How an Expert can use Imperfect Knowledge to Improve an Imperfect Theory

Russell Greiner and Jie Cheng
Department of Computing Science
University of Alberta
Edmonton, Alberta T6G 2H1 Canada

Christian Darken
Adaptive Information and Signal Processing
Siemens Corporate Research
Princeton, NJ 08540 USA

Abstract

This report addresses the challenge of using auxiliary information I_A to improve a given theory, encoded as a belief net $\mathcal{B}_E$. In contrast with many other "knowledge revision" systems, we deal with the situation where this I_A may be *imperfect*, which means $\mathcal{B}_E$ should not necessarily incorporate that information. Instead, we provide tools to help the expert decide how to use I_A. After presenting objective criteria for measuring how much I_A *differs* from $\mathcal{B}_E$, we discuss ways to evaluate whether this difference is *statistically significant*. We then provide tools to *isolate* the differences — to tell the domain expert which parts of the belief net (*e.g.*, which links, and/or which nodes) account for the discrepancy.

Two of our tools involve techniques that are of independent interest: *viz.*, the use of a non-central χ^2-test to compute the relative likelihood of two similar belief nets, and a sensitivity analysis that provides the "error-bars" around the answers returned by a belief net, as a function of the samples used to learn it.

Keywords: theory revision, learning, systematic errors, knowledge acquisition

1 Introduction

Human experts are indispensable in producing effective Decision Support Systems (DSSs). They typically provide the initial structure of the DSS, including the qualitative "what depends on what" knowledge, as well as some of the quantitative information. Unfortunately, their knowledge is not always completely correct: first, there may be significant gaps in their knowledge; *e.g.*, an expert who knows the electrical system of a large plant may not be as familiar with the hydraulic parts. Second, the expert may be wrong in certain aspects — due to human fallability, misunderstandings, outdated knowledge, or the fact that people are notoriously bad at providing quantitative information — *e.g.*, we often under-estimate the probability of rare events, among other problems [KT82].

It is therefore important to correct and augment an expert's knowledge, using some other source(s) of knowledge — perhaps other human experts, or data produced by the plant, or by a simulator. Unfortunately, such alternative sources are seldom perfect: The alternative expert has the same class of limitations as the original domain expert. And a set of samples that trace the behaviour of the plant is also problematic, in terms of both quality (due to measurement error, as well as the problems induced if the plant is not stationary) and quantity (as it may be very hard to obtain enough relevant samples). While quantity may not be an issue if the samples are produced by a simulator, the quality may be further compromised, as a simulator is only as good as its model, and many models have systematic (irremediable) problems — perhaps by being able to handle only singleton faults while the plant may have multiple problems, etc.

This report addresses the challenges of using this auxiliary, albeit imperfect, information to help improve the theory (*e.g.*, DSS[1]) produced by the domain expert. In particular, we start with a theory that has been approved by a domain expert, typically because he authored this system. We will assume this theory is encoded as a *belief network*, called $\mathcal{B}_E$. The expert then has access to another body of auxiliary information I_A — perhaps encoded as an alternative belief net $I_A \equiv \mathcal{B}_A$, or (for most of our analyses) provided as a set of samples $I_A \equiv S_A = \{s^{(j)}\}$, or whatever. The expert is then allowed to use this additional informa-

[1]While we word our task in terms of a diagnostic decision support tool, our ideas apply to reasoning in general, especially when the reasoner is encoded as a belief net.

tion to revise his initial theory, doing anything from completing ignoring I_A and keeping the original $\mathcal{B}_E$-based DSS, to throwing out his $\mathcal{B}_E$ in favor of (a DSS based on) I_A, or more typically, incorporating some information from I_A within $\mathcal{B}_E$. Note that we are intentionally leaving the expert "in the loop", as he may well possess insights and "implicit" knowledge that go beyond the information explicitly embodied within his $\mathcal{B}_E$. We provide below a set of tools to help the expert in this task.

Section 2 first provides the formal framework for our work. It begins with a mini-summary of belief nets — the formalism we will use to encode the probabilistic knowledge used by our DSS. It then discusses ways to measure how closely the expert's $\mathcal{B}_E$ *corresponds* to (or more importantly, *differs from*) another model. We present two formal evaluation-approaches here: the "query based" approach assumes that we know which questions will be posed to the eventual DSS, while the "distribution only" model, that we do not know this "future query" information. Section 2 concludes with a synopsis of the relevant literature — explaining in particular why our results are not subsumed by the work on *learning/revising* belief nets.

