Learning Accurate Belief Nets using Explicitly-Labeled Queries[1]

Russ Greiner and Wei Zhou *Department of Computing Science, University of Alberta*

Bayesian belief nets (BNs) are typically used to answer a range of queries, where each answer requires computing the probability of a particular variable (*e.g.*, possible diagnosis) given some specified evidence. An effective BN-learning algorithm should, therefore, learn an *accurate* BN — *i.e.*, one that returns the correct answers to these specific queries. Our earlier [GGS97] motivated this objective, arguing that it makes effective use of the data that is encountered, and that it can be more appropriate than the typical "maximum likelihood" algorithms for learning BNs [Hec95]. This abstract will summarize, and extend, those results: first overviewing the complexities inherent in this task and then providing an effective algorithm.

We assume there is a stationary underlying distribution $P(\cdot)$ over N (discrete) random variables $\mathcal{V} = \{V_1, \ldots, V_N\}$. A user can "interact" with this distribution by asking *queries*, each of the form "What is value of $P(\mathbf{X} = \mathbf{x} \mid \mathbf{Y} = \mathbf{y})$?", where $\mathbf{X}, \mathbf{Y} \subset \mathcal{V}$. The "label" of each such query is the (numeric) probability value of this conditional event, over the underlying distribution — *e.g.*, "0.65" is the label of the labeled query "$P(\mathit{cancer} \mid \mathit{female}, \mathit{35yo}, \mathit{smoker}) = 0.65$". We assume there is a (stationary) distribution $sq(\cdot)$ over the set of all possible legal queries, where $sq(\mathbf{X} = \mathbf{x};\ \mathbf{Y} = \mathbf{y})$ is the probability that the query "What is the value of $P(\mathbf{X} = \mathbf{x} \mid \mathbf{Y} = \mathbf{y})$?" will be asked.

N.b., the query distribution $sq(\cdot)$ can be completely unrelated to the underlying distribution $P(\cdot)$ — *e.g.*, even though *"What is* $P(\text{Cancer} \mid \text{female, 35yo, smoker})$*?"* is asked 35% of the time, the actual value of $P(\text{Cancer} \mid \text{female, 35yo, smoker})$ could be 0, or 1, or any other value.

We can evaluate any belief net B by its "expected L_2-error", with respect to the actual query distribution $sq(\cdot)$ and underlying distribution $P(\cdot)$:

$$\text{err}_{sq,P}(B) \quad = \quad \sum_{\mathbf{x},\mathbf{y}} sq(\mathbf{x};\mathbf{y}) \cdot [B(\mathbf{x} \mid \mathbf{y}) - P(\mathbf{x} \mid \mathbf{y})]^2$$

where the sum is over all assignments $\mathbf{x}, \mathbf{y}$ to all subsets $\mathbf{X}, \mathbf{Y}$ of variables $\mathcal{V}$, each $P(\mathbf{x} \mid \mathbf{y})$ corresponds to the label for this query and $B(\mathbf{x} \mid \mathbf{y})$ to the value that B assigns to this query. (Typically $sq(\mathbf{x};\mathbf{y})$ will be 0 for most $\mathbf{x}$, $\mathbf{y}$ pairs.) The learner seeks a belief net that minimizes this score:

$$B^*_{err} \quad = \quad \underset{B}{\text{argmin}}\{\ \text{err}_{sq,p}(B)\ \}$$

We focus on the task of filling-in the CPtables $\Theta = \{c_{q^{(i)}|\mathbf{r}^{(i)}}\}_i$ of a given belief net structure G (where each $c_{q|\mathbf{r}}$ corresponds to the BN's value of $P(Q = q \mid \mathbf{R} = \mathbf{r})$, where $\mathbf{R} \subset \mathcal{V}$ is the set of Q's parents in G) when the learner is given an explicit set of "labeled

[1] See also [ZG99], which also considers 2 other learning models. We assume the reader is familiar with belief nets (aka Bayesian networks); see [Pea88].

queries", $LQ = \{\langle \mathbf{X}_i = \mathbf{x}_i;\ \mathbf{Y}_i = \mathbf{y}_i;\ p_i \rangle\}_{i=1}^{M}$, where these $[\mathbf{X}_i = \mathbf{x}_i,\ \mathbf{Y}_i = \mathbf{y}_i]$ queries are drawn from the query distribution $sq(\cdot)$, then "labeled" $p_i \in [0,1]$ based on the underlying probability $P(\cdot)$. This LQ set will *not* include every possible query, and in particular will omit any query that $sq(\cdot)$ assigns 0 probability.

