Towards Understanding of Pseudo-independent Domains

Y. Xiang
Department of Computer Science, University of Regina
Regina, Saskatchewan, Canada S4S 0A2 yxiang@cs.uregina.ca

Abstract

A *pseudo-independent* (PI) domain is a problem domain where a proper subset of a set of collectively dependent variables displays marginal independence. Common algorithms for learning belief networks cannot learn a faithful representation of the domain dependence when the data is obtained from a PI domain. Since we usually have no *a priori* knowledge whether a domain of interest is PI or not, we may learn an incorrect belief network, suffer from the consequence, and be not aware of it. Design of more reliable learning algorithms depends highly on a better understanding of these domains. This paper reports our progress towards such a goal.

We characterize the whole spectrum of discrete PI domains with formal definitions. This forms a basis for studying them. We present our progress on parameterization of PI domains which eventually will lead to a better understanding of the mechanism that forms PI domains. Whether PI domains exist in practice is a common concern. We show that parity and modulus addition problems are special PI domains, which provides positive evidence. Application of our results to learning is discussed.

Keywords: machine learning, modeling, belief networks.

1 INTRODUCTION

A belief network (BN) [5], a Bayesian network or a decomposable Markov network (DMN) [8], consists of a graphical structure whose nodes are labeled by domain variables and a joint probability distribution (jpd) that is factorized according to the structure. The structure encodes probabilistic dependence among domain variables and the jpd quantifies the strength of the dependence. The structure of a Bayesian network is a directed acyclic graph and that of a DMN is a chordal graph.[1] BNs are becoming widely applied to AI tasks where representing and reasoning with uncertain knowledge are essential. As an alternative and supplement to encoding uncertain knowledge from domain experts, learning BNs from data is studied by many [1, 2, 8]. The learning task is to take as input a dataset observed from a problem domain,

[1]For the purpose of this paper, we shall use DMNs.

and to infer one or more sparse BN(s) that can later be used to answer probabilistic queries about the domain.

It is important that a learned BN is both a *faithful* and a *compact* representation of the probabilistic dependence among domain variables. If it is not faithful, answers to queries obtained from it will be *incorrect*. If it is not compact, query processing will be *inefficient*. Since what we have is the data and the domain dependence is *unknown* to us, how do we know that the learned BN is faithful and compact? Pearl [5] showed that the structure of a faithful BN is an I-map (defined below), and the structure of a both faithful and compact BN is a minimal I-map (defined below). How can we tell if a learned BN is a minimal I-map? We cannot in practice. Instead, we depend on the *reliability* of our learning algorithm. We must ensure that the algorithm used will learn an approximate minimal I-map when the data is obtained from any one of a wide range of problem domains. Otherwise, we may learn an *unfaithful* BN, receive *incorrect* answers to queries from it, and suffer from the decision errors.

Formally, the concepts involved can be described as follows: For disjoint subsets A, B and C of nodes in a graph G, we use $< A|C|B >$ to denote that nodes in C intercept all paths between A and B. A graph G is an *I-map* of a problem domain with a set N of variables if there is an one-to-one correspondence between nodes of G and variables in N such that for all disjoint subsets A, B and C of N, $< A|C|B >$ implies that A and B are conditionally independent given C. G is a *minimal* I-map of a problem domain, if no link in G can be removed such that the resultant graph is still an I-map.

Common algorithms for learning BNs rely on identifying local dependence among variables. For example, an algorithm may start with a graph without links. It compares all graphs with a single link (local dependence) and chooses the one that best fits the data. It then compares all graphs with an additional link, and continues the process until some stopping condition is met. It has been shown [7, 8] that for a class of problem domains, the BNs learned by these algorithms are *not* approximate I-maps. That is, these algorithms are *unreliable*.

What feature this class of difficult domains? In these domains, a proper subset X of a set Y of collectively dependent variables displays marginal independence. They are thus termed *pseudo-independent* (PI) domains [8]. Since members of Y are collectively dependent, a faithful BN has links between each pair of members. On the other hand, since members of X are marginally independent, the above algorithms cannot find a link to add among them, which results in a non-I-map.

How can we improve the reliability of learning algorithms? Since they fail in PI domains, we need to improve our understanding about PI domains, which will lead to the design of more reliable algorithms. This paper presents our progress towards such understanding. We characterize the whole spectrum of discrete PI domains (Sections 2 and 3) with formal definitions. We present a parameterization of full PI domains (defined below) as the initial step towards parameterization of general PI domains (Section 4), which will lead to a better understanding of the mechanism that forms PI domains. A commonly asked question is "do PI domains exist in practice?". We show in Section 5 that the parity problems and modulus addition problems are both PI domains. We discuss in Section 6 the application of our results to learning.

2 NOTIONS OF PROBABILISTIC DEPENDENCE

The concept of conditional independence is well known. In this section, we distinguish several notions of probabilistic dependence (particularly the collective and general

dependence) that are less commonly used but are essential to the understanding of PI domains.

Let N be a set of discrete variables in a problem domain. Each variable is associated with a set of possible values. We shall denote the values by consecutive integers 0, 1, 2, ... A *configuration* or a *tuple* of $N' \subseteq N$ is an assignment of values to every variable in N', e.g., $(X_1 = 0, X_2 = 1, \ldots)$ which we shall denote by $(x_{1,0}, x_{2,1}, \ldots)$. A *probabilistic domain model* (PDM) over N determines the probability of every tuple of N' for each $N' \subseteq N$. Without confusion, we shall use *problem domain* and PDM interchangeably.

