Learning a contingently acyclic, probabilistic relational model of a social network

UBC Technical Report TR-2009-08

Peter Carbonetto, Jacek Kisyński, Michael Chiang and David Poole
Dept. of Computer Science
University of British Columbia
Vancouver, B.C., Canada V6T 1Z4

April 6, 2009

Abstract

We demonstrate through experimental comparisons that modeling relations in a social network with a directed probabilistic model provides a viable alternative to the standard undirected graphical model approach. Our model incorporates special latent variables to guarantee acyclicity. We investigate the inference and learning challenges entailed by our approach.

1 Introduction

It has long been known in the sciences that social context matters. Epidemiologists, for instance, have studied the spread of infectious diseases like HIVs with social networks, while sociologists have studied how risk-taking behaviours are learned in social peer groups (Pearson & Michell, 2000). Broadly speaking, social network analysis is concerned with the nature of relationships, and how the structure of relationships influences other processes. The point of departure for statistical representations of social network structure is the class of models proposed by Frank and Strauss (1986) and Besag (1974), now known as *exponential random graph models* or $p^\star$ (Carrington et al., 2005). This modeling approach relies on the (famously unpublished) theorem of Hammersley and Clifford to provide the necessary link between a preliminary dependency analysis and the final probabilistic model. The argument for this approach is that social relations are intrinsically interdependent with no obvious form of causation, so the aim is to develop models that hypothesize possible forms of interdependence, or "autocorrelation."

This class of models, also known to many researchers as *undirected probabilistic graphical models* or *Markov random fields*, has witnessed a resurgence of popularity in other well-explored domains, notably computer vision and collaborative filtering. The key, again, is that such formalisms naturally represent interdependence, such as constancy of motion in neighbouring image pixels (Sun et al., 2008). hydrogen bond interactions in secondary protein structure (Muñoz & Eaton, 1999) or similar tastes in movies (Salakhutdinov et al., 2007), and they can incorporate simple factors to form rich, predictive models without having to worry about avoiding cycles in the underlying graph.

The undirected formalism is not without its problems, however. First, the difficulties of learning the model parameters—for instance, by maximizing the likelihood of the model given the data—are well-noted (Hunter et al., 2008). Another possible approach is to compute the maximum likelihood estimator via stochastic approximation (Younes, 1991), but this may involve repeated, computationally intensive simulations of a Markov chain. In some cases, the contrastive divergence approximation provides a more realistic alternative (Hinton, 2002). Due

to these difficulties, the *pseudo-likelihood* approximation still appears in the literature, despite severe criticism of its use (Snijders, 2002). Furthermore, undirected graphical models scale poorly to large social networks, including models that boast compact first-order representations, because inferring the result of a query always implicates every node in the network—even parts of the network for which we have no information. Because of this, it is argued, undirected models can be poorly suited for prediction in the presence of missing data (Marlin, 2008).

The main contribution of this paper is to show through experiment that a directed probabilistic model (Spiegelhalter et al., 1993) is an equally viable representation of "interdependent" relations in a non-trivial social network domain, in addition to having several important advantages, as we discuss below. We formulate a directed model that explains how people alter their smoking habits within their social network (Sec 2), and in a series of experiments (Sec. 4) we compare it to an undirected model—to be precise, a Markov logic network (Richardson & Domingos, 2006).[1] We introduce special latent random variables (related to the hypothesis variables in *multinets*; see Geiger & Heckerman, 1996) to ensure that the directed graph is *contingently acyclic* (Poole, 2000), a notion which is grounded on *context-specific independence* (Boutilier et al., 1996). (These variables also have an interesting interpretation in the social network domain, as we explain in Sec. 2.) The introduction of cycle-resolving latent variables allows us to surpass the representational limitations of directed graphical models caused by the need to avoid cycles.

Three main advantages of our directed representation over existing undirected social network models are that it is easier to learn (for instance, under complete information the maximum likelihood solution is easily obtained), the probabilities have a local interpretation as conditionals, and irrelevant nodes can be pruned from the directed graph (Shachter, 1998). For example, whenever we have no information about two individuals, we can prune the friendship relation between them. This is an important step for extending statistical models to large or infinite-sized domains. An alternative way to capture interdependencies in a directed model is to permit cycles, as in Relational Dependency Networks (Neville & Jensen, 2007), although this approach still shares the learning difficulties of undirected models. We use the Independent Choice Logic (Poole, 1997) to define our model, though it could also be written as a program in BLOG (Milch et al., 2005), for instance.

Since the cycle-resolving variables are not observed, we use the expectation maximization (EM) algorithm to learn the parameters of the social network model (Sec. 3). When all friendship and smoking relations are observed, the corresponding factor graph is highly interconnected so we must approximate inference in the E-step. (To be clear, a directed graph without cycles can still correspond to an undirected graph or factor graph with cycles.) Due to the particular structure of our network, a variational approximation based on the Bethe decomposition of the free energy is well-suited for this task (Heskes, 2006; Yedidia et al., 2005). Our study leads to unsolved problems that would be of interest to people researching new and better tractable inference and learning algorithms.

2 Description of the model

We describe an idealized relational probabilistic model of the relationship between smoking habits and the formation of friendships ("link existence"), a prototypical example of a relational domain where individuals influence each other. Our intent is to investigate the modeling and inference challenges that arise from studying a social network domain, not to construct a scientifically plausible analysis of smoking and risk-taking behaviour.

The conditional independence structure of our directed graphical model cannot be captured as a belief network (Pearl, 1988) because we don't know beforehand what is the set of parents of a random variable. In this capacity, our model fits within the definition of a *contingent Bayesian network* (Milch, 2006). Also, our representation is at a first-order level; we reason about relationships regarding collections of individuals. Since the independence relationships are only known when conditioned on certain random variables, and since the Independent Choice Logic of Poole (1997) naturally and compactly captures contingent (or context-specific) independencies at a first-order level, we define our model in ICL.

[1] We do not compare to discriminative undirected models because they explain entity attributes given social links (Taskar et al., 2002) and link existence given entity attributes (Taskar et al., 2004), but not both simultaneously.

2.1 Preliminaries

Preliminaries. We follow the Prolog convention and write logical variables in upper case, and predicate symbols in lower case. Throughout, X and Y refer to individuals; that is, they are logical variables whose domain is the set of people. The predicate $\text{smokes}(X) = \text{true}$ if and only if X smokes, and $\text{friends}(X, Y) = \text{true}$ if and only if X and Y are friends. Given an assignment of individuals to the logical variables, the predicates correspond to Boolean random variables. We define friendship as a symmetric, irreflexive relation, and enforce this constraint via some arbitrary total ordering $X \prec Y$ on the individuals.

In the social network, interdependencies arise between friendship and smoking. For example, a non-smoker might convince a friend to quit smoking, or the similar lifestyle choices of two smokers might make them more likely to become friends. A causal, temporal model might form an accurate description of these interdependencies, but it would be unwise to attempt to infer the history of events leading up to the present state.

In our directed probabilistic model, we regulate the direction of influence through a hidden predicate $\text{ind}(X)$, and learn a distribution over it. For each individual X, $\text{ind}(X)$ tells us, loosely speaking, whether X's decision to smoke is based on social factors, or whether it is governed other factors that are not captured by our model (*i.e.* X makes an *independent* decision to smoke). When $\text{ind}(X) = \text{true}$, X can persuade others to smoke (or not to smoke), but X cannot be persuaded. This is a coarse-grained depiction of influence, and there are many alternatives for analyzing interdependencies at a propositional level; for instance, Alice could influence Bob only if Bob does not influence Alice, either directly or indirectly through other people. However, it is inordinately difficult to learn propositional rules such as this one, and they may not be useful in new situations. Our first-order rules are simple, easily transferable and, as we show, work reasonably well.

