# A Probabilistic Network Versus Decision Rules

**S.K.M. Wong** and **C.J. Butz**

*Department of Computer Science*
*University of Regina*
*Regina, Saskatchewan, Canada S4S 0A2*
*E-Mail: {wong,butz}@cs.uregina.ca*

**Abstract** In this paper, we suggest the use of probabilistic networks in inductive learning as a plausible alternative to deriving explicit decision rules. Whereas a decision rule can only be applied if its antecedent is given, a probabilistic request can be issued involving any set of attributes and attribute values in the sample observations. We will also demonstrate that probabilistic networks can be represented in a generalized relational data model. This representation allows us to take full advantage of the query optimization techniques developed for relational databases in processing probabilistic requests.

## 1 Introduction

One of the key tasks in designing an expert system of a particular application is to specify appropriate inference (decision) rules. There are two popular approaches for the construction of these rules. The domain experts may have to directly supply the inference rules. This approach is usually very tedious and time consuming. Another method is to learn the inference rules from observed data. This latter approach is an automated process. However, we need to develop an efficient and effective learning algorithm in order to extract or mine the *rules* or *patterns* from a large set of sample observations.

The subject of inductive learning has been extensively studied [1, 5, 6, 7, 8, 13]. Many techniques have been developed and applied successfully to many areas of machine learning. Most of these learning methods derive explicit decision rules. In some cases, due to the insufficient information each derived inference rule is associated with a degree of uncertainty which is usually represented by a numeric value. The measure of uncertainty sometimes is chosen without a sound theoretical foundation, although it may work quite well in some applications. Another drawback in using explicit decision rules is that a rule cannot be used unless all of its antecedent conditions are known. Clearly, it would be much more desirable if such a limitation could be eliminated; one could then make a decision based on whatever information is available on hand.

The main objective of this paper is to present probabilistic networks as an alternative approach to using decision rules as an inference mechanism. We argue that the above drawbacks we mentioned are absent in the probabilistic approach. Moreover, effective methods can be developed to learn the dependency structure of a probabilistic model from observed data. This will greatly facilitate the representation of probabilistic knowledge and enhance the performance of the inference process.

This paper is organized as follows. In Section 2, we introduce a generalized relational data model to represent probabilistic networks. In Section 3, we first describe a learning algorithm for extracting or mining the data dependencies (independencies) satisfied by a probabilistic network from sample observations. We also describe in this section a method for constructing the dependency structure of a network if the conditional independencies are known à priori. We assume that the required marginal distributions in the former can be reliably determined from observed data, whereas in the lat-

| $A_1$ | $\ldots$ | $A_m$ | $f_{\phi_R}$ |
|---|---|---|---|
| $t_1[A_1]$ | $\ldots$ | $t_1[A_m]$ | $t_1[f_{\phi_R}] = \phi_R(t_1[R])$ |
| $t_2[A_1]$ | $\ldots$ | $t_2[A_m]$ | $t_2[f_{\phi_R}] = \phi_R(t_2[R])$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $t_s[A_1]$ | $\ldots$ | $t_s[A_m]$ | $t_s[f_{\phi_R}] = \phi_R(t_s[R])$ |

Figure 1: A joint distribution $\phi_R$ expressed as a *relation* $\Phi_R$.

ter they are directly supplied by the domain experts. The resulting dependency structure together with the estimated marginals (potentials) completely define the probabilistic network of interest. In Section 5, we demonstrate the advantages of probabilistic networks over the decision rule approach. The conclusion is presented in Section 6.

## 2 A Generalized Relational Data Model

We have shown that probabilistic networks [2] can be viewed as a generalization of the standard relational database model [9, 11]. This data model provides a unified view of probabilistic and traditional relational [4] concepts.

