CHAIN EVENT GRAPH MAP MODEL SELECTION

Peter A. Thwaites, Guy Freeman and Jim Q. Smith
Department of Statistics, University of Warwick, Coventry, U.K.

Keywords: Bayesian network, Chain event graph, Conjugate learning, Maximum a posteriori model.

Abstract: When looking for general structure from a finite discrete data set one can search over the class of Bayesian Networks (BNs). The class of Chain Event Graph (CEG) models is however much more expressive and is particularly suited to depicting hypotheses about how situations might unfold. Like the BN, the CEG admits conjugate learning on its conditional probability parameters using product Dirichlet priors. The Bayes Factors associated with different CEG models can therefore be calculated in an explicit closed form, which means that search for the maximum a posteriori (MAP) model in this class can be enacted by evaluating the score function of successive models and optimizing. Local search algorithms can be devised for the class of candidate models, but in this paper we concentrate on the process of scoring the members of this class.

1 INTRODUCTION

The Chain Event Graph (CEG), introduced in (Smith and Anderson, 2008; Thwaites et al., 2008; Smith et al., 2009), is a graphical model specifically designed to represent an analyst's knowledge of the structure of problems whose state spaces are highly asymmetric and do not admit a natural product structure. There are many scenarios in medicine, biology and education where such asymmetries arise naturally, and where the main features of the model class cannot be fully captured by a single BN or even a context specific BN. A key property of the CEG framework is that these graphical models are *qualitative* in their topologies – they encode sets of conditional independence statements about how things might happen, without prespecifying the probabilities associated with these events. Each CEG model can therefore be identified with a unique explanation of how situations might unfold.

The CEG is an event-based (rather than variable-based) graphical model, and is a function of an event tree. Any problem on a finite discrete data set can be modelled using an event tree, but they are particularly suited to problems with asymmetric state spaces. Unfortunately, it is almost impossible to read the conditional independence properties of a model from an event tree representation, as only trivial independencies are expressed within its topology. The CEG elegantly solves this problem, encoding a rich class of conditional independence statements through its edge and vertex structure.

So consider an event tree T with vertex set $V(T)$, directed edge set $E(T)$, and $S(T) \subset V(T)$, the set of the tree's non-leaf vertices or *situations* (Shafer, 1996)). A probability tree can then be specified by a transition matrix on $V(T)$, where absorbing states correspond to leaf-vertices. Transition probabilities are zero except for transitions to a situation's children (see Table 1).

Let $T(v)$ be the subtree rooted in the situation v which contains all vertices after v in T. We say that v_1 and v_2 are in the same *position* if:

- the trees $T(v_1)$ and $T(v_2)$ are topologically identical,
- there is a map between $T(v_1)$ and $T(v_2)$ such that the edges in $T(v_2)$ are labelled, under this map, by the same probabilities as the corresponding edges in $T(v_1)$.

Table 1: Part of the transition matrix for Example 1.

	v_1	v_2	v_3	v_4	v_5	v_6	$\dots$	v_∞^1	v_∞^2	$\dots$
v_0	θ_1	θ_2	θ_3	0	0	0	$\dots$	0	0	$\dots$
v_1	0	0	0	θ_5	0	0	$\dots$	θ_4	0	$\dots$
v_2	0	0	0	0	θ_4	θ_5	$\dots$	0	0	$\dots$
$\vdots$	$\vdots$					$\vdots$		$\vdots$	$\vdots$	

Thwaites P., Freeman G. and Smith J. (2009).
CHAIN EVENT GRAPH MAP MODEL SELECTION.
In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development*, pages 392-395
DOI: 10.5220/0002292403920395

The set $W(T)$ of positions w partitions $S(T)$. The *transporter* CEG (Thwaites et al., 2008) is a directed graph with vertices $W(T) \cup \{w_\infty\}$, with an an edge e from w_1 to $w_2 \neq w_\infty$ for each situation $v_2 \in w_2$ which is a child of a fixed representative $v_1 \in w_1$ for some $v_1 \in S(T)$, and an edge from w_1 to w_∞ for each leaf-node $v \in V(T)$ which is a child of some fixed representative $v_1 \in w_1$ for some $v_1 \in S(T)$.