We now proceed to define our ICL theory for the social network domain. An ICL theory consists of two parts: a deterministic controller specified as a logic program, and noisy inputs that comprise the *choice space*. (Virtually all probabilistic programming languages could be described as a combination of deterministic controller and noisy inputs.) The logic program consists of a set of clauses, and each clause is either an *atom*—for our purposes, an atom is of the form $r(t_1, t_2, \ldots)$ where r is a predicate symbol and each t_i is either a logical variable or a constant—or a rule of the form $h \leftarrow a_1 \wedge \ldots \wedge a_k$, where h is an atom and each a_i is either an atom or its negation. ICL requires that the logic program be *contingently acyclic* (Poole, 2000).

The noisy inputs are called *atomic choices* in ICL, denoted as ground instances of $\phi_k(X)$ or $\phi_k(X, Y)$ in the clauses below. Each $\phi_k(X)$ or $\phi_k(X, Y)$ can appear in the body of a rule, but not the head of a clause. Of particular interest are the atomic choices $\phi_0(X)$, introduced above as $\text{ind}(X)$. Our social network model has a very simple choice space, so we do not introduce ICL's general syntax for choice spaces.

2.2 Logic program

The rules for friendship are as follows. When $\text{ind}(X)$ and $\text{ind}(Y)$ are true, $\text{smokes}(X)$ and $\text{smokes}(Y)$ can legitimately be parents of $\text{friends}(X, Y)$ without creating a cycle in the directed graph, so we define clauses

$$\begin{aligned}
&\text{friends}(X,Y) \leftarrow X \prec Y \wedge \text{ind}(X) \wedge \text{ind}(Y) \\
&\qquad \wedge \neg\,\text{smokes}(X) \wedge \neg\,\text{smokes}(Y) \wedge \phi_1(X,Y) \\
&\text{friends}(X,Y) \leftarrow X \prec Y \wedge \text{ind}(X) \wedge \text{ind}(Y) \\
&\qquad \wedge \neg\,\text{smokes}(X) \wedge \text{smokes}(Y) \wedge \phi_2(X,Y) \\
&\text{friends}(X,Y) \leftarrow X \prec Y \wedge \text{ind}(X) \wedge \text{ind}(Y) \\
&\qquad \wedge \text{smokes}(X) \wedge \neg\,\text{smokes}(Y) \wedge \phi_2(X,Y) \\
&\text{friends}(X,Y) \leftarrow X \prec Y \wedge \text{ind}(X) \wedge \text{ind}(Y) \\
&\qquad \wedge \text{smokes}(X) \wedge \text{smokes}(Y) \wedge \phi_3(X,Y).
\end{aligned} \tag{1}$$

For those more familiar with Bayesian networks, it is instructive to see how the clauses above correspond to a conditional probability table (CPT). The clauses state that if both $\text{ind}(X)$ and $\text{ind}(Y)$ are true, then the corresponding entries of the CPT for $\text{friends}(X, Y)$ are

$$\begin{aligned}
&p(\text{friends}(X,Y) = \text{true} \mid \text{smokes}(X), \text{smokes}(Y), \\
&\qquad\qquad \text{ind}(X) = \text{true}, \text{ind}(Y) = \text{true}) \\
&= \begin{cases}
\text{smokes}(X) & \text{smokes}(X) & p \\
\text{true} & \text{true} & \theta_1 \\
\text{false} & \text{true} & \theta_2 \\
\text{true} & \text{false} & \theta_2 \\
\text{false} & \text{false} & \theta_3.
\end{cases}
\end{aligned} \tag{2}$$

(The binomial probabilities θ_k will be defined in the next part on semantics.) The remaining cases for friendship are covered by the clauses

$$\begin{aligned}\text{friends}(X,Y) \leftarrow & X \prec Y \wedge \text{ind}(X) \wedge \neg\,\text{ind}(Y) \\ & \wedge \neg\,\text{smokes}(X) \wedge \phi_4(X) \\ \text{friends}(X,Y) \leftarrow & X \prec Y \wedge \text{ind}(X) \wedge \neg\,\text{ind}(Y) \\ & \wedge \text{smokes}(X) \wedge \phi_5(X),\end{aligned} \tag{3}$$

and the analogous clauses with X and Y switched, and by the clauses

$$\begin{aligned}\text{friends}(X,Y) \leftarrow & X \prec Y \wedge \neg\,\text{ind}(X) \wedge \neg\,\text{ind}(Y) \\ & \wedge \phi_6(X,Y).\end{aligned} \tag{4}$$

when both X and Y can be influenced by others. The second set of clauses (3) says that whenever exactly one of the individuals is not influenced by others, then the corresponding entries of the CPT are given by

$$\begin{aligned}p(\text{friends}(X,Y) = \text{true} \mid & \text{smokes}(X), \text{smokes}(Y), \\ & \text{ind}(X) = \text{true}, \text{ind}(Y) = \text{false}) \\ & = \begin{cases} \text{smokes}(X) & p \\ \text{true} & \theta_4 \\ \text{false} & \theta_5. \end{cases}\end{aligned} \tag{5}$$

The clause (4) specifies CPT entries when $\text{ind}(X)$ and $\text{ind}(Y)$ are both false:

$$\begin{aligned}p(\text{friends}(X,Y) = \text{true} \mid & \text{smokes}(X), \text{smokes}(Y), \\ & \text{ind}(X) = \text{false}, \text{ind}(Y) = \text{false}) = \theta_6.\end{aligned} \tag{6}$$

We also include a rule to enforce symmetry of friendship:

$$\text{friends}(X,Y) \leftarrow Y \prec X \wedge \text{friends}(Y,X). \tag{7}$$

The clauses (1), (3), (4) and (7) in combination with the choice space define conditional probability distributions (CPDs) for $\text{friends}(X,Y)$ given values for, $\text{smokes}(X)$, $\text{smokes}(Y)$, $\text{ind}(X)$ and $\text{ind}(Y)$. From the CPTs written above, it is quite apparent that the clauses above provide a much more compact representation of friendship than a CPT over all 2^6 possible assignments to the random variables. Note that the conditional probability is deterministic conditioned on atomic choices $\phi_1(X,Y)$ through $\phi_5(X,Y)$.

Rules for smoking habits are as follows. The simplest case occurs when X's friends have no bearing on X's decision to smoke:

$$\text{smokes}(X) \leftarrow \text{ind}(X) \wedge \phi_7(X). \tag{8}$$

To determine whether X smokes when $\text{ind}(X) = \text{false}$, we aggregate "advice" from smokers and non-smokers through hidden predicates smoking-advice(X) and non-smoking-advice(X), or $\text{sa}(X)$ and $\text{nsa}(X)$ for short,

$$\begin{aligned}\text{sa}(X) \leftarrow & \exists Y\ \text{friends}(X,Y) \wedge \text{ind}(Y) \\ & \wedge\ \text{smokes}(Y) \wedge \phi_8(X,Y)\end{aligned} \tag{9}$$

$$\begin{aligned}\text{nsa}(X) \leftarrow & \exists Y\ \text{friends}(X,Y) \wedge \text{ind}(Y) \\ & \wedge\ \neg\,\text{smokes}(Y) \wedge \phi_9(X,Y),\end{aligned} \tag{10}$$

and then combine the advice through the clauses

$$\begin{aligned}\text{smokes}(X) &\leftarrow \neg\,\text{ind}(X) \wedge \neg\,\text{sa}(X) \wedge \neg\,\text{nsa}(X) \wedge \phi_{10}(X) \\ \text{smokes}(X) &\leftarrow \neg\,\text{ind}(X) \wedge \neg\,\text{sa}(X) \wedge \text{nsa}(X) \wedge \phi_{11}(X) \\ \text{smokes}(X) &\leftarrow \neg\,\text{ind}(X) \wedge \text{sa}(X) \wedge \neg\,\text{nsa}(X) \wedge \phi_{12}(X) \\ \text{smokes}(X) &\leftarrow \neg\,\text{ind}(X) \wedge \text{sa}(X) \wedge \text{nsa}(X) \wedge \phi_{13}(X).\end{aligned} \tag{11}$$