Let $R = \{A_1, \ldots, A_m\}$ denote a finite set of variables. A joint probability distribution (jpd) is a normalized nonnegative real-valued function $\phi_R$ on $R$. We can succinctly express a jpd $\phi_R$ as a generalized relation $\Phi_R$ in our model. The relation $\Phi_R$ is defined by the set of attributes $\{A_1, \ldots, A_m, f_{\phi_R}\}$. For convenience, we will say $\Phi_R$ is a relation over $R = \{A_1, \ldots, A_m\}$. It is understood that the attribute $f_{\phi_R}$ is implicitly used for defining the relation $\Phi_R$. Each row in $\Phi_R$ is defined by a tuple $t_i$ in a standard relation $r[R]$ as shown in Figure 1.

The operators on probabilistic distributions can be defined in terms of generalized relational operators.

If $\Phi_R$ is an relation and $X \subseteq R$, then the marginalization of $\Phi_R$ onto $X$ is the *marginal* relation, denoted $\Phi_R^{\downarrow X}$, with attributes $X \cup \{f_{\phi_R^{\downarrow X}}\}$, defined by:

$$\Phi_R^{\downarrow X} = \{t[X \cup \{f_{\phi_R^{\downarrow X}}\}] \ \mid \ t[X] \in \Phi_R[X], \text{ and}$$
$$t[f_{\phi_R^{\downarrow X}}] = \phi_R^{\downarrow X}(t[X]) = \sum_{t' \in \Phi_R, t'[X]=t[X]} \phi_R(t'[R])\}.$$

Let $\phi_X$ and $\phi_Y$ be two distributions over $X$ and $Y$, respectively. We can express the product $\phi_X \cdot \phi_Y$ as the *product join* $\Phi_X \times \Phi_Y$ of the corresponding relations $\Phi_X$ and $\Phi_Y$. That is, $\Phi_X \times \Phi_Y$ is a relation on the set of attributes $XY \cup \{f_{\phi_X \cdot \phi_Y}\}$ defined as follows:

$$\Phi_X \times \Phi_Y = \{t[XY \cup \{f_{\phi_X \cdot \phi_Y}\}] \mid t[XY] \in (\Phi_X \bowtie \Phi_Y),$$
$$\text{and } t[f_{\phi_X \cdot \phi_Y}] = \phi_X(t[X]) \cdot \phi_Y(t[Y])\},$$

where $XY$ denotes the union $X \cup Y$.

Let $\Phi_R$ be an relation. The inverse of $\Phi_R$ is the *inverse* relation, denoted $(\Phi_R)^{-1}$, with attributes $R \cup \{f_{(\phi_R)^{-1}}\}$, defined by:

$$(\Phi_R)^{-1} = \{t[R \cup \{f_{(\phi_R)^{-1}}\}] \ \mid \ t[R] \in \Phi_R[R] \text{ and}$$
$$t[f_{(\phi_R)^{-1}}] = 1/t[f_{\phi_R}] \text{ if } t[f_{\phi_R}] > 0, \text{ and}$$
$$t[f_{(\phi_R)^{-1}}] = t[f_{\phi_R}] \text{ otherwise } \}.$$

Generalized relational data dependencies can now be introduced. The fundamental notion of generalized multivalued dependency is introduced first. This generalized data dependency is equivalent to probabilistic conditional independence. Next the notion of a generalized join dependency is defined. A probabilistic network can be expressed as a generalized join dependency in our data model.

Let $X, Y, Z \subseteq R$ such that $Y \cap Z \subseteq X$, and $\Phi_R$ a relation over $R$. We say that the *generalized multivalued dependency* (GMVD), written $X \Rightarrow\Rightarrow Y \mid Z$, is satisfied by the relation $\Phi_R$

if and only if the marginal relation $\Phi_R^{\downarrow XYZ}$ of $\Phi_R$ can be factorized as follows:

$$\begin{aligned}\Phi_R^{\downarrow XYZ} &= \Phi_R^{\downarrow XY} \times \Phi_R^{\downarrow XZ} \times (\Phi_R^{\downarrow X})^{-1} \\ &\equiv \Phi_R^{\downarrow XY} \otimes \Phi_R^{\downarrow XZ}.\end{aligned}$$

We refer to the binary operation $\otimes$ defined above as the *generalized join.* (Note that it can be shown that $X \Rightarrow\Rightarrow Y \mid Z$ if and only if $X \Rightarrow\Rightarrow (Y - X) \mid (Z - X)$.)

