{V_o}(\mathrm{E}(\pi_W(\mathrm{P})))$ happens to be a unit set, or W=X, whichever comes first. However, if structures are replaced by immediate aggregation, very little progress toward sufficient narrowing of utility intervals is likely to be made at each iteration. Also, there will usually not be a unique immediate aggregate. A reasonable alternative to a sequence of immediate aggregates is:

$$(\{\{\mathrm{v}\} | \mathrm{v} \in \mathrm{V}_o\},\ \{\mathrm{V}_i \cap \mathrm{V}_o | \mathrm{V}_i \in \mathrm{X}, \mathrm{V}_i \cap \mathrm{V}_o \neq \emptyset\},\ \{\mathrm{V}_i | \mathrm{V}_i \in \mathrm{X}, \mathrm{V}_i \cap \mathrm{V}_o \neq \emptyset\},\ \mathrm{X}).$$

Applying this method to the problem of Example IV.1, the utility intervals calculated from $\pi_{\{I,C\}}(\mathrm{E}(\pi_{\{\{I\},\{C\}\}}(\mathrm{P}))) = \mathrm{E}(\{\pi_{\{I\}}(\mathrm{p}_1), \pi_{\{C\}}(\mathrm{p}_2)\})$ are indecisive:

$$U(a_1) = [-2.5,\ 525]$$

$$U(a_2) = [0,\ 15]$$

$$U(a_3) = [0,\ 205].$$

$\{\mathrm{V}_i \cap \{\mathrm{I,C}\} | \mathrm{V}_i \in \mathrm{X}, \mathrm{V}_i \cap \{\mathrm{I,C}\} \neq \emptyset\} = \{\{\mathrm{I}\},\{\mathrm{C}\}\}$ also. However, $\{\mathrm{V}_i | \mathrm{V}_i \in \mathrm{X}, \mathrm{V}_i \cap \{\mathrm{I,C}\} \neq \emptyset\} = \{\{\mathrm{I,B}\},\{\mathrm{B,C}\}\}$, and action a_1 emerges as uniquely admissible with utility intervals calculated from $\pi_{\{I,C\}}(\mathrm{E}(\pi_{\{\{I,B\},\{B,C\}\}}(\mathrm{P}))) = \pi_{\{I,C\}}(\mathrm{E}(\{\mathrm{p}_1,\mathrm{p}_2\}))$:

$$U'(a_1) = [208,\ 525]$$

$$U'(a_2) = [0,\ 9]$$

$$U'(a_3) = [82,\ 205].$$

Therefore, a linear program involving only 8 unknowns (or, counting the previous steps, two programs, one with 4 unknowns and one with 8) is sufficient to identify a unique best action, vs. 128 unknowns for the linear program associated with $\pi_{\{I,C\}}(E(P))$.

An algorithm of Maier and Ullman [27] for finding paths between vertices (attributes) in acyclic hypergraphs may also be useful for such problems:

```
1. Z:=X.
2. Repeat in any order until neither has any effect on the current value of Z:
   a. If a variable $v \notin V_o$ appears in only one element of Z, remove v from that element.
   b. If Z contains elements $V_i$ and $V_j$ such that $V_i \subset V_j$, then $Z := Z - \{V_i\}$. □
```

The resulting structure Z is both a refinement of X and a cover of some $V' \supseteq V_o$. For problems with large numbers of possibly irrelevant attributes, it may be reasonable to substitute Z for X in the aggregation sequence strategy. In Example IV.1, $Z = \{\{I, B\}, \{B, C\}\}$ and the same sequence of utility intervals is generated.

B. Probability Estimation

A *contingency table* [4] may be modelled as a function $f\colon \mathrm{dom}(V) \rightarrow \mathbf{N}$; thus a collection of contingency tables $F = \{f_1, \ldots, f_m\}$ is formally very similar to a probabilistic database. The obvious relative-frequency-preserving mapping t_{fp} to a probabilistic database is not injective. However, the number of observations may be stored separately and the original tables recovered.