These evaluation approaches allow the expert to measure the difference between $\mathcal{B}_E$ and I_A. Section 3 extends this, by providing "error-bars" around this difference, to determine whether the observed raw score is *significant*. The expert can use this result to decide whether to consider revising his $\mathcal{B}_E$. By applying this analysis at the single-query level, the expert can get some insights for *where* to consider revising; *i.e.*, which parts of the belief net seem different from the other corpus. Section 4 goes further, by providing specific tools designed to isolate the differences. In particular, one tool can identify which specific *causal links* in the belief net are not supported by I_A, and so are likely to be the source of the discrepancy. Another tool suggests which *features* are problematic — *i.e.*, which variables are handled differently in the two models. Finally, Section 5 summarizes our contributions.

2 Framework

2.1 Introduction to Belief Nets

In general, a belief net (a.k.a. Bayesian network, probability net, causal net) $\mathcal{B} = \langle N, L, \Theta \rangle$ is a laconic way to represent an arbitrary distribution. Technically, each random variable is represented by a node $n \in N$ in a directed acyclic graph (DAG). Each directed link $\ell \in L$ denotes a "probabilistic" (some say "causal") dependency, in the sense (for example) that having a particular disease "causes" a patient to exhibit some symptom. Each node $n_i \in N$ also includes a "conditional probability table" (aka "CPtable") $\theta_i \in \Theta$, which specifies the distribution of this variable, as a function of the values of its parents. If all of the variables are binary, we can represent the CPtable for the node X, with k parents $\mathrm{Pa}(X) = \{Y_1, \ldots, Y_k\}$, by a table with 2^k rows, each row representing one possible assignment to the node's parent-set. If the i^{th} row is indexed by $\langle Y_1, \ldots, Y_k \rangle = \langle y_1, \ldots, y_k \rangle$ (where each $y_i \in \{0,1\}$ is the value of the random variable Y_i), then the values for this row specify $P(X = 1 \mid \langle Y_1, \ldots, Y_k \rangle = \langle y_1, \ldots, y_k \rangle)$ and $P(X = 0 \mid \langle Y_1, \ldots, Y_k \rangle = \langle y_1, \ldots, y_k \rangle)$. (See [Pea88] for more information about Belief Nets in general.)

2.2 How closely does I_A correspond to $\mathcal{B}_E$?

As noted above, we will often want to determine how closely the given belief net $\mathcal{B}_E$ *corresponds* to the alternative model I_A, for some sense of the term "correspond". Here we focus on situations where this I_A is derived from a sample of data S_A. Section 2.2.1 addresses this question in the context where we have only the two distributions corresponding to $\mathcal{B}_E$ and S_A. Section 2.2.2 then provides a measure that applies when we also know the range of questions that the eventual DSS will have to answer. (Section 3 will later discuss the statistical significance of the measure.)

2.2.1 Distribution-Only (Query-Free) Model

One natural way to relate a set of tuples $I_A \sim S_A = \{s^{(j)}\}$ to a given distribution (here, represented by the belief net $\mathcal{B}_E$) is to compute the probability of S_A, given the distribution — *i.e.*, $P(S_A \mid \mathcal{B}_E) \equiv P_{\mathcal{B}_E}(S_A)$. The larger the score, the more likely that the observed S_A would be produced from $\mathcal{B}_E$; this supports the view that S_A corresponds to $\mathcal{B}_E$. Many belief-net–learning algorithms therefore use this measure (typically augmented with a regularizing term) to determine which belief net is best [Hec95].

If we view the sample $S = S_A$ as a distribution — *i.e.*, define $P_S(x) = 1/|S|$ if $x \in S$, and 0 otherwise[2] — this measure relates to the standard measure between two distributions: "Kullback-Leibler divergence" [KL51],

$$KL(P_1\,;\,P_2) \quad = \quad \sum_x P_1(x) \ln \frac{P_1(x)}{P_2(x)}$$

Hence,

$$KL(P_S; P_{\mathcal{B}}) \quad = \quad \sum_x P_S(x) \ln\left(\frac{P_S(x)}{P_{\mathcal{B}}(x)}\right) \quad =$$
$$-\ln(K) - \frac{1}{K}\sum_{s\in S} \ln(P_{\mathcal{B}}(s)) \;\; = \;\; -\ln(K) - \frac{1}{K}\ln(P_{\mathcal{B}}(S))$$

[2] To simplify the presentation, we will assume that each element of S appears only once in the sample. Also, we will use $P_{\mathcal{B}}(q)$ to refer to the probability that $\mathcal{B}$ assigns to the query q.