We say a *generalized acyclic join dependency* (GAJD) $\otimes\mathcal{R} = \{R_1, R_2, \ldots, R_n\}$ is satisfied by a relation $\Phi_R$, if $\Phi_R$ can be written as:

$$\Phi_R = (\ldots((\Phi_R^{\downarrow R_1} \otimes \Phi_R^{\downarrow R_2}) \otimes \Phi_R^{\downarrow R_3}) \ldots \otimes \Phi_R^{\downarrow R_n}),$$

where the sequence $R_1, R_2, \ldots, R_n$ is a hypertree (an acyclic hypergraph [4, 11]) construction ordering for $\mathcal{R}$.

Let $\Phi_R$ be a joint probability distribution and $\mathcal{R}$ an acyclic hypergraph. (In this paper we assume that $R = \cup R_i$, $R_i \in \mathcal{R}$.) A *Markov network* [2] is a pair $(\Phi_R, \mathcal{R})$ such that the GAJD, $\otimes\mathcal{R}$, is satisfied by $\Phi_R$. We call the acyclic hypergraph $\mathcal{R}$ the *dependency structure* of the Markov network $(\Phi_R, \mathcal{R})$, and $\Phi_R$ a *Markov distribution.*

The important point is that a jpd $\phi_R$ (see Figure 1) can be economically represented as the probabilistic network $(\Phi_R, \mathcal{R})$. Advantages of representing probabilistic networks in our generalized relational data model will be discussed in Section 4.

# 3 Constructing the Dependency Structure

In this section, we present a learning algorithm for mining the dependency structure $\mathcal{R}$ of a probability distribution $\phi_R$ (sample observations). (For more details, see [12].) We first characterize a Markov distribution by an entropy function.

Given a distribution $\phi_X$ on $X \subseteq R$, we can define a function as follows:

$$\begin{aligned}\mathcal{H}(\phi_X) &= -\sum_{t_X} \phi_X(t_X) \log \phi_X(t_X) \\ &= -\sum_{x} \phi_X(x) \log \phi_X(x), \quad (1)\end{aligned}$$

where $t$ is a tuple (configuration) of $X$ and $x = t_X = t[X]$ is a X-value, and $\mathcal{H}$ is the Shannon entropy.

From a Markov distribution,

$$\phi_R = \frac{\prod_{i=1}^n \phi_{R_i}}{\prod_{i=2}^n \phi_{R_{i^*} \cap R_i}} = \frac{\prod_{i=1}^n \phi_{R_i}}{\prod_{i=2}^n \phi_{R_i'}}, \qquad (2)$$

where $R_{i^*}$ is the branch [11] of $R_i$, and $R' = R_{i^*} \cap R_i$, $i = 2, 3, \ldots, n$, we obtain:

$$\mathcal{H}(\phi_R) = \sum_{i=1}^n \mathcal{H}(\phi_{R_i}) - \sum_{i=2}^n \mathcal{H}(\phi_{R_{i^*} \cap R_i}). \quad (3)$$

**Theorem 1** [12] Let $\mathcal{R} = \{R_1, R_2, \ldots, R_n\}$ be hypertree and $R = R_1 R_2 \ldots R_n$. Let the sequence $R_1, R_2, \ldots, R_n$ be a hypertree construction ordering for $\mathcal{R}$ such that $(R_1 R_2 \ldots R_{i-1}) \cap R_i = R_{i^*} \cap R_i$ for $1 \leq i^* \leq n-1$ and $2 \leq i \leq n$. A joint probability distribution $\phi_R$ factorized on $\mathcal{R}$ is a Markov distribution, if and only if

$$\mathcal{H}(\phi_R) = \sum_{i=1}^n \mathcal{H}(\phi_{R_i}) - \sum_{i=2}^n \mathcal{H}(\phi_{R_{i^*} \cap R_i}). \quad (4)$$

This theorem indicates that we can characterize a Markov distribution by an entropy function. We now outline how we learn the dependency structure of a Markov distribution.