For three disjoint sets A, B and C of variables, A and B are *conditionally independent* given C if $P(A|B,C) = P(A|C)$ *whenever* $P(B,C) > 0$. When $C = \phi$, A and B are *marginally independent*. If each variable X in a subset A is marginally independent of $A \setminus \{X\}$, then $P(A) = \prod_{X \in A} P(X)$. We shall say that variables in A are marginally independent.

A pair of variables X and Y are *pairwise dependent* if $P(X|Y) \neq P(X)$. Pairwise dependence is the opposite of marginal independence between the pair. A set N of variables are *collectively dependent* if for each proper subset $A \subset N$, there exists no proper subset $C \subset N \setminus A$ such that $P(A|N \setminus A) = P(A|C)$. Collective dependence does not eliminate possible marginal independence *within* a proper subset, as will be seen in Section 3.

A set N of variables are *generally dependent* if for any proper subset A, $P(A|N \setminus A) \neq P(A)$. Like collective dependence, general dependence does not eliminate possible marginal independence within a proper subset. Furthermore, collective dependence is not required. Namely, for some proper subset A, there can be a proper subset $C \subset N \setminus A$ such that $P(A|N \setminus A) = P(A|C)$.

General dependence is weaker dependence than collective dependence. Each may coexist with either conditional independence or marginal independence within proper subsets. General dependence is the opposite of marginal independence between a proper subset and the rest of domain variables.

3 FORMALIZATION OF PI DOMAINS

Without confusion, we shall use *PI domain* and *PI model* interchangeably. PI models (domains) can be classified into three types. The most restrictive type is *full* PI models.

Definition 1 (Full PI model) *A PDM over a set N ($|N| \geq 3$) of variables is a* `full` *PI model if the following two conditions hold: (S1) For each $X \in N$, variables in $N \setminus \{X\}$ are marginally independent. (S2) Variables in N are collectively dependent.*

Table 1 shows the jpd of a binary full PI model, where $X = (X_1, X_2, X_3, X_4)$ and marginals are $P(x_{1,0}) = 0.7$, $P(x_{2,0}) = 0.6$, $P(x_{3,0}) = 0.35$, $P(x_{4,0}) = 0.45$. Any subset of three variables are marginally independent, e.g., $P(x_{1,1}, x_{2,0}, x_{3,1}) = P(x_{1,1})\ P(x_{2,0})\ P(x_{3,1}) = 0.117$. The four variables are collectively dependent, e.g., $P(x_{1,1}|x_{2,0}, x_{3,1}, x_{4,0}) = 0.257$ and $P(x_{1,1}|x_{2,0}, x_{3,1}) = P(x_{1,1}|x_{2,0}, x_{4,0}) = P(x_{1,1}|x_{3,0}, x_{4,0}) = 0.3$.

In a full PI model, every proper subset of N displays marginal independence. This is relaxed in the *partial* PI models.

Definition 2 (Partial PI model) *A PDM over a set N ($|N| \geq 3$) of variables is a* `partial` *PI model if the following two conditions hold: (S1') There exists a*

X	$P(.)$	X	$P(.)$	X	$P(.)$	X	$P(.)$
$(0,0,0,0)$	0.0586	$(0,1,0,0)$	0.0517	$(1,0,0,0)$	0.0359	$(1,1,0,0)$	0.0113
$(0,0,0,1)$	0.0884	$(0,1,0,1)$	0.0463	$(1,0,0,1)$	0.0271	$(1,1,0,1)$	0.0307
$(0,0,1,0)$	0.1304	$(0,1,1,0)$	0.0743	$(1,0,1,0)$	0.0451	$(1,1,1,0)$	0.0427
$(0,0,1,1)$	0.1426	$(0,1,1,1)$	0.1077	$(1,0,1,1)$	0.0719	$(1,1,1,1)$	0.0353

Table 1: A full PI model.

partition $\{N_1, \ldots, N_k\}$ *(*$k \geq 2$*) of* N *such that variables in each subset* N_i *are generally dependent, and for each* $X \in N_i$ *and each* $Y \in N_j$ *(*$i \neq j$*),* X *and* Y *are marginally independent. (S2) Variables in* N *are collectively dependent.*

Table 2 shows the jpd of a partial PI model over two trinary variables and one binary variable, where $X = (X_1, X_2, X_3)$ and the marginals are $P(x_{1,0}) = 0.3,\ P(x_{1,1}) = 0.2,\ P(x_{1,2}) = 0.5, P(x_{2,0}) = 0.3,\ P(x_{2,1}) = 0.4,\ P(x_{2,2}) = 0.3, P(x_{3,0}) = 0.4,$ $P(x_{3,1}) = 0.6$. The partition is $\{\{X_1\}, \{X_2, X_3\}\}$. X_1 is marginally independent of each variable in the other subset, e.g., $P(x_{1,1}, x_{2,0}) = P(x_{1,1})\ P(x_{2,0}) = 0.06$. However the variables in the other subset are pairwise dependent, e.g., $P(x_{2,0}, x_{3,1}) = 0.1 \neq P(x_{2,0})\ P(x_{3,1}) = 0.18$. The three variables are collectively dependent, e.g., $P(x_{1,1}|x_{2,0}, x_{3,1}) = 0.1$ and $P(x_{1,1}|x_{2,0}) = P(x_{1,1}|x_{3,1}) = 0.2$. Similarly, $P(x_{2,1}|x_{1,0}, x_{3,1}) = 0.61,\ P(x_{2,1}|x_{1,0}) = 0.4,\ P(x_{2,1}|x_{3,1}) = 0.5$.