A problem for contingency table analysis is the occurrence of *sampling zeros*. These are values $f(t) = 0$ that are due to sample size limitations, and not to the impossibility of observing an entity with attributes t. Among other things, the presence of sampling zeros complicates certain common statistical procedures. Also, with small sample sizes, the relative frequencies are subject to extreme fluctuations when updated by further sampling.

A technique used to reduce both effects is to replace f with the distribution f′,

$$f'(t) = J(\pi_X(t_{fp}(f))) \times N,$$

where N is the sample size. (The values f′(t) may turn out not to be integers.) This method was used, notably, in the National Halothane Study [6], an examination of death rates following surgery under various anesthetics. Related methods have been employed by the U.S. Census Bureau [38].

Selection of the model X tends to be somewhat *ad hoc*. For tables with few attributes (*categories*), the model with all *two-factor effects* present, i.e.,

$$X = \{\{v, v'\} \mid v, v' \in V\},$$

is often used [4].

Recent experiments testing the behavior of reconstructability analysis provide strong evidence that replacing an initial relative frequency estimate with the join of its projections onto a suitable model also increases the accuracy of the estimate [16, 21, 35]. Suppose that a sample of N tuples $t \in \mathrm{dom}(V)$ is taken from a population for which p(t) is the actual probability of observing t. Let $p_N(t)$ denote the observed relative frequency of (entities with attribute) tuple t. It was discovered (in the course of massive experimentation involving models and domains of various sizes) that it is usually the case, for small N ($N < 5\times|\mathrm{dom}(V)|$, the usual rule-of-thumb minimum sample size for reliable application of many statistical techniques) and a model X from which p_N is approximately reconstructable, that

$$h(p_N, p) > h(J(\pi_X(p_N)), p),$$

where h denotes the sum of absolute deviations. (Directed divergence, in terms of which approximate reconstructability is measured, is not applicable to arbitrary pairs of distributions. [1])

The improvement in accuracy may be explained as follows [21, 35]. Suppose p is approximately reconstructable from $X = \{V_1, \ldots, V_m\}$. Since the ratios $N/|\mathrm{dom}(V_i)|$ will greatly exceed the ratio $N/|\mathrm{dom}(V)| = N/|\mathrm{dom}(V_1 \cup \cdots \cup V_m)|$, the marginals $\pi_{V_i}(p_N)$ will be much better approximations than is p_N itself. If the differences $h(\pi_{V_i}(p_N), \pi_{V_i}(p))$ are sufficiently small, then $h(p_N, p)$ will exceed $h(J(\pi_X(p_N), p)$.

Example IV.2: Distribution p of Example III.1 is a relative frequency distribution for 100 tuples randomly generated in accordance with the distribution:

Type	Plant	Defective	p′(t)
Chain	Lubbock	No	0.162
Chain	Lubbock	Yes	0.016
Chain	Waco	No	0.304
Chain	Waco	Yes	0.150
Sprocket	Lubbock	No	0.145
Sprocket	Lubbock	Yes	0.009
Sprocket	Waco	No	0.182
Sprocket	Waco	Yes	0.032

Projecting p onto $X = \{\{\mathrm{Type}, \mathrm{Plant}\}, \{\mathrm{Plant}, \mathrm{Defective}\}\}$ (Example II.2) and joining (Example III.4) results in an improved estimate of p′:

$$h(J(\pi_X(p)), p') = 0.152 < 0.258 = h(p, p'). \square$$

This method is essentially a *smoothing technique* [40], in which an initial estimate

$p_N \in P_V$ is replaced with one closer on some metric to an *ultrasmooth* estimate, usually the uniform distribution, $C(P_V)$.