Clauses (9) and (10) together with the choice space form a noisy-or aggregation over all smoking and non-smoking friends respectively for which $\text{ind}(X)$ is turned on. Following Pearl (1988), the noisy-or for advice from smokers is given by

$$\begin{aligned}p(\text{ps}(X) = \text{false} \mid & \{\text{friends}(X,Y), \text{ind}(Y), \text{smokes}(Y)\}) \\ & = (1-\theta_8)^{\text{num-smoking-friends}(X)},\end{aligned} \tag{12}$$

where num-smoking-friends(X) is defined to be the number of individuals Y such that Y is a smoker, X and Y are friends, and Y is either an independent thinker or Y is before X in the total ordering ($Y \prec X$). Rules (8-11) along with the choice space define CPDs for $\text{smokes}(X)$ given values for latent variables $\text{sa}(X)$, $\text{sa}(X)$ given values for $\text{friends}(X,Y)$, $\text{smokes}(Y)$ and $\text{ind}(Y)$ for all individuals Y, and likewise for $\text{nsa}(X)$.

2.3 Semantics

Our ICL theory consists of the collection of clauses (1), (3), (4), (7) and (8-11), and the choice space. In our

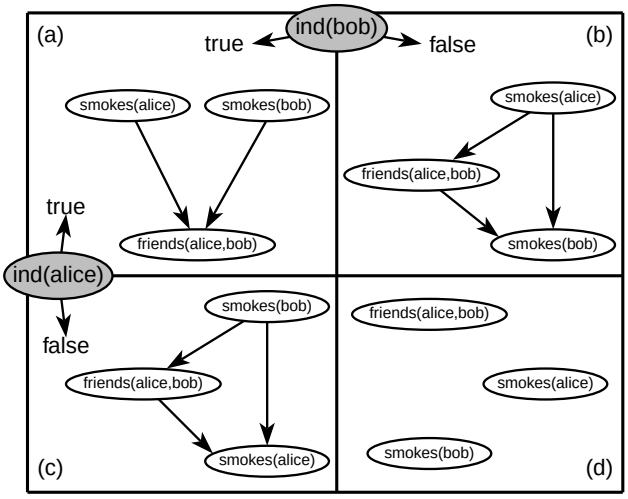

Figure 1: Illustration of how ind(X) works.

model, it suffices to say that when individuals are assigned to all logical variables, each ground instance of an atomic choice $\phi_k(X)$ follows a simple Bernoulli distribution $p_k(\nu)$ over $\nu \in \{\phi_k(X), \neg\, \phi_k(X)\}$ with probability of success θ_k. (The same then goes for each $\phi_k(X,Y)$.) These θ_k are the parameters θ of our model.

The semantics is given in terms of possible worlds. A *total choice* for choice space is a selection of exactly one atomic choice from each ground instance of $\{\phi_k(X), \neg\, \phi_k(X)\}$ or $\{\phi_k(X,Y), \neg\, \phi_k(X,Y)\}$ in the choice space. There is a *possible world* for each total choice. What is true in a possible world is defined by the atoms chosen by the total choice together with the logic program. The measure of a possible world is the product of values $p_k(\nu)$ for each ν selected by the total choice. The probability of the proposition is the sum of the measures of the possible worlds in which the proposition is true.

2.4 Acyclicity

We now elaborate on how the collection of clauses in the ICL theory above forms a contingently acyclic logic program. The propositional directed graphical model corresponding to the theory has cycles, but context-specific independence saves us: when we've assigned values to all ground instances of ind(X), the graphical model on the remaining random variables becomes acyclic.[2] As a result, our theory defines the joint probability for each configuration x of the random variables as the product

$$\begin{aligned} p(x|\theta) = {} & \textstyle\prod_X p(\text{smokes}(X) \,|\, \text{ind}(X), \text{sa}(X), \text{nsa}(X), \theta) \\ & \times \textstyle\prod_X p(\text{sa}(X)|\{\text{friends}(X,Y), \text{smokes}(Y), \text{ind}(Y)\}, \theta) \\ & \times \textstyle\prod_X p(\text{nsa}(X)|\{\text{friends}(X,Y), \text{smokes}(Y), \text{ind}(Y)\}) \\ & \times \textstyle\prod_X p(\text{ind}(X) \,|\, \theta) \times \prod_{X,Y} p(\text{friends}(X,Y) \,| \\ & \qquad \text{smokes}(X), \text{smokes}(Y), \text{ind}(X), \text{ind}(Y), \theta). \quad (13) \end{aligned}$$

Note that x, our notation for an assignment of all the random variables to binary values, has no relation to the logical variable X. This notion of contingent acyclicity is handled naturally in ICL, as the logic program becomes acyclic when values of all ground instances of ind(X) are known. We illustrate how the latent variables ind(X) ensure that we obtain a directed, acyclic graph with the following example.

2.5 An example

Refer to Fig. 1. We focus our attention on two individuals, Alice and Bob, within a larger social network domain, such that alice $\prec$ bob. The predicates ind(alice) and ind(bob) have four possible configurations, as depicted in Fig. 1. When ind(alice) $=$ true and ind(bob) $=$ true (Fig. 1a), their friendships hold no influence over their smoking habits, so their habits are allowed to influence the probability of becoming friends. In Fig. 1b, ind(alice) $=$ true and ind(bob) $=$ false. Whether Bob smokes depends on whether he is friends with Alice, and whether Alice smokes. Alice decides independently to smoke, which in turn affects her propensity to form a relationship with Bob. Note that Bob's decision to smoke also depends on other friends of his whom we haven't mentioned. Fig. 1c is the opposite case when ind(alice) $=$ false and ind(bob) $=$ true. Finally, in Fig. 1d, both ind(alice) and ind(bob) are false. Neither Alice nor Bob have influence over each other's smoking habits, although their smoking habits can still be influenced by other friends X for which ind$(X) =$ true.

[2] We caution that in general an acyclic logic program does not correspond to an acyclic directed graphical model.

3 Learning the model

During training, we observe friendships and smoking habits, and the objective is to find a collection of model parameters that maximizes the likelihood of the evidence. Since we have random variables that are not observed during training, we follow the expectation maximization (EM) recipe, which consists of iteratively choosing the parameters θ that maximize the *expected complete log-likelihood*, then computing the posterior distribution $p(x_U \mid x_E, \theta)$ of the unobserved random variables x_U given the observations or evidence x_E. In our case, x_E corresponds to the values smokes(X) for every X, and friends(X, Y) for every pair (X, Y). We clarify what are the unobserved variables x_U in Sec. 3.1. Note that each entry x_i of the random vector x corresponds to a ground instance of some predicate. The conditional independence structure does not allow us to compute expectations with respect to the posterior in a reasonable amount of time, so we adopt the approximate EM framework of Heskes et al. (2004).

If x_U could be observed, then the maximum likelihood estimator would amount to the vector θ that maximizes $\log p(x_E, x_U \mid \theta)$. Since we do not observe the x_U's, a seemingly sensible course of action would be to optimize the *incomplete log-likelihood*

$$\log \textstyle\sum_{x_U} p(x_E, x_U \mid \theta). \tag{14}$$

This, however, will be difficult to optimize because it is not clear how to exploit the conditional independence structure of the model. Suppose instead we average over the unobserved variables, and instead work with the *expected complete log-likelihood:*

$$\ell(\theta) \equiv \textstyle\sum_{x_U} q(x_U) \log p(x_E, x_U \mid \theta), \tag{15}$$

where $q(x_U)$ is the "averaging distribution." If the averaging distribution is chosen correctly—precisely, if $q(x_U)$ is equal to the distribution of x_U conditioned on the evidence x_E and the parameter vector θ—then the stationary points of the expected log-likelihood (15) are the same as the the stationary points of (14). In other words, optimizing the incomplete log-likelihood or optimizing the expected log-likelihood (with the right averaging distribution) amount to the same solution.