X	$P(.)$	X	$P(.)$	X	$P(.)$	X	$P(.)$	X	$P(.)$	X	$P(.)$
$(0,0,0)$	0.05	$(0,1,1)$	0.11	$(1,0,0)$	0.05	$(1,1,1)$	0.08	$(2,0,0)$	0.10	$(2,1,1)$	0.11
$(0,0,1)$	0.04	$(0,2,0)$	0.06	$(1,0,1)$	0.01	$(1,2,0)$	0.03	$(2,0,1)$	0.05	$(2,2,0)$	0.01
$(0,1,0)$	0.01	$(0,2,1)$	0.03	$(1,1,0)$	0	$(1,2,1)$	0.03	$(2,1,0)$	0.09	$(2,2,1)$	0.14

Table 2: A partial PI model.

Proposition 3 establishs the relation between the two types of PI models defined so far. The proof is trivial. Its converse is not true in general since variables in some subset of a partial PI model may be pairwise dependent as in Table 2.

Proposition 3 *A full PI model is a partial PI model.*

A partial PI model involves the entire set N of domain variables. An embedded PI submodel displays the same dependence pattern but involves only a proper subset of N.

Definition 4 (Embedded PI submodel) *Let a PDM be over a set* N *of generally dependent variables. A proper subset* $N' \subset N$ *(*$|N'| \geq 3$*) of variables forms an* `embedded` *PI submodel if the following two conditions hold: (S4)* N' *forms a partial PI model. (S5) The partition* $\{N_1, \ldots, N_k\}$ *of* N' *by S1' extends into* N*. That is, there is a partition* $\{A_1, \ldots, A_k\}$ *of* N *such that* $N_i \subseteq A_i$*, (*$i = 1, .., k$*), and for each* $X \in A_i$ *and each* $Y \in A_j$ *(*$i \neq j$*),* X *and* Y *are marginally independent.*

Definition 4 requires that variables in N are generally dependent. It eliminates the possibility that a proper subset is marginally independent of the rest of N.

Table 3 shows the jpd of PDM with an embedded PI model over three variables X_1, X_2 and X_3, where the marginals are $P(x_{1,0}) = 0.3,\ P(x_{2,0}) = 0.6,\ P(x_{3,0}) = 0.3,\ P(x_{4,0}) = 0.34,\ P(x_{5,0}) = 0.59$. Within the embedded PI model, the partition

$(X_1,..,X_5)$	$P(.)$	$(X_1,..,X_5)$	$P(.)$	$(X_1,..,X_5)$	$P(.)$	$(X_1,..,X_5)$	$P(.)$
$(0,0,0,0,0)$	0	$(0,1,0,0,0)$	.0018	$(1,0,0,0,0)$	.0080	$(1,1,0,0,0)$	.0004
$(0,0,0,0,1)$	0	$(0,1,0,0,1)$	.0162	$(1,0,0,0,1)$	.0720	$(1,1,0,0,1)$	.0036
$(0,0,0,1,0)$	0	$(0,1,0,1,0)$	.0072	$(1,0,0,1,0)$	.0120	$(1,1,0,1,0)$	.0006
$(0,0,0,1,1)$	0	$(0,1,0,1,1)$	.0648	$(1,0,0,1,1)$	.1080	$(1,1,0,1,1)$	.0054
$(0,0,1,0,0)$	.0288	$(0,1,1,0,0)$	.0048	$(1,0,1,0,0)$	.0704	$(1,1,1,0,0)$	.0864
$(0,0,1,0,1)$	.0072	$(0,1,1,0,1)$	.0012	$(1,0,1,0,1)$	.0176	$(1,1,1,0,1)$	.0216
$(0,0,1,1,0)$	.1152	$(0,1,1,1,0)$	.0192	$(1,0,1,1,0)$	.1056	$(1,1,1,1,0)$	.1296
$(0,0,1,1,1)$	.0288	$(0,1,1,1,1)$	.0048	$(1,0,1,1,1)$	.0264	$(1,1,1,1,1)$	.0324

Table 3: A PDM containing an embedded PI model.

consists of subsets $A = \{X_1\}$ and $B = \{X_2, X_3\}$. Outside the PI submodel, A extends to include X_4 and B extends to include X_5. Each variable in one subset is marginally independent of each variable in the other subset, e.g., $P(x_{1,1}, x_{5,0}) = P(x_{1,1})\ P(x_{5,0}) = 0.413$. Variables in the same subset are pairwise dependent, e.g., $P(x_{2,1}, x_{3,0}) = 0.1 \neq P(x_{2,1})\ P(x_{3,0}) = 0.12$. The three variables in the submodel are collectively dependent, e.g., $P(x_{1,1}|x_{2,0}, x_{3,1}) = 0.55$ and $P(x_{1,1}|x_{2,0}) = P(x_{1,1}|x_{3,1}) = 0.7$. However, X_4 is independent of other variables given X_1, and X_5 is independent of other variables given X_3, e.g., $P(x_{5,1}|x_{2,0}, x_{3,0}, x_{4,0}) = P(x_{5,1}|x_{3,0}) = 0.9$.