The best known such method is *convex smoothing*, in which an estimate $\hat{p}$ on the Euclidean line between p_N and $C(P_V)$ is selected:

$$\hat{p} = \lambda p_N + (1-\lambda) C(P_V),$$

for some $0 \leq \lambda \leq 1$. The reconstruction technique, $\hat{p} = J(\pi_X(p_N))$, is analogous, and its smoothing effect may be explained in terms of the probabilistic algebra. Corresponding to $\lambda = 0$ and $\lambda = 1$ are the models $\{\emptyset\}$ and $\{V\}$, respectively, by Lemmas 6 and 7. For fixed p_N, the more refined (the closer in the lattice of models to $\{\emptyset\}$) X is, the closer $\hat{p}$ is to $C(P_V)$ as measured by directed divergence, from the identity [1]

$$d(p, C(P_V)) = H(C(P_V)) - H(p),$$

for any $p \in P_V$, and Theorem 9. Thus, a chain $(\{\emptyset\}, \cdots, \{V\})$ of immediate aggregates in the lattice of models of V corresponds to the interval $[0, 1]$ of values for λ. (The parameter λ may be selected to minimize risk, e.g. expected squared-error, given p_N [14]. The model X is selected by performing reconstructability analysis on p_N [35].)

V.Conclusions

The probabilistic algebra discussed in [9] is extended to include several new operators and is reexpressed more perspicuously. Homomorphisms between various probabilistic and relational subsystems are noted. A number of new, strictly probabilistic, results are derived and are shown to be of some practical use, e.g., for decision support.

An interesting project would be to explore connections between the real-valued probabilistic algebra discussed here, the algebra for interval-valued distributions sketched in [34], and the operators introduced by Barbara′ *et al*. [3] for their probabilistic-relational model.

Acknowledgment

This was written while I was on sabbatical at the University of Rochester. I thank Henry Kyburg and the rest of the University of Rochester computer science department for a very stimulating year. I am grateful to Martin Zwick for mentioning in 1982 that someone should do something about the obvious correspondence between results in (general) system theory and (relational) database theory. I am most grateful to Roger Cavallo for suggesting that the two of us do something about it and, more generally, for many years of stimulation, beginning in 1978.

VI. References

[1] J. Aczel and Z. Daroczy, *On Measures of Information and their Characterizations*, Academic Press, New York, 1975.

[2] W. R. Ashby, "Constraint analysis of many-dimensional relations," *General Systems Yearbook*, vol. 9, pp. 99-105, 1964.

[3] D. Barbara′, H. Garcia-Molina, and D. Porter, "The management of probabilistic data," *IEEE Trans. Knowl. Data Eng.*, to appear.

[4] Y. Bishop, S. Fienberg, and P. Holland, *Discrete Multivariate Analysis*, MIT Press, Cambridge, MA, 1975.

[5] D. Brown, "A note on approximations to discrete probability distributions," *Information and Control*, vol. 2, pp. 386-392, 1959.

[6] J. Bunker *et al.*, Eds., *The National Halothane Study*, National Institutes of Health, U.S. Government Printing Office, 1969.

[7] R. Cavallo and J. DeVoy, "Iterative and recursive algorithms for tree and partition search of the lattice of structure models," *Int. J. General Systems*, to appear.

[8] R. Cavallo and G. Klir, "Reconstructability analysis of multi-dimensional relations: a theoretical basis for computer-aided determination of acceptable systems models," *Int. J. of General Systems*, vol. 5, pp. 143-171, 1979.

[9] R. Cavallo and M. Pittarelli, "The theory of probabilistic databases," in *Proc. 13th Int. Conf. Very Large Databases*, Aug. 1987, pp. 71-81.

[10] P. Diaconis and S. Zabell, "Updating subjective probability," *J. Am. Stat. Assoc.*, vol. 77, pp. 822-830, 1982.

[11] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*, Academic Press, New York, 1980.

[12] R. Fagin, "Degrees of acyclicity for hypergraphs and relational database schemes," *JACM*, vol. 30, pp. 514-550, 1983.

[13] R. Fagin and M. Vardi, "The theory of data dependencies – a survey," in M. Anshel and W. Gewirtz, Eds., *Mathematics of Information Processing*, American Mathematical Society, Providence, RI, 1986.

[14] S. Fienberg and P. Holland, "Simultaneous estimation of multinomial cell probabilities," *J. Am. Stat. Assoc.*, vol. 68, pp. 683-691, 1973.

[15] B. R. Frieden, "Dice, entropy, and likelihood," *Proc. of the IEEE*, vol. 73, pp. 1764-1770, 1985.