EM can actually be understood, following Neal and Hinton (1998), as coordinate descent on the variational free energy

$$\begin{aligned} F(\theta, q) \equiv &- \textstyle\sum_{x_U} q(x_U) \log p(x_E, x_U \mid \theta) \\ &+ \textstyle\sum_{x_U} q(x_U) \log q(x_U). \end{aligned} \tag{16}$$

It is effectively the Kullback-Leibler divergence (Cover & Thomas, 1991) between the target posterior $p(x_U \mid x_E, \theta)$ and some distribution $q(x_U)$ that approximates the posterior. The term on left is the negative of the expected log-likelihood, and the term on the right is the negative entropy. From (16), the M-step reduces to finding a θ that maximizes the the expected complete log-likelihood, and the E-step reduces to finding a distribution $q(x_U)$ that best matches the posterior $p(x_U \mid x_E, \theta)$; see Heskes et al. (2004). The main difficulty lies in the E-step: direct minimization of F is infeasible due to an intractable entropy. One strategy is to restrict the class of distributions $q(x_U)$ to those that factorize in an analytically convenient fashion (Neal & Hinton, 1998). An alternative strategy is to approximate the intractable entropy by a collection of entropies on small clusters of variables. This yields belief propagation (Yedidia et al., 2005). If we choose these clusters wisely, we will obtain a tractable E-step (see Sec. 3.2), and the approximate M-step may resemble the true maximum likelihood estimator.

3.1 Maximization step

It is difficult to compute the maximum likelihood solution of the noisy-or aggregation factor for sa(X) when it is written as (12), following the standard prescription (Pearl, 1988), because maximization roughly amounts to finding the root of a polynomial. ICL directly provides us with a solution to this conundrum through the atomic choices $\phi_8(X, Y)$ that appear in the aggregation (9). We name these atomic choices is(X, Y), short for "influences to smoke" because Y is counted in X's decision to smoke when is$(X, Y) = \text{true}$. These latent variables act as noisy versions of the aggregated causes; they are generated according to the choice space, and the final aggregation is achieved with a deterministic factor (9). (Note this variable is not symmetric like friendship; is$(X, Y) =$ is(Y, X) does not necessarily hold.) Similarly, in the aggregation for non-smokers

we write $\phi_9(X, Y)$ as influences-not-to-smoke(X, Y), or ins(X, Y) for short. Thus, the atomic choices of importance are is(X, Y), ins(X, Y), and the cycle-resolving latent variables ind(X). These are precisely the unobserved variables x_U. The remaining atomic choices can be ignored because they are easily summed out from the disjoint rules.

Each random variable indexed by i is generated by some CPD $p(x_i \mid x_{\pi[i]})$, where $\pi[i]$ is the set of parents, or predecessors, of node i in the directed graph. We separate the vertices of the directed graph into two sets: 1) the set of variables A that are generated by deterministic noisy-or aggregation factors, namely instances of sa(X) and nsa(X), and 2) the remaining variables B. From the ICL semantics, $p(x_i = \text{true} \mid x_{\pi[i]}) = \theta_k$ and $p(x_i = \text{false} \mid x_{\pi[i]}) = 1 - \theta_k$ for all CPDs that are not aggregation factors. Given the factorization (13), the expected complete log-likelihood works out to be simply

$$\ell(\theta) = \sum_{i \in B} \sum_{x_{vars[i] \cap U}} q(x_{vars[i] \cap U}) \log p(x_i \mid x_{\pi[i]}, \theta) + \text{constant}, \quad (17)$$

where $vars[i]$ is defined to be the intersection of $\pi[i]$ and $\{i\}$, which is precisely all the variables implicated in the ith conditional probability. The deterministic aggregation factors do not matter in the M-step because they are not affected by the choice of θ.

Since each θ_k represents a binomial success rate, we introduce uniform Beta priors

$$p(\theta_k \mid \alpha, \beta) \propto \theta_k^{\alpha-1}(1 - \theta_k)^{\beta-1}, \quad (18)$$

and compute the *maximum a posteriori* solution to the penalized log-likelihood. Taking partial derivatives of the penalized objective and equating them to zero, we obtain roots $\theta_k = a_k/b_k$, where

$$a_k = \alpha - 1 + \sum_{i \in B \cap E} \sum_{x_{\pi[i] \cap U}} \mathbb{I}[k, i, x]\, \delta_{\text{true}}(x_i)\, q(x_{\pi[i] \cap U}) + \sum_{i \in B \cap U} \sum_{x_{\pi[i] \cap U}} \mathbb{I}[k, i, x]\, q(x_i = \text{true}, x_{\pi[i] \cap U}) \quad (19)$$

$$b_k = \alpha + \beta - 2 + \sum_{i \in B} \sum_{x_{vars[i] \cap U}} \mathbb{I}[k, i, x]\, q(x_{vars[i]}), \quad (20)$$

and where the delta-Dirac function $\delta_y(x) = 1$ if and only if $x = y$, and $\mathbb{I}[k, i, x] = 1$ if and only if $p(x_i = \text{true} \mid x_{\pi[i]})$ is a function of θ_k.

3.2 Expectation step

The missing quantities in the M-step are the marginal probabilities $q(x_i = \text{true}, x_{\pi[i] \cap U})$ for every $i \in U$, and the marginal probabilities $q(x_{\pi[i] \cap U})$ for every $i \in E$. We now explain how to estimate these marginals.

The best known tractable solution is to frame the inference problem—the problem of computing the marginals $q(x_{vars[i] \cap U})$ and $q(x_{\pi[i] \cap U})$—as an optimization problem using variational methodology, then to approximate the optimization problem using a *region-based approximation* (Yedidia et al., 2005) so we can compute the marginals efficiently. Let's look at this approximate solution in detail.

Factor graphs. The probability distribution of interest can be described in general terms as a product of non-negative functions $f_C(x_C)$ called *factors*. The probability of the configuration x_U is written as

$$p(x_U \mid x_E, \theta) = \frac{1}{Z} \prod_C f_C(x_C), \quad (21)$$

Each C refers to a subset of U, so that x_C represents the restriction of configuration x_U to the subset C. The normalizing constant Z is designed to ensure that $p(x_U)$ represents a valid probability; the probabilities of all configurations must sum to one.

A *factor graph* is used to express the factorization structure of the probability distribution (Kschischang et al., 2001). A factor graph has two sets of vertices: variable nodes and factor nodes. Ordinarily, variables nodes are drawn as circles and factor nodes are depicted as squares. An edge connects a variable node i to a factor node C if and only if x_i is one of the arguments of the factor ($i \in C$). The symbol C serves two roles: to index a factor as in f_C, and to refer to a collection of variable nodes as in x_C. It is assumed that no two factors are defined on the same subset C.

Each unobserved variable introduces a variable node i to the factor graph, and each CPT $p(x_i \mid x_{\pi[i]})$ introduces a factor node C, which is then linked to the variable nodes in $vars[i] \cap U$. The factor graph of the posterior $p(x_U \mid x_E, \theta)$ is nearly fully connected, as each CPT for friends(X, Y) introduces a factor between ind(X) and ind(Y). Also, each aggregation rule creates a large factor over latent variables sa(X) and is(X, Y) for all Y that are smoking friends of X, or over the variable nsa(X) and

latent causes ins(X, Y). Discovering useful substructure is a daunting task.