Since variables in a PI submodel are collectively dependent, in a minimal I-map of the PDM, the submodel is complete. Figure 1 shows I-maps for PI models in Table 1 through 3.

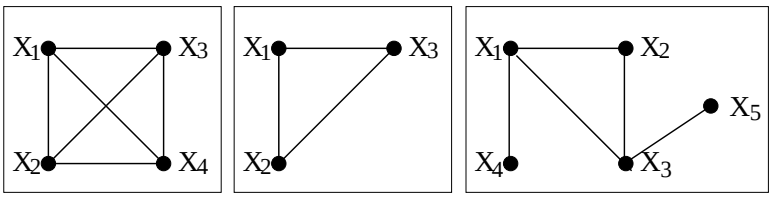

Figure 1: The I-maps for PI models in Table 1 (left), 2 (middle) and 3 (right).

Note that a PI model is *not* something we wish to construct, but rather is a problem domain of interest. We usually don't know a priori whether a domain is PI or not. What we try to construct (learn) is a BN faithfully representing the domain no matter it is PI or not.

4 PARAMETERIZATION OF PI DOMAINS

A general discrete probability distribution has $k^n - 1$ independent parameters (values) where n is the number of variables and k is the number of possible values each variable can take. A PI model is more constrained and has less parameters. We study how a PI model is composed of these parameters. Such an understanding can provide hints for how to learn these models, can provide a direct method to simulate these models which can then be used to test our learning algorithms, and can guide us in determining what problem domains may or may not contain a PI model.

In this section, we present results on parameterization of full PI models as our progress towards parameterization of general PI models. The number of parameters of a PDM that contains an embedded PI submodel depends on the number of parameters

of the PI submodel and the number of parameters in the rest of the model. Hence the parameterization of general PI models depends on the parameterization of partial PI models. By Proposition 3, full PI models are special cases of partial PI models. We therefore have chosen to tackle first the parameterization of full PI models. The following Theorem 5 applies to a *general* (vs binary) *full* PI model and Theorem 8 applies to a *general binary* (vs uniform marginal) *full* PI model.

Theorem 5 *The total number of parameters of a full PI model is* $W = W_1 + W_2$. *The number* W_1 *is the count of marginal parameters (marginals),*

$$W_1 = \sum_{i=1}^{n}(D_i - 1),$$

where n *is the total number of variables and* $D_i \geq 2$ *is the number of values that the* i*th variable can take. The number* W_2 *is the count of joint probability parameters (joints),*

$$W_2 = 1 + \sum_{i=1}^{n}\sum_{j=1}^{C(n,i)}\prod_{k=1, X_{j_k}\in Y_j}^{i}(D_{j_k} - 2),$$

where j *ranges from 1 to the total number of combinations taking* i *variables out of* n *each time,* $Y_j = \{X_{j_1}, ..., X_{j_i}\}$ *denotes one combination of* i *variables, and* D_{j_k} *is the number of values that the variable* X_{j_k} *can take.*

Proof:

In a full PI model, each variable may have a different marginal distribution, which implies W_1.

To derive W_2, we assume that the W_1 marginals have been specified and we then determine the value for each joint. We group the joints according to the number of variables taking non-zero values. For example, the group GP_0 contains a single joint $P(x_{1,0}, \ldots, x_{n,0})$, and the group GP_1 contains joints $P(x_{1,l_1}, x_{2,0}, \ldots, x_{n,0})$, $P(x_{1,0}, x_{2,l_2}, x_{3,0}, \ldots, x_{n,0})$, $\ldots$, where $l > 0$. We determine the values of joints group by group in ascending order of the group index.

The single joint of GP_0 is not uniquely determined by the W_1 marginals. We can specify this joint subject to the constraint $P(x_{1,0}, \ldots, x_{n,0}) \neq \prod_{i=1}^{n} P(x_{i,0})$ due to S2. This gives the first term, 1, in the formula for W_2.

Next, we consider GP_i ($i \geq 1$) where each joint has i variables with non-zero values. There are $C(n, i)$ ways to choose the i variables. Denote each set of i variables by $Y_j = \{X_{j_1}, ..., X_{j_i}\}$, where $1 \leq j \leq C(n, i)$. Denote the set of joints in GP_i associated with Y_j by J. For each joint in J, each X_{j_k} ($1 \leq k \leq i$) may take any value from $\{1, \ldots, D_{j_k} - 1\}$, and hence $|J| = \prod_{k=1, X_{j_k}\in Y_j}^{i}(D_{j_k} - 1)$. If we restrict the value range to $\{1, \ldots, D_{j_k} - 2\}$, there are exactly $\prod_{k=1, X_{j_k}\in Y_j}^{i}(D_{j_k} - 2)$ distinct such joints in J. Denote this subset of joints by J'. None of the joints in J' can be uniquely determined by the other joints in J' plus the W_1 marginals, the joints in $GP_0, \ldots, GP_{i-1}$, S1 and S2. However, once we have specified joints in J', each joint in $J \setminus J'$ is uniquely determined through S1, e.g.,

$$\sum_{j=0}^{D_i - 1} P(x_{1,l_1}, \ldots, x_{i-1,l_{i-1}}, x_{i,j}, x_{i+1,0}, \ldots, x_{n,0}) = (\prod_{k=1}^{i-1} P(x_{k,l_k}))(\prod_{k=i+1}^{n} P(x_{k,0})),$$

where $l > 0$. Hence the contribution of GP_i to W_2 is $\sum_{j=1}^{C(n,i)} \prod_{k=1, X_{j_k} \in Y_j}^{i} (D_{j_k} - 2)$. The formula for W_2 now follows. $\square$

For a full PI model over four trinary variables, $W_1 = 8,\ W_2 = 16,\ W = 24$ (compare with $3^4 - 1 = 80$ parameters for a general jpd). If one of the variables is binary, then $W_1 = 7,\ W_2 = 8,\ W = 15$ (compare with $2 * 3^3 - 1 = 53$). A binary full PI model of n variables has a very small number of parameters. Since $W_1 = n$ and $W_2 = 1$, we have $W = n + 1$ (compare with $2^n - 1$). It differs from a marginally independent model of the same number of variables by just one parameter.