To address the two obvious learning challenges: First, note that learning is typically difficult when conditioning events are extremely small; we therefore define, for any $\gamma > 0$,

$$\mathcal{BN}_{\Theta \succeq \gamma}(G) \quad = \quad \{\, B \in \mathcal{BN}(G) \mid \forall c_{q|\mathbf{r}} \in \Theta, c_{q|\mathbf{r}} \geq \gamma \,\}$$

to be the subset of BNs (instantiating the structure G) whose CPtable entries are bounded above γ. Then

Theorem 1 *Given any belief net structure G, requiring the specification of K CPTable entries $\Theta = \{c_{q_i|\mathbf{r}_i}\}_{i=1}^{K}$, let $\hat{B} \in \mathcal{BN}_{\Theta \succeq \gamma}(G)$ be the BN that has minimum* empirical *score*

$$\widehat{err}^{(LQ)}(B) \ = \ \frac{1}{|LQ|} \sum_{\langle \mathbf{x}; \mathbf{y};\, p \rangle \in LQ} [B(\mathbf{x} \,|\, \mathbf{y}) - p]^2$$

with respect to a sample LQ of

$$M_{LQ}(\epsilon, \delta, \gamma) \quad = \quad \frac{18}{\epsilon^2} \left(\log \frac{2}{\delta} \ + \ K \log \frac{6\,K}{\gamma\,\epsilon} \right)$$

labeled queries from $sq(\cdot)$. Then, with probability at least $1 - \delta$, $\hat{B}$ will be no more than ϵ worse than the optimal member of $\mathcal{BN}_{\Theta \succeq \gamma}(G)$, B^* — i.e.,

$$P[\ err_{sq,\,p}(\hat{B}) \ \leq \ err_{sq,\,p}(B^*) + \epsilon\ ] \ \geq \ 1 - \delta. \qquad \blacksquare$$

This sample complexity remains polynomial even if $\gamma = 1/2^N$.

While the sample complexity is not bad, the computational complexity is problematic:

Theorem 2 *It is NP-hard to compute the CPTables for a given belief net structure that produce the BN with the minimum error wrt a given set of labeled queries LQ. This hardness result holds even if we consider only members of $\mathcal{BN}_{\Theta \succeq 1/2^N}(G)$.*

N.b., we cannot simply fill in the CPtables using the frequency estimates [Hec95, CH92], as our training data does NOT include such "tuples".[2]

This hardness result inspired us to build a hill-climbing algorithm, that changes the CPtables along this gradient. One obvious approach, inspired by [BKRK97], is simply to update each individual CPtable entry $c_{q|\mathbf{r}}$ based on $\frac{\partial\ \widehat{\mathrm{err}}^{(LQ)}(B)}{\partial\ c_{q|\mathbf{r}}}$. However, that ignores the constraints that $c_{q|\mathbf{r}} \geq 0$ and $\sum_i c_{q_i|\mathbf{r}} = 1$. We therefore used the parameterization $c_{q_i|\mathbf{r}} \ = \ \frac{e^{\beta_{q_i,\mathbf{r}}}}{\sum_i e^{\beta_{q_i,\mathbf{r}}}}$, and sought the optimal values of $\{\beta_{q,\mathbf{r}}\}$. This required the result:

[2] Of course, we could consider using a sample of tuples to approximate the "p_i" labels — *i.e.*, learn from only "unlabeled" queries and tuple samples. Even here using the frequency estimates is problematic, as it may produce a belief net that has unnecessarily high error if the given BN-structure is incorrect, and even if the structure is correct, it may still converge very slowly to the appropriate CPtable values; see [GGS97].

Theorem 3 *Given a set of queries* $LQ = \{[\mathbf{x}_i, \mathbf{y}_i, p_i]\}$*, the total gradient wrt a single* $\beta_{q\mathbf{r}}$ *term (corresponding to the CPtable entry* $c_{q|\mathbf{r}}$*) is*

$$\frac{\partial P(\mathbf{X} \mid \mathbf{Y})}{\partial \beta_{q\mathbf{r}}} = P(\mathbf{X}, q, \mathbf{r} \mid \mathbf{Y}) - P(\mathbf{X} \mid \mathbf{Y}) P(q, \mathbf{r} \mid \mathbf{Y}) - c_{q|\mathbf{r}} [P(\mathbf{X}, r \mid \mathbf{Y}) - P(\mathbf{X} \mid \mathbf{Y}) P(\mathbf{r} \mid \mathbf{Y})]$$

where $B(\mathbf{x} \mid \mathbf{y})$ *is the value the belief net* B *assigns to the "*$P(\mathbf{x} \mid \mathbf{y})$*" query.*

We implemented this algorithm, and demonstrated that it worked effectively over a variety of both artificial and real domains; see [ZG99].

Open problems: 1. Remove the annoying "$\Theta \succeq \gamma$" restriction. 2. Produce learning algorithms that are guaranteed to return CPtable entries that produce a good *approximation* to the optimal ones. 3. Extend these results to consider ways to learn the *structure* of a belief net (as well as CPtable entries).

References

[BKRK97] J. Binder, D. Koller, S. Russell, and K. Kanazawa. Adaptive probabilistic networks with hidden variables. *Machine Learning*, 29:213–244, 1997.

[CH92] G. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–347, 1992.

[GGS97] R. Greiner, A. Grove, and D. Schuurmans. Learning Bayesian nets that perform well. In *UAI-97*, 1997.

[Hec95] D. Heckerman. A tutorial on learning with Bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, 1995.

[Pea88] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[ZG99] W. Zhou and R. Greiner. Learning accurate belief nets. Technical report, UofAlberta CS, 1999. http://www.cs.ualberta.ca/~greiner/BN-results.html#accurate.

[Russ Greiner; Department of Computing Science; University of Alberta; 615 General Service Bldg; Edmonton, Alberta T6G 2H1; Canada
greiner@cs.ualberta.ca 780 492-5461]

Contributed for [either oral/poster] presentation