The Bethe method. The strategy described here, whose roots lie in the early work of Bethe and Kikuchi in statistical physics, is to approximate the intractable sums in the variational free energy F by a linear combination of more manageable terms F_R. The R represents a "region" or "cluster" of the undirected graphical model, and is a subset of U. Bethe (1935) proposed an approximation to the variational free energy F by forcing the entropy to decompose as a product of entropy terms on the sets C and singleton sets $\{i\}$. This approximation is generally referred to as the *Bethe free energy*. The junction graph method is a natural generalization of the Bethe method in which the large regions (the sets C) and the small regions (the singletons $\{i\}$) can be chosen with greater freedom. A *junction graph* is ordinarily used to formalize these notions; see Aji and McEliece (2001).

A *region graph* is a graph with directed edges and labeled vertices. It generalizes the notion of the junction graph. Each vertex is labeled by a *region* of the target factor graph. A region is defined to be a collection of variable nodes and factor nodes, with the single restriction that if a factor belongs to a region, then all its arguments also belong to the region. We denote a region by the capital letter R. Depending on the context, the symbol R may alternately refer to a collection of variable nodes, a collection of factor nodes, or a node of the region graph. In this manner, we may use x_R to denote the configuration x restricted to the set $R \subseteq U$, we may use the notation $C \in R$ to refer to factors C that are members of region R, and we say that $q_R(x_R)$ denotes the marginal density function defined at region R.

Given a region graph, its corresponding *region free energy* is defined to be

$$\tilde{F}(q) \equiv \sum_R c_R \bar{U}_R - \sum_R c_R H_R, \tag{22}$$

where the average energy and entropy of region R are, respectively,

$$\bar{U}_R = -\sum_{x_R} \sum_{C \in R} q_R(x_R) \log f_C(x_C) \tag{23}$$

$$H_R = -\sum_{x_R} q_R(x_R) \log q_R(x_R). \tag{24}$$

We define $q_R(x_R)$ to be the marginal probability defined on region R, and c_R to be the "counting" number (also called the "overcounting" number) for region R. If the counting numbers are well-chosen, then decomposition of the average energy is exact.

Yedidia et al. (2005) give a recipe for coming up with reasonable counting numbers c_R for a given region graph. A good choice of numbers c_R ensures that we only count the contribution of each subset once in $\tilde{F}$. This insight is the basis for the *cluster variation method.* The observation made by McEliece and Yildirim (2002) is that this recipe is connected to results in combinatorial mathematics and, in particular, the theory of partially ordered sets. By introducing a *partial ordering* on the regions, we can treat the collection of regions as a partially ordered set, or *poset*, where the partial ordering is the set inclusion relation, and we can then draw the regions as vertices in a Hasse diagram. Since we have described the regions R as elements of a poset, we can frame the choice of counting numbers as a counting problem on the poset, and use the principle of inclusion and exclusion for partially ordered sets (otherwise the *Möbius inversion principle*) to come up with the answer (Bogart, 1990). Fortunately, the region graph construction won't be quite this complicated for the factor graph induced by the social network model when all the smoking habits and friendship relations are observed.

Suppose we define two sets of regions. The regions in the first set correspond to the maximal subsets C, and their counting numbers are set to 1. The regions in the second set correspond to the singletons $\{i\}$. We set their counting numbers to be equal to $1 - d_i$, where the degree d_i of the ith variable node is defined to be the number of neighbouring factor nodes in the factor graph, or the number of factors with which the ith variable participates. Provided all friends(X, Y) and smokes(X) are observed, a region graph defined in this way ensures that the average energy is exact, and that the contribution of every subset of variable nodes is only counted once in $\tilde{F}$. This is so because: 1) every non-empty intersection of two regions is a member of the region graph, and 2) the counting numbers are equivalent to those obtained as a solution to the Möbius inversion principle. This particular region-based approximation is equivalent to the Bethe approximation.

Expanding and simplifying (22), the Bethe approxi-

mation to the variational free energy is given by

$$\begin{aligned}\tilde{F}(q) = &-\sum_C \sum_{x_C} q_C(x_C) \log f_C(x_C) \\ &+ \sum_C \sum_{x_C} q_C(x_C) \log q_C(x_C) \\ &+ \sum_i (1-d_i) \sum_{x_i} q_i(x_i) \log q_i(x_i), \end{aligned} \tag{25}$$

where $q_i(x_i)$ and $q_C(x_C)$ are the *pseudo-marginals* defined on the variable and factor nodes of the factor graph.

Solution to the Bethe method. The object is now to come up with marginals $q_C(x_C)$ and $q_i(x_i)$ that minimize the approximate variational free energy (25). The immediate form of the objective appears to be problematic because it could involve a summation over a large number of configurations x_C when $f_C(x_C)$ is an aggregation factor. We will address this concern shortly.

The optimization problem is to minimize $\tilde{F}(q)$ subject to three types of constraints: 1) the pseudo-marginals must be non-negative, 2) they must sum to one, and 3) the pseudo-marginals on neighbouring regions should agree. Thus, the constrained, nonconvex program is to minimize $\tilde{F}$ subject to non-negativity constraints

$$q_C(x_C) \geq 0 \quad \text{and} \quad q_i(x_i) \geq 0, \tag{26}$$

normalization constraints

$$\textstyle\sum_{x_C} q_C(x_C) = 1 \quad \text{and} \quad \sum_{x_i} q_i(x_i) = 1, \tag{27}$$

and consistency constraints

$$\textstyle\sum_{x_{C\setminus\{i\}}} q_C(x_C) = q_i(x_i), \tag{28}$$

for every factor node C, for every neighbouring variable node $i \in C$, and then again for every configuration x_i.

The standard course of action is to use results in duality to locate solutions. This leads to the familiar sum-product updates (Yedidia et al., 2005). The Lagrangian function for the constrained optimization problem is

$$\begin{aligned}\tilde{L}(q,\gamma,\lambda) = \tilde{F}(q) &+ \sum_C \gamma_C \big\{ \textstyle\sum_{x_C} q_C(x_C) - 1 \big\} \\ &+ \sum_C \sum_{i \in C} \sum_{x_i} \lambda_{C,i}(x_i) \big\{ q_i(x_i) - \textstyle\sum_{x_{C\setminus\{i\}}} q_C(x_C) \big\} \\ &+ \sum_i \gamma_i \big\{ \textstyle\sum_{x_i} q_i(x_i) - 1 \big\}, \end{aligned} \tag{29}$$

where the γ_i and γ_C are the Lagrange multipliers associated with the normalization constraints, and the $\lambda_{C,i}(x_i)$ are the Lagrange multipliers for the consistency constraints. It is assumed that all the probabilities are strictly positive so that the Lagrange multipliers associated with the non-negativity constraints vanish. For a candidate point to be optimal, the gradient of the Lagrangian with respect to the primal variables must vanish. The partial derivatives of the Lagrangian (29) with respect to the primal variables are given by

$$\begin{aligned}\frac{\partial \tilde{L}}{\partial q_i(x_i)} &= (1-d_i)(1+\log q_i(x_i)) + \gamma_i \\ &\quad + \textstyle\sum_{C \in N(i)} \lambda_{C,i}(x_i) \end{aligned} \tag{30}$$

$$\begin{aligned}\frac{\partial \tilde{L}}{\partial q_C(x_C)} &= 1 + \log q_C(x_C) - \textstyle\sum_C \log f_C(x_C) + \gamma_C \\ &\quad - \textstyle\sum_{i \in C} \lambda_{C,i}(x_i), \end{aligned} \tag{31}$$

where $N(i)$ is the set of factor nodes adjacent to the ith variable node in the factor graph. We recover the coordinate ascent equations by equating the partial derivatives to zero and solving for $q_C(x_C)$ and $q_i(x_i)$:

$$q_i(x_i) \propto \textstyle\prod_{C \in N(i)} (\exp \lambda_{C,i}(x_i))^{\frac{1}{d_i - 1}} \tag{32}$$

$$q_C(x_C) \propto f_C(x_C) \textstyle\prod_{i \in C} \exp \lambda_{C,i}(x_i), \tag{33}$$