A general partial PI model will have more parameters than a full PI model with the same set of variables. This is because a proper subset of variables in a general partial PI model may be dependent on each other. Additional parameters are needed to determine exactly how they are dependent on each other within the subset. Our current research is attempting the parameterization of general partial PI models.

Note that although W parameters can be *non-uniquely* specified in a full PI model, they *cannot* be specified *independently*, but rather must follow S1 and S2. Propositions 6, 7 and their combination Theorem 8 show how S1 and S2 constrain the W parameters in the binary case.

Proposition 6 *The jpd of a full PI model over $n \geq 3$ binary variables has the following form:*

$$P(x_{1,0}, .., x_{n,0}) = P(x_{n,0}|x_{1,0}, .., x_{n-1,0}) \prod_{i=1}^{n-1} P(x_{i,0}), \tag{1}$$

$$P(x_{l_1,1}, .., x_{l_w,1}, x_{l_{w+1},0}, .., x_{l_n,0}) =$$

$$\frac{\prod_{i=1}^{n} P(x_{i,0})}{\prod_{i=1}^{w} P(x_{l_i,0})} \sum_{i=1}^{w} (-1)^{i+1} (\prod_{j=1}^{w-i} P(x_{l_j,1}))(\prod_{k=w-i+2}^{w} P(x_{l_k,0})) + (-1)^w P(x_{1,0}, .., x_{n,0}) \tag{2}$$

where $w = 1, .., n$, and $\{l_1, ..., l_w\}$ and $\{l_{w+1}, ..., l_n\}$ form a partition of $\{1, ..., n\}$,

$$P(x_{n,0}) = \max_i P(X_i), \tag{3}$$

$$P(x_{n,0}|x_{1,0}, .., x_{n-1,0}) \neq P(x_{n,0}), \tag{4}$$

and the largest w ($w \geq 2$) probabilities $P(X_{l_i})$ ($i = 1, .., w$) satisfy

$$\sum_{i=1}^{w} (-1)^{i+1} (\prod_{j=1}^{w-i} (1 - P(X_{l_j})))(\prod_{k=w-i+2}^{w} P(X_{l_k})) + (-1)^w \frac{P(x_{n,0}|x_{1,0}, .., x_{n-1,0})}{P(x_{n,0})} \prod_{i=1}^{w} P(X_{l_i}) \geq 0. \tag{5}$$

Proof:

Since we are interested in probabilistic domains, we assume that all marginals are in $(0, 1)$ instead of [0,1]. To simplify the notation, we denote

$$P(x_{1,0}, .., x_{i-1,0}, x_{i,1}, x_{i+1,0}, .., x_{j-1,0}, x_{j+1,0}, .., x_{n,0})$$

by $R(i|j)$ where the vertical bar is not to be confused with the conditioning bar in $P(X_i|X_j)$. That is, the tuple has $X_i = 1$, X_j marginalized out, and $X_k = 0$ for all $k \neq i, j$. We denote $P(x_{1,0}, .., x_{n,0})$ by $R(.)$ and denote $P(x_{1,0}, .., x_{j-1,0}, x_{j+1,0}, .., x_{n,0})$ by $R(|j)$. We group $P(X_1, .., X_n)$ into $GP_0, .., GP_n$ where GP_i contains probabilities of tuples in which exactly i variables take value 1.

Suppose the domain is a full PI model, then S1 and S2 hold. S1 implies that for each $n-1$-tuple, its probability is the product of the corresponding marginals. That is, the domain satisfies n such constraints one for each subset of $n-1$ variables.

Let p denote $P(x_{n,0}|x_{1,0},..,x_{n-1,0}) \in [0,1]$. Due to S1, we have

$$R(.) = R(|n)\ p = P(x_{1,0})..P(x_{n-1,0})\ p.$$

Due to S2, we have $R(.) = P(x_{1,0})..P(x_{n-1,0})\ p \neq P(x_{1,0})..P(x_{n,0})$. Hence we have equation (1) and condition (4), and the single probability $R(.)$ in GP_0 is derived.

