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

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

The companion abstract "*Learning Accurate Belief Nets using Explicitly-Labeled Queries*" provides an algorithm for learning the CPtables for a given belief structure from a set of labeled queries $LQ = \{\langle \mathbf{X}_i = \mathbf{x}_i;\ \mathbf{Y}_i = \mathbf{y}_i;\ p_i\rangle\}_{i=1}^{M}$, with the understanding that each $\langle \mathbf{X}_i = \mathbf{x}_i;\ \mathbf{Y}_i = \mathbf{y}_i;\ p_i\rangle$ corresponds to the claim that the "correct" value for the query $P(\,\mathbf{X}_i = \mathbf{x}_i \,|\, \mathbf{Y}_i = \mathbf{y}_i\,)$ is $p_i \in [0,1]$, where of course $\mathbf{X}_i, \mathbf{Y}_i$ are subsets of the variables, whose legal values include $\mathbf{x}_i$ and $\mathbf{y}_i$ respectively. The goal there is to find the belief net $B^*_{err} = \operatorname{argmin}_B\{\,\widehat{\text{err}}^{(LQ)}(\,B\,)\}$ with minimum error:

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

Note this LQ is a sample, drawn from the distribution of *queries*, $sq(\,\cdot\,)$ (which we assume to be stationary, but unknown to the user); and so this $\widehat{\text{err}}^{(LQ)}(\,B\,)$ score is an approximation to the true error

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

That abstract argued that this idea, of learning belief nets (BN) from explicitly-labeled queries, has several advantages over the standard approach of finding the belief net that is most likely, given a set of domain tuples. First, queries are naturally available because people use BNs to answer queries. Therefore query data is intuitive to use and easily obtained. Second, a query contains extra information on what variable is queried and what variables are observed. This extra information allows the learning algorithm to focus on updating only the parameters relevant to improving the BN's performance on answering these queries; this means the learning process can be more (sample) efficient. Third, training BNs on queries is a more correct approach if the trained BNs will later be used to answer queries.

That algorithm, however, requires knowing the correct labels for a set of queries — *i.e.*, the learner must know the label $p_i = P(\,\mathbf{x}_i\,|\,\mathbf{y}_i\,)$ in the given $\langle \mathbf{x}_i; \mathbf{y}_i; p_i\rangle$ "training" data. Such values may not be easy to obtain. In some situations, for example, we may only know that a specific patient had cancer, given certain information and evidence (*e.g.*, `cancer` given `35yo`, `female`, and `smoker`). A doctor might record this in a table — see the first line of Table 1. The other rows correspond perhaps to other patients, etc. Note that this table does not specify the exact probability p_i that a patient has a disease given evidence, but instead provides just the instances. Also, the data in this table is quite "sparse", as it only includes values for the evidence and query variables for the questions that were asked.

[1] See also [ZG99]. We assume the reader knows about belief nets (aka Bayesian networks); see [Pea88].

Age	Gend	Smok	Temp	LivBio	Btest	EKG	Cancer	Menin
35	F	Y	$\star$	$\star$	$\star$	$\star$	Yes	$\star$
25	M	N	$\star$	$\star$	$\star$	$\star$	No	$\star$
$\star$	$\star$	$\star$	$\star$	True	$\star$	$\star$	Yes	$\star$
$\star$	F	$\star$	$\star$	False	True	$\star$	$\star$	Yes
$\star$	F	$\star$	High	True	$\star$	$\star$	$\star$	No

Table 1: Sample of Implicitly Labeled Queries

We therefore consider the task of learning a good BN, given such a table, together with the knowledge that `Cancer` and `Menin` are queried variables and the rest are evidence variables. As we assume the most-likely outcomes typically occur, it makes sense to seek a BN that maximizes the "conditional likelihood" of the queried variables, given the evidence. We therefore define the "(empirical) conditional log likelihood" of a belief net B as

$$\widehat{\mathrm{CLL}}^{(IQ)}(\,B\,) \;=\; \frac{1}{|IQ|} \sum_{\langle \mathbf{x},\mathbf{y}\rangle \in IQ} \log(\,\hat{P}_B(\,\mathbf{x}\,|\,\mathbf{y}\,)\,) \qquad (1)$$

which of course is an approximation to the "(true) conditional log likelihood" of a belief net B, given the true query distribution $sq(\,\cdot\,)$ and true underlying distribution $P(\,\cdot\,)$:

$$\mathrm{CLL}_{sq,P}(\,B\,) \;=\; \sum_{\langle \mathbf{x},\mathbf{y}\rangle} sq(\,\mathbf{x}\,;\,\mathbf{y}\,) \;\cdot\; \log(P(\,\mathbf{x}\,|\,\mathbf{y}\,))$$

Our learner seeks the belief net that maximizes this score.

Unfortunately,

Theorem 1 *It is NP-hard to find the values for the CPtables of a fixed BN-structure that produce the largest (empirical) conditional likelihood (Equation 1) for a given set of implicitly-labeled queries.* ∎

Motivated by [BKRK97], we therefore built a hill-climbing algorithm, called *ILQ*, that climbs along the gradient

$$\frac{\partial\, \widehat{\mathrm{CLL}}^{(IQ)}(\,BN\,)}{\partial\, c_{q|\mathbf{r}}} \;=\; \frac{1}{c_{q|\mathbf{r}}}\,[B(\,q,\mathbf{r}\,|\,\mathbf{x},\mathbf{y}\,) - B(\,q,\mathbf{r}\,|\,\mathbf{y}\,)]$$

Our experiments show that this *ILQ* learner performed well on many different problems.

As one illustrative example, consider the net $B_{AXC} = \boxed{A \to X \to C}$, in the context when the X variable is never present; here the data D contained m copies of $\langle 1 \star 1\rangle$ and m copies of $\langle 0 \star 0\rangle$. Given that C was a query variable and A was evidence, our *ILQ* algorithm correctly instantiated the B_{AXC} network by making $A \equiv X \equiv C$, which clearly is appropriate.

In contrast, other standard algorithms, such as the APN gradient-ascent algorithm [BKRK97] or standard EM, performed poorly here, repeatedly returning the values $c_{x|a} = c_{x|\neg a}$ and $c_{c|x} = c_{c|\neg x} = 0.5$, where in general $c_{w|z}$ is CPtable associated with the $Z \to W$ link. [ZG99] proves that those values are in fact (respectively) points of 0-derivative and fixed-points. ([ZG99] also provides other examples that illustrate *ILQ*'s effectiveness.)

Open problems: 1. Produce efficient learning algorithms that return CPtable entries that produce a good *approximation* to the optimal ones. 2. 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.

[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.

[Wei Zhou; Department of Computing Science; University of Alberta; 615 General Service Bldg; Edmonton, Alberta T6G 2H1; Canada
wei@cs.ualberta.ca]

Contributed for [either oral/poster] presentation