Next, by making the substitutions

$$\lambda_{C,i}(x_i) = \log m_{i \to C}(x_i) \tag{34}$$

$$m_{i \to C}(x_i) = \textstyle\prod_{C' \in N(i) \setminus \{C\}} m_{C' \to i}(x_i), \tag{35}$$

the expressions for the marginals become

$$q_i(x_i) \propto \textstyle\prod_{C \in N(i)} m_{C \to i}(x_i) \tag{36}$$

$$q_C(x_C) \propto f_C(x_C) \textstyle\prod_{i \in C} m_{i \to C}(x_i), \tag{37}$$

which give us the familiar expressions for the marginal beliefs. The message update from variable node i to factor node C is given in (35), so the remaining piece of the puzzle is the update equation for a message passed from C to i. Starting from (28), then plugging (36) and (37) into this identity, we obtain the sum-product rule

$$m_{C \to i}(x_i) \propto \sum_{x_{C\setminus\{i\}}} f_C(x_C) \prod_{j \in C \setminus \{i\}} m_{j \to C}(x_j). \tag{38}$$

In summary, the sum-product message updates represent descent directions of the Bethe free energy (25) subject

to the constraint that the pseudo-marginals remain locally consistent. There is some concern that these updates will oscillate indefinitely, so we implemented an E-step that is guaranteed to converge by iteratively solving a convex relaxation of $\tilde{F}$ (Heskes, 2006). In the experiments (Sec. 4), we compared the quality of the solutions obtained from both convergent and non-convergent implementations of the E-step.

The Bethe approximation does not immediately lead to a tractable message-passing algorithm because we still have to deal with a potentially monstrous summation for any sum-product message sent from an aggregation factor. What we have is one of the simplest examples of a *causally independent* factor (Zhang & Poole, 1996), and this fact guarantees us an efficient way to compute the summation.

Causal independence. Zhang and Poole (1996) define causal independence as follows. Causal variables $x \equiv \{x_1, \ldots, x_n\}$ are causally independent with respect to aggregate variable e if there exists a commutative, associative binary operator $*$, a collection of random variables $\xi \equiv \{\xi_1, \ldots, \xi_n\}$ with the same set of realizations as x, and a probability density $p(\xi \,|\, x)$ such that

1. $e = \xi_1 * \cdots * \xi_n$
2. $p(\xi_i \,|\, \xi_{-i}, x) = p(\xi_i \,|\, x_i)$,

where ξ_{-i} is defined to be the collection of all the introduced random variables except for ξ_i. A simple but useful result of causal independence is that the probability of e given x can be written as

$$p(e \,|\, x) = \textstyle\sum_{\xi} p(\xi_1 \,|\, x_1) \cdots p(\xi_n \,|\, x_n), \qquad (39)$$

where the summation is over all realizations ξ such that $e = \xi_1 * \cdots * \xi_n$.

The definition of causal independence extends with little extra effort to factors: a causally independent factor $f(e, x)$ would be described as an arithmetic decomposition on factors $f_i(\xi_i, x_i)$. We can then show that this notion applies directly to the sum-product update (38), in which one of the random variables involved in the message update is the aggregate variable $\mathrm{sa}(X)$ or $\mathrm{nsa}(X)$, and the remaining random variables are the causes $\mathrm{is}(X, Y)$ or $\mathrm{ins}(X, Y)$. A similar observation can be used to derive efficient message-passing updates for probabilistic decoding of low-density parity check codes; see Moon (2005).

To derive the efficient message update (38) for the case when $f_C(x_C)$ is a noisy-or factor, we need to consider two cases. In the first case, x_i is an aggregation variable. Rewriting the sum-product message update as

$$m_{C \to i}(x_i) \propto \sum_{x_{C \setminus \{i\}}} f_C(x_C) \prod_{j \in C} g_j(x_j), \qquad (40)$$

then the message for $x_i = \mathrm{false}$ is derived to be

$$m_{C \to i}(\mathrm{f}) \propto \textstyle\prod_{j \in C} g_j(\mathrm{f}), \qquad (41)$$

and the message for $x_i = \mathrm{true}$ is proportional to

$$\begin{aligned} m_{C \to i}(\mathrm{t}) \propto\ & g_i(\mathrm{t}) \textstyle\prod_{j \in C \setminus \{i\}} \sum_{x_j} g_j(x_j) \\ & - g_i(\mathrm{t}) \textstyle\prod_{j \in C \setminus \{i\}} g_j(\mathrm{f}), \end{aligned} \qquad (42)$$

where t stands for true and f stands for false. In the second case, $x_{i'}$ is one of the causes (x_i is the aggregate variable). The message sent to variable node i' works out to be

$$\begin{aligned} m_{C \to i'}(\mathrm{f}) =\ & (g_i(\mathrm{f}) - g_i(\mathrm{t})) \textstyle\prod_{j \in C \setminus \{i\}} g_j(\mathrm{f}) \\ & + g_i(\mathrm{t})\, g_{i'}(\mathrm{f}) \textstyle\prod_{j \in C \setminus \{i, i'\}} \sum_{x_j} g_j(x_j). \end{aligned} \qquad (43)$$

$$m_{C \to i'}(\mathrm{t}) = g_i(\mathrm{t})\, g_{i'}(\mathrm{t}) \textstyle\prod_{j \in C \setminus \{i, i'\}} \sum_{x_j} g_j(x_j). \qquad (44)$$

The inference strategy we have outlined in this section is not necessarily appropriate for making predictions about a social network when arbitrary friendships and smoking habits are unknown. When we only need to make a single prediction $\mathrm{smokes}(X)$ or $\mathrm{friends}(X, Y)$, however, a straightforward way to obtain a prediction is to estimate the Bayes factor (Kass & Raftery, 1995) from $\tilde{F}$ for two cases, when the query variable is true and when it is false.

What we have described does not strictly adhere to Bayesian principles, because we do not adjust the model to reflect evidence obtained after the training phase, and because we replace the integral over the model parameters θ with a single mode. However, this is standard practice for learning in graphical models.

4 Experiments

We ran three experiments to assess the behaviour of the proposed network model. For the first two, we used data

generated from artificial processes. The third experiment comprised an actual social network analysis of smoking in grade school adolescents.

4.1 Experimental setup

In all the experiments, we trained our model with two versions of EM following the description of Sec. 3: one with a non-convergent E-step ("loopy" belief propagation), and another with an E-step based on a convergent message passing algorithm. The only real parameter to adjust was the Beta prior on the model parameters θ_k. We chose a weak, uniform prior $\alpha_k = 4, \beta_k = 4$.

We compared the performance of our model to an undirected probabilistic graphical model represented as a Markov logic network, or MLN (Richardson & Domingos, 2006). We used the software Alchemy (Kok et al., 2009) to learn weighted formulae of the form

$$\texttt{Smokes}(x) \wedge \texttt{Friends}(x, y) \Rightarrow \texttt{Smokes}(y) \qquad (45)$$

for various non-redundant combinations of its atoms, and $\texttt{Friends}(x, y) \Rightarrow \texttt{Friends}(y, x)$ to enforce symmetry of friendship.[3] We tried more complex models that had more rules such as reflexivity and transitivity of friendship, but they offered no advantage. Alchemy implements the pseudo-likelihood approximation for learning, and includes a specialized satisfiability solver MC-SAT for inferring queries.

In one of the experiments, we compared to special cases of our model when $p(\text{ind}(X) = \text{true}) = 0$ (called the "independent friendship" model since all decisions regarding friendship are unaffected by smoking habits), and the "independent smokers" model when $p(\text{ind}(X) = \text{true}) = 1$.[4]

Part 1. In the first set of experiments, we generated artificial social networks from our directed model with pre-specified model parameters. The control variable was the prior on $\text{ind}(X) = \text{true}$, which we varied from $1/10$ (most friendships are generated randomly) to $9/10$ (most smoking habits are generated randomly) in intervals of $1/10$. Such an experiment may appear to be unfair, but it was unexpectedly challenging, probably due to the difficulty of recovering the latent behaviour of individuals. For each $p(\text{ind}(X) = \text{true})$, we ran 16 independent trials, and in each trial we generated training and test sets, each containing 8 isolated social networks with populations of size $n = 50$.