For the position w in our transporter CEG, we define the *floret* $F(w)$ to be w together with the set of outgoing edges from w. We say that w_1 and w_2 are in the same *stage* if:

- the florets $F(w_1)$ and $F(w_2)$ are topologically identical,
- there is a map between $F(w_1)$ and $F(w_2)$ such that the edges in $F(w_2)$ are labelled, under this map, by the same probabilities as the corresponding edges in $F(w_1)$.

The CEG $C(T)$ is then a mixed graph with vertex set $W(C)$ equal to the vertex set of the transporter CEG, directed edge set $E_d(C)$ equal to the edge set of the transporter CEG, and undirected edge set $E_u(C)$ consisting of edges which connect the component positions of each stage $u \in U(C)$, the set of stages. The CEG-construction process is illustrated in Example 1, and an example CEG in Figure 2.

Example 1

Consider the tree in Figure 1 which has 11 atoms (root-to-leaf paths). Symmetries in the tree allow us to store the distribution in 5 conditional tables which contain 11 (6 free) probabilities. The transporter CEG is produced by combining the vertices $\{v_4, v_5, v_7\}$ into one position w_4, the vertices $\{v_6, v_8\}$ into one position w_5, and all leaf-nodes into a single sink-node w_∞. The CEG C (Figure 2) has an undirected edge connecting the positions w_1 and w_2 as these lie in the same stage – their florets are topologically identical, and the edges of these florets carry the same probabilities.

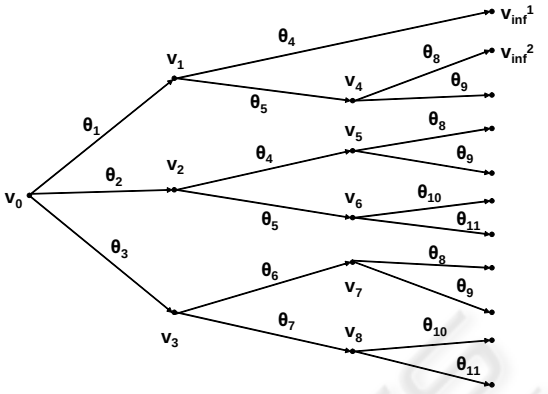

Figure 1: Tree for Example 1.

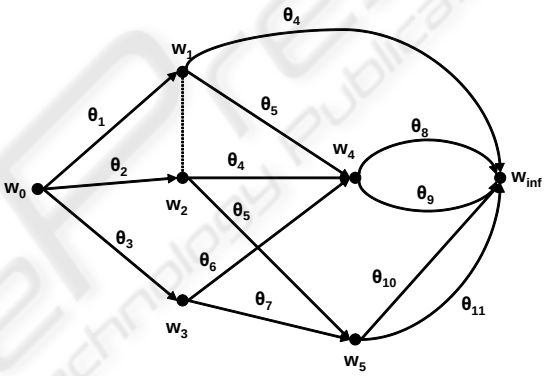

Figure 2: CEG for Example 1.

From our CEG definition, if $w_1, w_2 \in u$ for some u, then the corresponding edges in the florets $F(w_1)$ and $F(w_2)$ carry the same probabilities. So, for each member u of the set of stages prescribed by the model under consideration for our CEG, we can label the edges leaving u by their probabilities under this model. We can then let x_{un} be the **total** number of sample units passing through an edge labelled π_{un}; and the likelihood $L(\boldsymbol{\pi})$ for our CEG model is given by

$$L(\boldsymbol{\pi}) = \prod_u \prod_n \pi_{un}^{x_{un}}$$

For BNs, the assumptions of local and global independence, and the use of Dirichlet priors ensures conjugacy. The analogue for CEGs is to give the vectors of probabilities associated with the stages independent Dirichlet distributions. Then the structure of the likelihood $L(\boldsymbol{\pi})$ results in prior and posterior distributions for the CEG model which are products of Dirichlet densities. The result of this conjugacy is