[16] A. Hai and G. Klir, "An empirical investigation of reconstructability analysis: probabilistic systems," *Int. J. Man-Machine Studies*, vol. 22, pp. 163-192, 1985.

[17] M. Higashi, "A systems modelling methodology: probabilistic and possibilistic approaches," Ph.D. dissertation, SUNY-Binghamton, Binghamton, NY, 1984.

[18] E. T. Jaynes, "On the rationale of maximum-entropy methods," *Proc. of the IEEE*, vol. 70, pp. 939-952, 1982.

[19] E. T. Jaynes, "Prior information and ambiguity in inverse problems," in D. McLaughlin, Ed., *Inverse Problems, SIAM-AMS Proceedings*, vol. 14, pp. 151-166, 1984.

[20] G. Klir, *Architecture of Systems Problem Solving*, Plenum Press, New York, 1985.

[21] G. Klir, "Reconstruction principle of inductive reasoning," *Revue Int. de Systemique*, vol. 4, pp. 65-78, 1990.

[22] P. M. Lewis, "Approximating Probability Distributions to Reduce Storage Requirements," *Information and Control*, vol. 2, pp. 214-225, 1959.

[23] R. Loui, "Decisions with indeterminate probabilities," *Theory and Decision*, vol. 21, pp. 283-309, 1986.

[24] J. MacQueen and J. Marschak, "Partial knowledge, entropy, and estimation," *Proc. Nat. Acad. Sci.*, vol. 72, pp. 3819-3824, 1975.

[25] R. Madden and W. R. Ashby, "The identification of many-dimensional relations," *Int. J. Systems Science*, vol. 3, pp. 343-356, 1972.

[26] D. Maier, *The Theory of Relational Databases*, Computer Science Press, Rockville, MD, 1983.

[27] D. Maier and J. Ullman, "Connections in acyclic hypergraphs," *Proc. ACM Symp. on Principles of Database Systems*, pp. 34-39, 1982.

[28] M. Mariano, "Aspects of inconsistency in reconstructability analysis," Ph.D. Dissertation, SUNY-Binghamton, Binghamton, NY, 1987.

[29] K. McConway, "Marginalization and linear opinion pools," *J. Am. Stat. Assoc.*, vol. 76, pp. 410-414, 1981.

[30] K. Nambiar, "Some analytic tools for the design of relational database systems," in *Proc 6th Int. Conf. Very Large Data Bases*, 1980, pp. 417-428.

[31] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann, San Mateo, CA, 1988.

[32] F. Piepel, "Calculating centroids in constrained mixture experiments," *Technometrics*, vol. 25, pp. 279-283, 1983.

[33] M. Pittarelli, "Identification of discrete probability distributions from partial information," Ph.D. Dissertation, SUNY-Binghamton, Binghamton, NY, 1988.

[34] M. Pittarelli, "Probabilistic databases for decision analysis," *Int. J. Intelligent Systems*, vol. 5, pp. 209-236, 1990.

[35] M. Pittarelli, "A note on probability estimation using reconstructability analysis," *Int. J. General Systems*, vol. 18, pp. 11-21, 1990.

[36] M. Pittarelli, "Decisions with probabilities over finite product spaces," *IEEE Trans. Systems, Man, and Cybernetics*, vol. 21, pp. 1238-1242, 1991.

[37] T. Seidenfeld, "Entropy and uncertainty," *Philosophy of Science*, vol. 53, pp. 467-491, 1986.

[38] F. Stephan *et al.*, "The sampling procedure of the 1940 population census," *J. Am. Stat. Assoc.*, vol. 35, pp. 615-630, 1940.

[39] Y. Tian, "Probabilistic databases over acyclic schemes," Master's Thesis, SUNY Institute of Technology, Utica, NY, 1988.

[40] D. Titterington, "Common structure of smoothing techniques in statistics," *Int. Stat. Review*, vol. 53, pp. 141-170, 1985.

Michael Pittarelli
Computer Science Department
SUNY Institute of Technology
Utica, NY 13504-3050
mike@sunyit.edu