Part 2. In the second set of experiments, we generated data from a temporal process that bore little resemblance to the simple model we propose in this paper. In our simulation, individuals were occasionally pressured to change their smoking habits, they started or stopped smoking due to external factors, formed new friendships either by chance encounters or through mutual friends, or stopped being friends, sometimes because of friends' smoking habits. At any time step, an individual X might begin or stop smoking, depending on whether or not X's friends smoke. Furthermore, the structure of the network evolved over time: at any time step $\text{friends}(X, Y)$ may become true with some probability that depends on whether X and Y smoke. The likelihood of these interactions depended on proximity according to a latent location. Precisely, individuals were sampled at geographic locations, and people that lived close by were more likely to become friends than those that lived far away. Since none of the tested models possessed such details, we did not expect them to perform well. We ran three experiments with populations of size $n = 20$, 100 and 200. Each training and test set consisted of 5 separate populations. The data sets exhibited considerable variance in the number of friendships and smokers.

Part 3. In the final experiment, we learned a social network model from a year-long longitudinal study of smoking and drug use in a cohort of $n = 150$ teenagers attending a school in Scotland (Pearson & Michell, 2000). It is purported to be the first scientific study in the UK to adopt social network methodology for analyzing smoking and drug-taking behaviour. The authors only recorded reciprocal friendship links. They gathered other information, such as gender, and used this information to assess the strength of links. This information would have surely improved the quality of predictions (e.g. girls tended to be friends with girls, smoking was less prevalent among boys due to perception that it affects performance in sports). We trained the models on the data collected when the students were in grade 2, and validated the models on the survey data from a

[3] In practice, we found that the number of smoking rules of the form (45) had a significant impact on accuracy, so we always tried to pick a set that worked well.

[4] We also remove $\text{ind}(Y)$ from (9,10) in the former model.

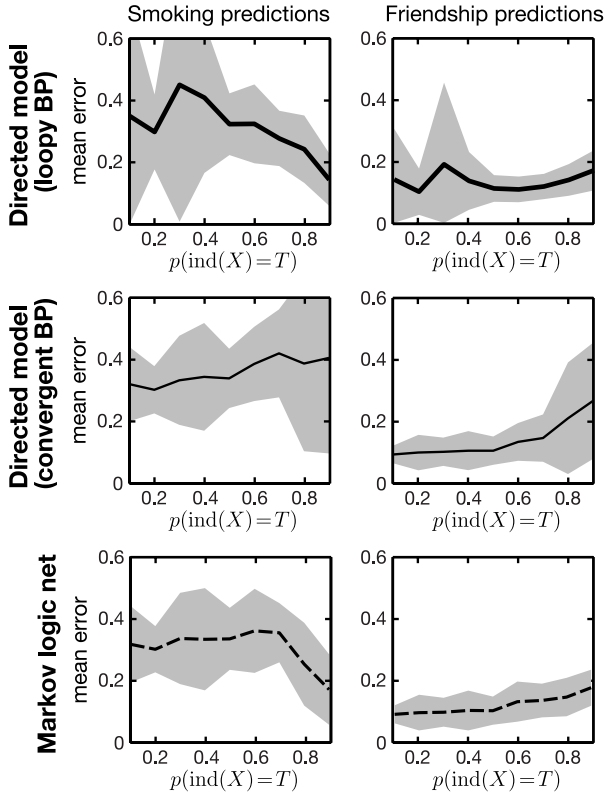

Figure 2: Average smoking and friendship prediction error from the directed model (trained with a loopy and convergent E-step) and the MLN for varying proportions of $\mathrm{ind}(X) = \mathrm{true}$, over 16 independent trials.

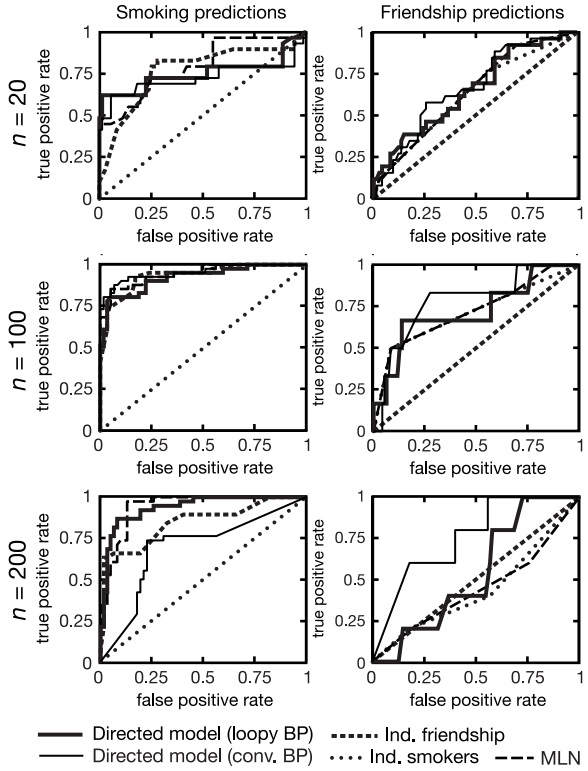

Figure 3: ROC curves for smoking and friendship predictions on test sets from the artificial temporal process.

year later in grade 3. These social networks were very sparsely connected and highly transitory, hence none of approaches tested here were able to to learn a useful model of friendship.

4.2 Experiment results

Part 1. The results of the first experiment are shown in Fig. 2. For each model, a single test consisted of computing the maximum a posteriori estimate of $\mathrm{smokes}(X)$ or $\mathrm{friends}(X, Y)$ for a particular individual X or pair of individuals (X, Y) given information regarding the habits and friendships of the remaining portion of the testing network. Fig. 2 was then obtained by taking the mean error of these tests. The shaded region is the 90% confidence interval. As expected, the accuracy of the MLN *(bottom)* and the directed model with non-convergent, loopy belief propagation *(top)* got better as more and more individuals were not influenced by their peers, but what is surprising is that the performance of the directed model with convergent belief propagation *(middle)* did not improve, and even degraded slightly—we currently have no explanation for this behaviour. The loopy implementation *(top)* was not completely satisfactory either as its performance varied considerably in networks with few $\mathrm{ind}(X) = T$. At the right-most end of the spectrum, when $\mathrm{ind}(X) = T$ for most individuals X, it is still possible for the model to make useful inferences about smoking habits by conditioning on observations about friendship. Within the confidence intervals, we obtained about the same level of performance for the directed and undirected models, barring the unexpected effects of an

approximate E-step.

Part 2. Results of the second set of simulations are shown as receiver operating characteristic (ROC) curves in Fig. 3. Tests were done in the same manner as before: for each test, we left out one smoking or friendship observation. Unsurprisingly, these simple social network models did not quite capture the complexity of the artificial process, particularly in predicting friendships. We did not observe a degradation in the performance of the convergent implementation like we did in the first experiment, although it is interesting to note that it did much better at predicting friendships in the large ($n = 200$) network at the expense of poor prediction of smoking habits. As expected, the "independent smokers" and "independent friendship" models did no better than the worst possible (*i.e.* a straight line) at predicting, respectively, smokes(X) and friends(X, Y). It is significant that the directed, contingently acyclic model: 1) outperformed these two simple relational models on both predictions of smoking and friendship, and 2) tended to make predictions about as accurately as the Markov logic network model.