2 LEARNING CEGs

As the CEG can express a richer class of conditional independence structures than the BN, CEG model selection allows for the automatic identification of more subtle features of the data generating process than it would be possible to express (and therefore to evaluate) through the class of BNs. In this section we introduce the techniques for learning CEGs and compare them with those for learning BNs.

that the marginal likelihood of each CEG is therefore the product of the marginal likelihoods of its component florets. Explicitly, the marginal likelihood of a CEG C is

$$\prod_u \frac{\Gamma(\sum_n \alpha_{un})}{\Gamma(\sum_n(\alpha_{un}+x_{un}))} \prod_n \frac{\Gamma(\alpha_{un}+x_{un})}{\Gamma(\alpha_{un})}$$

where, as above

- u indexes the stages of C
- n indexes the outgoing edges of each stage
- α_{un} are the exponents of our Dirichlet priors
- x_{un} are the data counts

As we are actually interested in $p(model \mid data)$, and this is proportional to $p(data \mid model) \times p(model)$, we need to set both parameter priors and prior probabilities for the possible models.

Exactly analogously with BNs, parameter modularity in CEGs implies that whenever CEG models share some aspect of their topology, we assign this aspect the same prior distribution in each model. When such priors reflect our beliefs in a given context, this can reduce our problem dramatically to one of simply expressing prior beliefs about the possible floret distributions (ie. the local differences in model structure). As each CEG model is essentially a partition of the vertices in the underlying tree into sets of stages, this requirement ensures that when two partitions differ only in whether or not some subset of vertices belong to the same stage, the prior expressions for the models differ only in the term relating to this stage. The separation of the likelihood means that this local difference property is retained in the posterior distribution.

Now, our candidate set is much richer than the corresponding candidate BN set, and will probably contain models we have not previously considered in our analysis. Again, evoking modularity, if we have no information to suggest otherwise, we follow standard BN practice and let $p(model)$ be constant for all models in the class of CEGs. We now use the logarithm of the marginal likelihood of a CEG model as its score, and maximise this score over our set of candidate models to find the MAP model.

Our expression has the nice property that the difference in score between two models which are identical except for a particular subset of florets, is a function of the subscores only of the probability tables on the florets where they differ. Various fast deterministic and stochastic algorithms can therefore be derived to search over the model space, even when this is large – see (Freeman and Smith, 2009). This property is of course shared by the class of BNs.

We set the priors of the hyperparameters so that they correspond to counts of dummy units through the graph. This can be done by setting a Dirichlet distribution on the root-to-sink paths, and for simplicity we choose a uniform distribution for this. It is then easy to check that in the special case where the CEG is expressible as a BN, the CEG score above is equal to the standard score for a BN using the usual prior settings as recommended in, for example, (Cooper and Herskovits, 1992; Heckerman et al., 1995). As a comparison with our CEG-expression; given Dirichlet priors and a multivariate likelihood, the marginal likelihood on a BN is expressible as

$$\prod_{i \in V} \left[\prod_m \frac{\Gamma(\sum_n \alpha_{imn})}{\Gamma(\sum_n(\alpha_{imn}+x_{imn}))} \prod_n \frac{\Gamma(\alpha_{imn}+x_{imn})}{\Gamma(\alpha_{imn})} \right]$$

where

- i indexes the set of variables of the BN
- n indexes the levels of the variable X_i
- m indexes vectors of levels of the parental variables of X_i

The importance of this result is that were we first to search the space of BNs for the MAP model, then we could seamlessly refine this model using the CEG search score described above. Such embellishments will allow us to search over models containing significant amounts of context specific information. Furthermore any model we find will have an associated interpretation which can be stated in common language, and can be discussed and critiqued by our client/expert for its phenomenological plausibility.