Next we consider the probabilities in GP_w $(1 \leq w \leq n)$. Without losing generality, we derive $R(1,..,w)$.

$$\begin{aligned}
& R(1,..,w) = R(1,..,w-1|w) - R(1,..,w-1) \\
= \ & R(1,..,w-1|w) - R(1,..,w-2|w-1) + R(1,..,w-2) \\
= \ & \sum_{i=1}^{w}(-1)^{i+1}\ R(1,..,w-i|w-i+1) + (-1)^w\ R(.) \\
= \ & \frac{\prod_{i=1}^{n} P(x_{i,0})}{\prod_{i=1}^{w} P(x_{i,0})} \sum_{i=1}^{w}(-1)^{i+1}(\prod_{j=1}^{w-i} P(x_{j,1}))(\prod_{k=w-i+2}^{w} P(x_{k,0})) + (-1)^w P(x_{1,0},..,x_{n,0})
\end{aligned}$$

This is equation (2). Next, we use induction to derive the condition under which $R(1,..,w)$ is a valid probability. We consider first $R(j)$ in GP_1. We have

$$R(j) = (\prod_{i=1}^{n} P(x_{i,0}))/P(x_{j,0}) - (\prod_{i=1}^{n-1} P(x_{i,0}))\ p = (\prod_{i=1}^{n-1} P(x_{i,0}))\ (P(x_{n,0}) - P(x_{j,0})\ p)/P(x_{j,0}).$$

It is a valid probability iff $P(x_{n,0}) \geq P(x_{j,0})\ p$. This is always satisfied if $P(x_{n,0}) = \max_i P(X_i)$ which is condition (3).

Without losing generality, we then consider $R(1,j)$ $(1 < j \leq n)$:

$$R(1,j) = \frac{\prod_{i=1}^{n} P(x_{i,0})}{P(x_{1,0})\ P(x_{j,0})}\ (P(x_{1,1}) - P(x_{j,0}) + \frac{P(x_{1,0})\ P(x_{j,0})\ p}{P(x_{n,0})}).$$

For it to be a valid probability, we must have

$$P(x_{1,1}) - P(x_{j,0}) + \frac{P(x_{1,0})\ P(x_{j,0})\ p}{P(x_{n,0})} \geq 0.$$

We shall view the left-hand side as a function $f(P(x_{1,0}))$ of $P(x_{1,0})$ with other parameters held constant. Note that $P(x_{i,0}) = 1 - P(x_{i,1})$. The first order derivative of $f(P(x_{1,0}))$ is $f'(P(x_{1,0})) = -1 + P(x_{j,0})\ p/P(x_{n,0})$. We have $f'(P(x_{1,0})) \leq 0$ *since* $P(x_{n,0}) \geq P(x_{j,0})\ p$. This implies that $f(P(x_{1,0}))$ is non-increasing with $P(x_{1,0})$, i.e., if $f(P(x_{1,0})) \geq 0$ then for any $\alpha < P(x_{1,0})$, we have $f(\alpha) \geq 0$. Since $P(x_{1,0}), P(x_{j,0})$ and all marginals are symmetric, we conclude that if $1 - P(X_i) - P(X_j) + \frac{P(X_i)\ P(X_j)\ p}{P(x_{n,0})} \geq 0$s holds for the largest marginals $P(X_i)$ and $P(X_j)$, then each value in GP_2 defined by equation (2) is a valid probability. We have thus verified condition (5) for $w = 2$.

We now make the inductive assumption that each value in GP_{w-1} ($w > 2$) is a valid probability if condition (5) holds. We show that each value in GP_w is a valid probability if condition (5) holds. Without losing generality, we consider $R(1,..,w)$. For $R(1,..,w)$ to be a valid probability, it must be the case

$$\sum_{i=1}^{w}(-1)^{i+1}(\prod_{j=1}^{w-i}P(x_{j,1}))(\prod_{k=w-i+2}^{w}P(x_{k,0}))+(-1)^w\frac{p}{P(x_{n,0})}\prod_{i=1}^{w}P(x_{i,0})\geq 0.$$

We shall view the left-hand side as a function $f(P(x_{1,0}))$ of $P(x_{1,0})$ with other parameters held constant. The first order derivative of $f(P(x_{1,0}))$ is

$$f'(P(x_{1,0}))=\sum_{i=1}^{w-1}(-1)^{i}(\prod_{j=2}^{w-i}P(x_{j,1}))(\prod_{k=w-i+2}^{w}P(x_{k,0}))+(-1)^w\frac{p}{P(x_{n,0})}\prod_{i=2}^{w}P(x_{i,0}).$$

From the inductive assumption on GP_{w-1}, we know

$$-[\sum_{i=1}^{w-1}(-1)^{i}(\prod_{j=2}^{w-i}P(x_{j,1}))(\prod_{k=w-i+2}^{w}P(x_{k,0}))+(-1)^w\frac{p}{P(x_{n,0})}\prod_{i=2}^{w}P(x_{i,0})]\geq 0.$$

Thus $f'(P(x_{1,0})\leq 0$, i.e., $f(P(x_{1,0}))$ is non-increasing with $P(x_{1,0})$. This implies that if $f(P(x_{1,0}))\geq 0$, then for any $\alpha < P(x_{1,0})$, we have $f(\alpha)\geq 0$. Since $P(x_{1,0}),..,P(x_{w,0})$ and all marginals are symmetric, we conclude that if condition (5) holds for the largest w marginals, then each value in GP_w is a valid probability. The proposition is proven. □

Proposition 7 *A PDM over $n \geq 3$ binary variables is a full PI model if its jpd satisfies equations (1) and (2) subject to conditions (3) through (5).*

Proof: Since equation (1) and condition (4) hold, S2 is true. To show S1 holds, it suffices to show $R(.)+R(j)=\prod_{i=1}^{j-1}P(x_{i,0})\prod_{i=j+1}^{n}P(x_{i,0})$ and $R(1,..,w)+R(1,..,w+1)=\prod_{i=1}^{w}P(x_{i,1})\prod_{i=w+2}^{n}P(x_{i,0})$.