Our method for learning the dependency structure of a Markov distribution is based on minimizing the Kullback-Leibler $I(\phi, \phi')$ cross-entropy [3], a measure of closeness between a true distribution $\phi$ and an approximate distribution $\phi'$. The cross entropy is defined by:

$$I(\phi, \phi') = \sum_x \phi(x) \log \frac{\phi(x)}{\phi'(x)},$$

where $x$ is a configuration (tuple) of the set of attributes $R = \{A_1, \ldots, A_m\}$. With a fixed $\phi$, we can choose, from the set of all possible Markov distributions $\{\phi'\}$, the distribution $\phi_0$ that minimizes $I(\phi, \phi')$. For a Markov distribution, the cross-entropy can be written as $I(\phi, \phi') = \mathcal{H}(\phi') - \mathcal{H}(\phi)$, where $\mathcal{H}$ is the Shannon entropy as defined by equation (1). Thus, for a given $\phi$, minimizing $I(\phi, \phi')$ is equivalent to minimizing the entropy $\mathcal{H}(\phi')$. That is,

$$\min_{\phi'' \in \{\phi'\}}(I(\phi, \phi'')) = \min_{\phi'' \in \{\phi'\}}(\mathcal{H}(\phi'')). \quad (5)$$

An approximate method [12] for computing $\phi_0$ is outlined as follows. Initially, we may assume that all the attributes are probabilistically independent, i.e., there exists no edge between any two nodes (attributes) in the undirected graph representing the Markov distribution. Then an edge is added to the graph subject to the restriction that the resultant hypergraph must be a hypertree. The undirected graph of the Markov distribution with minimum entropy is being selected as the graph for further addition of other edges. This process is repeated until a predetermined *threshold*, which defines the rate of decrease of entropy between successive distributions, is reached. The output of this iterative procedure is the dependency structure of the *minimum* Markov distribution $\phi_0$. That is, $\phi_0$ satisfies both equations (4) and (5). From the output hypertree, we can infer the probabilistic conditional independencies which are satisfied by $\phi_0$.

A second approach to constructing the dependency structure $\mathcal{R}$ of a probabilistic network, when sample data is not available or reliable, is to have the domain expert supply a set of probabilistic conditional independencies. We briefly outline an automated process for the construction of a Markov network [10] in these situations.

We use a complete set of inference axioms for GMVDs defined on a *fixed* set of attributes to remove redundant and detect inconsistent (conflicting) independencies. The remaining set of GMVDs, called a *conflict-free full minimum cover* [10], is used to systematically construct the dependency structure of the desired Markov network. (Note that the resulting acyclic hypergraph is in fact a *perfect-map*. That is, every GMVD logically implied by the input set can be inferred from the acyclic hypergraph, and every GMVD inferred from the acyclic hypergraph is logically implied by the input set.)

We write $X \Rightarrow\Rightarrow Y_1 \mid Y_2 \mid \ldots \mid Y_m$, to summarize the set of GMVDs $\{X \Rightarrow\Rightarrow Y_1, X \Rightarrow\Rightarrow Y_2, \ldots, X \Rightarrow\Rightarrow Y_m\}$. The *dependency basis* [10] of $X$, written $Dep(X)$, is the set

$$Dep(X) = \{W_1, W_2, \ldots, W_n\}.$$

The usefulness of $Dep(X)$ is that if the GMVD $X \Rightarrow\Rightarrow Y$ is logically implied by the input set of GMVDs, then $Y$ is the union of sets in $Dep(X)$.

Given a conflict-free full minimum cover $G$ over a set $R$ of attributes, a *construction algorithm* factorizes $R$ into two sets of attributes on the basis of a known conditional independency in $G$. That is, given $XYZ = R$, the set of attributes $R$ is replaced by $R \cap XY$ and $R \cap XZ$, where the GMVD $X \Rightarrow\Rightarrow Y \mid Z$ is in $G$. Each of these new subsets may be further factorized on the basis of another known GMVD in $G$.