Part 3. Finally, we examine the results from the adolescent smoking and drug use study in Fig. 4. Overall, we observe trends similar to our previous experiments on synthetic data. The MLN displayed some advantage in accuracy of smoking habits, but did worse in predicting friendships. Friendship predictions were globally poor, as we forewarned. These results do nonetheless clearly suggest that more detailed expert knowledge must be inputed into the model to obtain useful scientific inferences.

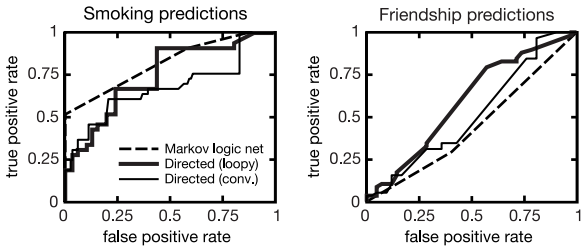

Figure 4: ROC curves for smoking and friendship predictions on the adolescent social network test data.

5 Conclusions

Contrary to common practice, we developed a directed graphical model for social networks and an approximate EM algorithm for training the model. Our experiments on both synthetic and actual data of friendships and smoking habits showed that a directed model can predict interdependencies equally as well as a similarly expressive undirected model in a simple but challenging social network domain. Our experiments also highlight the need for more work into robust convergent message passing algorithms for belief propagation.

There are many open research questions in extending our ideas to larger and more challenging social network domains. One important open question is how to design directed graphical representations of social networks that transfer to populations of different sizes. In so doing, one could learn the model parameters from a small network for which data has been collected, and use it to make predictions in much larger social networks. Another unresolved problem is how to efficiently handle queries with arbitrary sets of observations in large social networks—it is far from clear how to exploit such model structure for conducting inference at a first-order level (Poole, 2003), and for developing approximate sum-product message passing algorithms.

References

Aji, S. M., & McEliece, R. J. (2001). The generalized distributive law and free energy minimization. *Proceedings of the 39th Allerton Conference* (pp. 672–681).

Besag, J. (1974). Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society, Series B*, *36*, 192–236.

Bethe, H. A. (1935). Statistical theory of superlattices. *Proceedings of the Royal Society of London*, *150*, 552–575.

Bogart, K. P. (1990). *Introductory combinatorics*. Academic Press. 2nd edition.

Boutilier, C., Friedman, N., Goldszmidt, M., & Koller, D. (1996). Context-specific independence in Bayesian networks. *Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence* (pp. 115–123).

Carrington, P. J., Scott, J., & Wasserman, S. (Eds.). (2005). *Models and methods in social network analysis*. Cambridge University Press.

Cover, T. M., & Thomas, J. A. (1991). *Elements of information theory*. Wiley.

Frank, O., & Strauss, D. (1986). Markov graphs. *Journal of the American Statistical Association*, *81*, 832–842.

Geiger, D., & Heckerman, D. (1996). Knowledge representation and inference in similarity networks and Bayesian multinets. *Artificial Intelligence*, *82*, 45–74.

Heskes, T. (2006). Convexity arguments for efficient minimization of the Bethe and Kikuchi free energies. *Journal of Artificial Intelligence Research*, *26*, 153–190.

Heskes, T., Zoeter, O., & Wiegerinck, W. (2004). Approximate expectation maximization. In *Advances in neural information processing systems*, vol. 16, 353–360. MIT Press.

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*, *14*.

Hunter, D. R., Goodreau, S. M., & Handcock, M. S. (2008). Goodness of fit of social network models. *Journal of the American Statistical Association*, *103*, 248–248.

Kass, R. E., & Raftery, A. E. (1995). Bayes factors. *Journal of the American Statistical Association*, *90*, 773–795.

Kok, S., Sumner, M., Richardson, M., Singla, P., Poon, H., Lowd, D., Wang, J., & Domingos, P. (2009). *The Alchemy system for statistical relational AI* (Technical Report). University of Washington.

Kschischang, F. R., Frey, B. J., & Loeliger, H.-A. (2001). Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, *47*, 498–519.

Marlin, B. M. (2008). *Missing data problems in machine learning*. Doctoral dissertation, University of Toronto.

McEliece, R. J., & Yildirim, M. (2002). Belief propagation on partially ordered sets. In D. Gilliam and J. Rosenthal (Eds.), *Mathematical systems theory in biology, communications, computation and finance*, 275–299. Springer.

Milch, B. (2006). *Probabilistic models with unknown objects*. Doctoral dissertation, University of California, Berkeley.

Milch, B., Marthi, B., Russell, S., Sontag, D., Ong, D. L., & Kolobov, A. (2005). BLOG: Probabilistic models with unknown objects. *Proceedings of the 19th International Joint Conference on Artificial Intelligence* (pp. 1352–1359).

Moon, T. K. (2005). *Error correction coding: mathematical methods and algorithms*. Wiley-Interscience.

Muñoz, V., & Eaton, W. A. (1999). A simple model for calculating the kinetics of protein folding from three-dimensional structures. *Proc. of the National Academy of Sciences*, *96*.

Neal, R., & Hinton, G. (1998). A view of the EM algorithm that that justifiies incremental, sparse, and other variants. In M. I. Jordan (Ed.), *Learning in graphical models*, 355–368. Kluwer Academic.

Neville, J., & Jensen, D. (2007). Relational dependency networks. In L. Getoor and B. Taskar (Eds.), *Introduction to statistical relational learning*, 239–268. MIT Press.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann.

Pearson, M., & Michell, L. (2000). Smoke Rings: social network analysis of friendship groups, smoking and drug-taking. *Drugs: education, prevention and policy*, *7*, 21–37.

Poole, D. (1997). The Independent Choice Logic for modelling multiple agents under uncertainty. *Artificial Intelligence*, *94*.

Poole, D. (2000). Abducting through negation as failure: Stable models with the Independent Choice Logic. *Journal of Logic Programming*, *44*, 5–35.

Poole, D. (2003). First-order probabilistic inference. *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, 985–991.

Richardson, M., & Domingos, P. (2006). Markov logic networks. *Machine Learning*, *62*, 107–136.

Salakhutdinov, R., Mnih, A., & Hinton, G. (2007). Restricted Boltzmann machines for collaborative filtering. *Proc. of the 24th Intl. Conf. on Machine Learning* (pp. 791–798).

Shachter, R. D. (1998). Bayes-Ball: The rational pastime (for determining irrelevance and requisite information in belief networks and influence diagrams). *Proc. of the 14th Conf. on Uncertainty in Artificial Intelligence* (pp. 480–487).

Snijders, T. A. B. (2002). Markov chain Monte Carlo estimation of exponential random graph models. *Journal of Social Structure*, *3*.

Spiegelhalter, D. J., Dawid, A. P., Lauritzen, S. L., & Cowell, R. G. (1993). Bayesian analysis in expert systems. *Statistical Science*, *8*, 219–247.

Sun, D., Roth, S., Lewis, J., & Black, M. J. (2008). Learning optical flow. *Proceedings of the 10th European Conference on Computer Vision*.

Taskar, B., Abbeel, P., & Koller, D. (2002). Discriminative probabilistic models for relational data. *Proc. of the 18th Conf. on Uncertainty in Artificial Intelligence* (pp. 485–492).

Taskar, B., Wong, M.-F., Abbeel, P., & Koller, D. (2004). Link prediction in relational data. In *Advances in neural information processing systems*, vol. 16, 659–666. MIT Press.

Yedidia, J. S., Freeman, W. T., & Weiss, Y. (2005). Constructing free-energy approximations and generalized belief propagation algorithms. *IEEE Transactions on Information Theory*, *51*, 2282–2312.

Younes, L. (1991). Stochastic gradient estimation strategies for Markov random fields. *Proceedings of the Spatial Statistics and Imaging Conference*.

Zhang, N. L., & Poole, D. (1996). Exploiting causal independence in Bayesian network inference. *Journal of Artificial Intelligence Research*, *5*, 263–313.