From equations (1) and (2) with $w=1$, we obtain $R(.)+R(j)=(\prod_{i=1}^{n}P(x_{i,0}))/P(x_{j,0})$.

From equation (2), we obtain

$$\begin{aligned}
& R(1,..,w)+R(1,..,w+1)\\
= & \frac{\prod_{i=1}^{n}P(x_{i,0})}{\prod_{i=1}^{w}P(x_{i,0})}\sum_{i=1}^{w}(-1)^{i+1}(\prod_{j=1}^{w-i}P(x_{j,1}))(\prod_{k=w-i+2}^{w}P(x_{k,0}))+\\
& \frac{\prod_{i=1}^{n}P(x_{i,0})}{\prod_{i=1}^{w+1}P(x_{i,0})}\sum_{i=1}^{w+1}(-1)^{i+1}(\prod_{j=1}^{w-i+1}P(x_{j,1}))(\prod_{k=w-i+3}^{w+1}P(x_{k,0}))\\
= & \frac{\prod_{i=1}^{n}P(x_{i,0})}{\prod_{i=1}^{w+1}P(x_{i,0})}[\sum_{i=1}^{w}(-1)^{i+1}(\prod_{j=1}^{w-i}P(x_{j,1}))(\prod_{k=w-i+2}^{w+1}P(x_{k,0}))+\\
& \sum_{i=1}^{w+1}(-1)^{i+1}(\prod_{j=1}^{w-i+1}P(x_{j,1}))(\prod_{k=w-i+3}^{w+1}P(x_{k,0}))]\\
= & \frac{\prod_{i=1}^{n}P(x_{i,0})}{\prod_{i=1}^{w+1}P(x_{i,0})}\prod_{j=1}^{w}P(x_{j,1}). \qquad \square
\end{aligned}$$

Combining Propositions 6 and 7, we obtain the following theorem.

Theorem 8 *A PDM over $n \geq 3$ binary variables is a full PI model iff its jpd satisfies equations (1) and (2) subject to conditions (3) through (5).*

Equations (1) and (2) describe how each joint probability is composed of the $n+1$ parameters $P(x_{i,0})$ $(i = 1, \ldots, n)$ and $P(x_{n,0}|x_{1,0}, \ldots, x_{n-1,0})$. Conditions (3) through (5) describe the constraints that they must observe.

5 DO PI DOMAINS EXIST?

Are PI models simply mathematical constructs without practical ground? Theorem 8 shows how easy it is to form a binary full PI model. Replace $P(x_{n,0}|x_{1,0}, \ldots, x_{n-1,0})$ in condition (5) by a parameter $r \in [0, 1]$. Suppose $r \neq P(x_{n,0})$ exists such that all marginals of a truly marginally independent domain are in the right range dictated by condition (3) and the modified condition (5). To turn this domain into a PI model, all it takes is to change $P(x_{n,0}|x_{1,0}, \ldots, x_{n-1,0})$ from $P(x_{n,0})$ to r. This analysis provides evidence that PI models should not be rare in real world domains. In the following, we provide more evidence by analyzing two special cases of PI models.

It is well known that parity problems cause difficulty to many machine learning algorithms, see for example [4, 3, 6]. A parity problem can be described as follows: A set of marginally independent input variables $\{X_1, .., X_{n-1}\}$ each take the value 0 or 1 with an equal chance. An output variable X_n takes 0 or 1 such that the total number of 1's is even (for even parity).

The following proposition shows that parity problems form a special case of binary full PI models.

Proposition 9 *A parity problem over n variables is a binary full PI model with $P(X_i) = 0.5$ $(i = 1, .., n)$. For even parity, $P(x_{n,0}|x_{1,0}, .., x_{n-1,0}) = 1$, and for odd parity, $P(x_{n,0}|x_{1,0}, .., x_{n-1,0}) = 0$.*

Proof: It suffices to show the even parity case due to the symmetry. From the problem description, clearly $P(X_i) = 0.5$ $(i = 1, .., n-1)$ and $P(x_{n,0}|x_{1,0}, .., x_{n-1,0}) = 1$. We first show $P(X_n) = 0.5$. For the set of $n-1$ input variables, there are 2^{n-1} $n-1$-tuples each occurring with probability 0.5^{n-1}. Exactly half of them contain an even number of 0's. Denote this group of 2^{n-2} $n-1$-tuples by G_e and the other group of 2^{n-2} $n-1$-tuples by G_o.

Each $n-1$-tuple extends into two n-tuples when X_n is added. If a $n-1$-tuple is in G_e, the n-tuple with $X_n = 0$ has non-zero probability. If a $n-1$-tuple is in G_o, the n-tuple with $X_n = 0$ has zero probability. Hence we have $P(x_{n,0}) = 2^{n-2} * 0.5^{n-1} = 0.5$, the probability by which a $n-1$-tuple belongs to G_e.

Next, we show that equations (1) and (2) are satisfied subject to conditions (3) through (5). It is trivial to verify equation (1) and conditions (3) and (4). The right-hand side of equation (2) becomes $0.5^{n-1}((-1)^w - \sum_{i=1}^{w}(-1)^i)$ whose value is 0.5^{n-1} when w is even, and 0 otherwise. This is the expected form for the jpd. The left-hand side of condition (5) becomes

$$\sum_{i=1}^{w}(-1)^{i+1}0.5^{w-1} + (-1)^w 0.5^{w-1} = 0.5^{w-1}[\sum_{i=1}^{w}(-1)^{i+1} + (-1)^w].$$

When w is even, the sum is 0.5^{w-1}. When w is odd, the sum is 0. □

Note that the distinction between a unique output variable X_n and the other input variables in a parity problem is unnecessary. Once the n variables behave according to a particular parity, we would not be able to tell which one is the output variable since each variable has the same marginals and each may assume the role of the output.