Given a conflict-free full minimum cover $G$ and a *p-ordering sequence* [10] ($X_1$, $X_2$, ..., $X_p$) of the elements in $\mathbf{X} = \{X \mid X \Rightarrow\Rightarrow Y|Z$ is in $G\}$, Algorithm 2 constructs an acyclic hypergraph $\mathcal{R}$ representing the dependency structure of the input GMVDs as follows:

*Algorithm 2*

Input: a set of GMVDs $G$ and a p-ordering sequence of $\mathbf{X}$

$\mathcal{R}^0 := \{R\}$;

for $i := 1$ to $p$
    $\mathcal{R}^{i-1} := \mathcal{R}^{i-1} - \{R_j\}$, where $X_i \subseteq R_j$;
    $\mathcal{R}^i := \mathcal{R}^{i-1} \cup \{X_i(R_j \cap W) \mid W \in Dep(X_i)$ and $R_j \cap W \neq \emptyset\}$;
Output: $(\mathcal{R}^p)$

**Example 1** Let $R = \{A_1, \ldots, A_9\}$ and let $(\{A_1\}, \{A_2\}, \{A_3\}, \{A_1, A_2\}, \{A_1, A_3\}, \{A_2, A_3\})$ be a p-ordering sequence of $\mathbf{X}$ in the conflict-free full minimum cover of the following GMVDs: $\{A_1\} \Rightarrow\Rightarrow \{A_7\} \mid \{A_2, \ldots, A_6, A_8, A_9\}$, $\{A_2\} \Rightarrow\Rightarrow \{A_8\} \mid \{A_1, A_3, \ldots, A_7, A_9\}$, $\{A_3\} \Rightarrow\Rightarrow \{A_9\} \mid \{A_1, A_2, A_4, \ldots, A_8\}$, $\{A_1, A_2\} \Rightarrow\Rightarrow \{A_4\} \mid \{A_7\} \mid \{A_8\} \mid \{A_3, A_5, A_6, A_9\}$, $\{A_1, A_3\} \Rightarrow\Rightarrow \{A_5\} \mid \{A_7\} \mid \{A_9\} \mid \{A_2, A_4, A_6, A_8\}$, $\{A_2, A_3\} \Rightarrow\Rightarrow \{A_6\} \mid \{A_8\} \mid \{A_9\} \mid \{A_1, A_4, A_5, A_7\}$.

The output acyclic hypergraph $\mathcal{R}$ generated by Algorithm 2 is: $\mathcal{R}^6 = \mathcal{R} = \{ \{A_1, A_7\}, \{A_2, A_8\}, \{A_3, A_9\}, \{A_1, A_2, A_4\}, \{A_1, A_3, A_5\}, \{A_2, A_3, A_6\}, \{A_1, A_2, A_3\}\}$.

Once the dependency structure $\mathcal{R}$ is constructed, the required marginal distributions corresponding to the hyperedges $R_i \in \mathcal{R}$ need to be elicited from the domain experts.

We have now constructed a probabilistic network using either sample observations or supplied probabilistic conditional independencies. We next demonstrate the usefulness of probabilistic networks for reasoning.

# 4 Advantages of Our Representation

There are a number of advantages in representing the probabilistic knowledge as a generalized relational model. First, we can answer any request (query) posed on a probabilistic network. Such a query may involve any subset of attributes taking on any values in their respective domains. In contrast, decision rules can only be applied if all its antecedents are given. In our method, the numeric measure of uncertainty is automatically defined as the conditional probability which is a fundamental notion in the theory of probability. Thus, any ad hoc choice of uncertainty measure associated with decision rules can be avoided in our approach. Moreover, our generalized relational data model can take advantage of the query optimization techniques developed for conventional databases in the inference process.

Suppose that sample observations are represented according to the attributes $\{A_1, \ldots, A_n, D_1\}$, where $D_1$ denotes the expert's classification according to some concept.