Example 1 Continued

For the CEG in Figure 2, we put a uniform prior over the 11 root-to-leaf paths, which in turn allows us to assign our stage priors as follows: we assign a $Di(3,4,4)$ prior to the stage identified by w_0, a $Di(3,4)$ prior to the stage $u_1 \equiv (w_1, w_2)$, a $Di(2,2)$ prior to each of the stages identified by w_3 and w_5, and a $Di(3,3)$ prior to the stage identified by w_4. We would then have a marginal likelihood of

$$\frac{\Gamma(11)}{\Gamma(11+N)}\frac{\Gamma(3+x_{01})\Gamma(4+x_{02})\Gamma(4+x_{03})}{\Gamma(3)\Gamma(4)\Gamma(4)}$$
$$\times\frac{\Gamma(7)}{\Gamma(7+x_{01}+x_{02})}\frac{\Gamma(3+x_{14}+x_{24})\Gamma(4+x_{15}+x_{25})}{\Gamma(3)\Gamma(4)}$$
$$\times\frac{\Gamma(4)}{\Gamma(4+x_{03})}\frac{\Gamma(2+x_{36})\Gamma(2+x_{37})}{\Gamma(2)\Gamma(2)}$$
$$\times\frac{\Gamma(6)}{\Gamma(6+x_{15}+x_{24}+x_{36})}\frac{\Gamma(3+x_{48})\Gamma(3+x_{49})}{\Gamma(3)\Gamma(3)}$$
$$\times\frac{\Gamma(4)}{\Gamma(4+x_{25}+x_{37})}\frac{\Gamma(2+x_{5\cdot10})\Gamma(2+x_{5\cdot11})}{\Gamma(2)\Gamma(2)}$$

where, with a slight abuse of notation, we let for example x_{24} be the data value associated with the edge leaving w_2 labelled θ_4; and where N is the sample size $=\sum_{n=1}^{3} x_{0n}$.

In this paper we have concentrated on the principle of assigning a score to a member of a candidate class. For a more formal presentation of an algorithm for searching over this class see (Freeman and Smith, 2009). An expanded version of this paper appears at http://www2.warwick.ac.uk/fac/sci/statistics/crism/research/2009/paper09-07, including an example demonstrating the versatility of our method, and an extended discussion section.

Note that the inputs to our search algorithm will consist of a candidate set of models and data from the problem we are modelling. The candidate set may be constrained as described above. The output of the algorithm will be the MAP model given the data and our candidate set. As with learning BNs, *exhaustive* search will be superexponential in the number of problem variables. However, as with BNs for large problems, fast local search algorithms can be devised which quickly explore subclasses of CEGs that for contextual reasons are expected to explain the data well.

ACKNOWLEDGEMENTS

This research has been partly funded by the EPSRC as part of the project *Chain Event Graphs: Semantics and Inference* (grant no. EP/F036752/1).

REFERENCES

Cooper, G. F. and Herskovits, E. (1992). A Bayesian method for the induction of Probabilistic Networks from data. *Machine Learning*, 9(4):309–347.

Freeman, G. and Smith, J. Q. (2009). Bayesian model selection of Chain Event Graphs. Research Report, CRiSM.

Heckerman, D., Geiger, D., and Chickering, D. (1995). Learning Bayesian Networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243.

Shafer, G. (1996). *The Art of Causal Conjecture*. MIT Press.

Smith, J. Q. and Anderson, P. E. (2008). Conditional independence and Chain Event Graphs. *Artificial Intelligence*, 172:42–68.

Smith, J. Q., Riccomagno, E. M., and Thwaites, P. A. (2009). Causal analysis with Chain Event Graphs. Submitted to *Artificial Intelligence*.

Thwaites, P. A., Smith, J. Q., and Cowell, R. G. (2008). Propagation using Chain Event Graphs. In *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence*, Helsinki.