Recently, it was shown [6] that parity problems are special cases of modulus addition problems. The latter display similar properties of parity problems and cause difficulty to ID3-like algorithms. A modulus addition problem can be described as follows: A problem domain consists of a set of marginally independent and uniformly distributed input variables $\{X_1, .., X_{n-1}\}$ and an output variable X_n. Each X_i $(i = 1, ..., n)$ has the domain $\{0, 1, \ldots, D_i - 1\}$ where $D_i \geq 2$ such that for each $i < n$, $D_i = k_i\ D_n$, where k_i is a positive integer. X_n is the sum of $X_1, .., X_{n-1}$ modulo D_n.

Proposition 10 shows that modulus addition problems are also special cases of full PI models (not restricted to binary).

Proposition 10 *A modulus addition problem is a full PI model.*

Proof:

It suffices to show that S1 and S2 holds. That S2 holds is trivially true. To show that S1 holds, we need only to show $P(X_n|X_1, .., X_{n-2}) = P(X_n)$. First,

$$\begin{aligned} P(X_n|X_1, .., X_{n-2}) &= \sum_{X_{n-1}} P(X_n X_{n-1}|X_1, .., X_{n-2}) \\ &= \sum_{X_{n-1}} P(X_n|X_1, .., X_{n-1}) P(X_{n-1}|X_1, .., X_{n-2}) \\ &= \sum_{X_{n-1}} P(X_n|X_1, .., X_{n-1}) P(X_{n-1}) = (1/D_{n-1}) \sum_{X_{n-1}} P(X_n|X_1, .., X_{n-1}) \end{aligned}$$

Since X_n is determined given $X_1, .., X_{n-1}$, $P(X_n|X_1, .., X_{n-1})$ is either 0 or 1. Given the values of $X_1, .., X_{n-2}$, as X_{n-1} takes values $0, .., D_{n-1} - 1$ in sequence, X_n will go through each value in its domain exactly k_{n-1} times. Hence the summation in the above equation equals to k_{n-1} and we obtain

$$P(X_n|X_1, .., X_{n-2}) = k_{n-1}/D_{n-1} = 1/D_n.$$

Due to the symmetry of inputs, the above equation holds for any $n - 2$ inputs. That is, the conditioning is irrelevant (removable), which implies $P(X_n|X_1, .., X_{n-2}) = P(X_n) = 1/D_n$. □

The above analysis provides positive evidence for the existence of PI models in practice. Due to their existence, we cannot blindly apply unreliable learning algorithms to just any problem domain as the quality of learning outcomes will be unpredictable.

6 APPLICATION OF RESULTS TO LEARNING

We have shown that PI domains do exist. Common algorithms for learning belief networks are not *reliable* in the sense that they cannot learn an approximate I-map when the domain is PI. Development of more *reliable* algorithms depends highly on a better understanding of these domains.

Our progress towards such a goal includes formal characterization of all three types of PI models. This formalization provides a basis for the study of these models. We also provided a parameterization of full PI models that laid some groundwork towards the parameterization of a general PI model. Such parameterization is directly useful to learning in several ways:

1. It provides hints for how to learn these models effectively, which will facilitate the design of new learning algorithms.

2. It guides us in determining whether a problem domain may be a PI model. For instance, if we know that a particular subdomain has more parameters than dictated by Theorem 5, then we are confident that this subdomain cannot be a full PI model. Such information is useful for configuring the learning algorithms.

3. It provides direct methods for simulation of PI models to be used in testing learning algorithms. For example, Theorem 8, Propositions 9 and 10 provide direct methods to simulate full PI models.

4. It provides hints for compact representation of PI submodels in a learned BN.

Acknowledgement

This work is supported by grant OGP 0155425 and CRD 193296 from the Natural Sciences and Engineering Research Council, and a grant from the Institute for Robotics and Intelligent Systems in the Networks of Centres of Excellence Program of Canada.

References

[1] G.F. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, (9):309–347, 1992.

[2] D. Heckerman, D. Geiger, and D.M. Chickering. Learning Bayesian networks: the combination of knowledge and statistical data. *Machine Learning*, 20:197–243, 1995.

[3] G.H. John, R. Kohavi, and K. Pfleger. Irrelevant features and the subset selection problem. In *Proc. 11th Inter. Conf. on Machine Learning*, pages 121–129, 1994.

[4] G. Pagallo and D. Haussler. Boolean feature discovery in empirical learning. *Machine Learning*, (5):71–99, 1990.

[5] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[6] C. Thornton. Parity: the problem that won't go away. In G. McCalla, editor, *Advance in Artificial Intelligence*, pages 362–374. Springer, 1996.

[7] Y. Xiang, S.K.M. Wong, and N. Cercone. Critical remarks on single link search in learning belief networks. In *Proc. 12th Conf. on Uncertainty in Artificial Intelligence*, pages 564–571, Portland, 1996.

[8] Y. Xiang, S.K.M. Wong, and N. Cercone. A 'microscopic' study of minimum entropy search in learning decomposable Markov networks. *Machine Learning*, 26(1):65–92, 1997.