In the decision rule approach, one would derive explicit decision rules such as $(X) \longrightarrow^d (Y)$, where $X$ is the antecedent, $Y$ is the set of decision attributes, and $d$ is the degree of uncertainty associated with this decision rule. For example, we may have a decision rule $(A_1 := 0, A_3 := 0) \longrightarrow^{0.78} (D_1 := \text{“}+\text{”})$. These rules are used to *classify* new observations according the concept of interest. However, if the values of the new observation $A_e = \epsilon, \ldots, A_g = \gamma$ are *not* equal to the values in the antecedent of any derived decision rule, then the new observation can not be classified. On the other hand, we can always compute the conditional probability $p(D_1 := \text{“}+\text{”} \mid A_e := \epsilon, \ldots, A_g := \gamma)$.

In our generalized relational data model, a classification of a new observation can be processed efficiently in the same fashion as a standard query posed to a relational database. In general, given that a new observation takes on values $A_e := \epsilon, \ldots, A_g := \gamma$, and one wants to determine the probability of $A_a, \ldots, A_d$, i.e., the conditional probability $p(A_a, \ldots, A_d \mid A_e := \epsilon, \ldots, A_g := \gamma)$, we transform this input information to a relational query. The resultant relation is the *answer* to the classification of the new observation.

Thus, any new observation can be classified by submitting SQL queries to the standard database management system.

# 5 Conclusion

In this paper, we have suggested the use of probabilistic networks as a plausible alternative to deriving explicit decision rules in inductive learning. Whereas a decision rule can only be applied if its antecedent is given, a probabilistic request can be issued involving any set of attributes and attribute values in the sample observations. We have also shown that probabilistic networks can be represented in a generalized relational data model. This representation allows us to take full advantage of the established query optimization techniques developed for relational databases in processing probabilistic requests.

# References


[1] J.W. Grzymala-Busse. *LERS - A system for learning from examples based on rough sets*, chapter Intelligent Decision Support Handbook of Applications and Advances of the Rough Sets Theory. Kluwer Academic Press, Norwell, MA, 1992.

[2] P. Hajek, T. Havranek, and R. Jirousek. *Uncertain Information Processing in Expert Systems.* CRC Press, 1992.

[3] S. Kullback and R. Leibler. Information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86, 1951.

[4] D. Maier. *The Theory of Relational Databases.* Principles of Computer Science. Computer Science Press, Rockville, Maryland, 1983.

[5] R.S. Michalski, J.G. Carbonall, and T.M. Mitchell, editors. *Machine Learning: the Artificial Intelligence Approach.* Tioga Press, Palo Alto, 1983.

[6] Z. Pawlak. *Rough Sets: Theoretical Aspects of Reasoning About Data.* Kluwer, Boston, 1991.

[7] J.R. Quinlan. C4.5. *Programs for Machine Learning*, 1993.

[8] P. Utgoff. Id5: An incremental id3. In *Fifth International Conference on Machine Learning*, pages 107–120. Morgan Kaufmann Publishers, 1988.

[9] S.K.M. Wong. An extended relational data model for probabilistic reasoning. *Journal of Intelligent Information Systems*, 9:181–202, 1997.

[10] S.K.M. Wong and C.J. Butz. A method for constructing the dependency structure of a probabilistic network. In *International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems*, 1998.

[11] S.K.M. Wong, C.J. Butz, and Y. Xiang. A method for implementing a probabilistic model as a relational database. In *Eleventh Conference on Uncertainty in Artificial Intelligence*, pages 556–564. Morgan Kaufmann Publishers, 1995.

[12] S.K.M. Wong and Y. Xiang. Construction of a markov network from data for probabilistic inference. In *Third International Workshop on Rough Sets and Soft Computing*, pages 562–569, 1994.

[13] S.K.M. Wong, W. Ziarko, and R.L. Ye. Comparison of rough-set and statistical methods in inductive learning. *International Journal of Man-Machine Studies*, 24:53–72, 1986.