(using the fact that the $s_i \in S$ are iid), where $K = |S|$, and the "x" in the first summation ranges over all $2^{O(n)}$ assignments to the variables. Hence, holding S fixed, the $\mathcal{B}$ that maximizes $P_{\mathcal{B}}(S)$ will be the one that minimizes $KL(P_S; \mathcal{B})$.

KL is non-negative, and (in the discrete case) is 0 iff $\forall x\ P_{S_A}(x) \equiv \mathcal{B}_E(x)$ — *i.e.*, if the belief net $\mathcal{B}_E$ and the sample S_A coincide. Thus $KL = 0$ means the two distributions will "behave" identically, by providing the same answer to any query.

2.2.2 Query-Based Model

While this "equivalence" condition is desirable, it is unlikely to hold. Fortunately, we typically do not require so strong a condition. In general, it is sufficient to know that the two distributions will provide similar answers to the *relevant* queries — *i.e.*, to the queries that are actually posed.

We therefore can consider *a distribution over the queries* $P_Q(q)$ — *i.e.*, we consider the probability of *asking* a question "q", where each question q is of the form *"What is the probability $P(A = a \mid \vec{B} = \vec{b})$?"*, where A is a variable, a is a value of A, $\vec{B} = \langle B_1, \ldots, B_r \rangle$ is a vector of zero or more variables, and $\vec{b} = \langle b_1, \ldots, b_r \rangle$ is a corresponding vector of values. For example, the assertion P_Q(Cancer; Male, Age=42, Smoker) = 0.35; means that 35% of the time, the DSS will ask *"What is P(Cancer | Male, Age=42, Smoker)?"*; *i.e.*, for the probability that a patient has Cancer, given that the patient is male, is 42 years old, and is a smoker. The DSS may also ask *"What is P(Cancer | Female, Age=35, Smoker)?"* 5% of the time; *"What is P(Hepatitis | Jaundice)?"* 18% of the time; etc.

Note that the probability of asking a query (P_Q) can be totally unrelated to the underlying probability P — *e.g.*, even though *"What is P(Cancer | Male, Age=42, Smoker)?"* is asked 35% of the time, the actual value of P(Cancer | Male, Age=42, Smoker) could be 0, or 1, or any other value. We must therefore distinguish between

"tuple distributions": which specifies the probability that this (conditional) event will happen, *e.g.*, P(Cancer | Male, Age=42, Smoker) = 0.1.

"query distributions": which specifies the probability of *posing* some queries, such as *"What is P(Cancer | Male, Age=42, Smoker)?"* is asked 0.35 of the time.

Note that the distributions associated with the expert's assessment $\mathcal{B}_E$ and with the tuples S_A are each tuple distributions.

Here, we would like to know that these two "tuple-distributions" $P_{\mathcal{B}}$ and P_S provide essentially the same answers *for these queries*. Note that we don't care if the distributions provide very different answers to the other 0-probability queries — *e.g.*, if we never ask P(Smoker | Male), we should not care if the two distributions provide very different answers. For example, if P_Q("P(Smoker | Male)") = 0, then we should not worry if $P_{\mathcal{B}}$(Smoker | Male) = 0 but P_S(Smoker | Male) = 1.

To state this more quantitatively, given a distribution over queries P_Q, we can measure the difference between $\mathcal{B}$ and S as

$$Err_Q(\mathcal{B}, S) \;=\; \sum_q P_Q(q)\,[P_{\mathcal{B}}(q) - P_S(q)]^2 \qquad (1)$$

(We use this L_2 measure — $[P_{\mathcal{B}}(q) - P_S(q)]^2$ — as it is convenient for the derivations used in Section 3; we could have used essentially any other cost function.)

While $KL(\mathcal{B}, S) = 0$ implies $Err_Q(\mathcal{B}, S) = 0$ for *any* query distribution P_Q, the converse is false; in fact, it is possible for $KL(\mathcal{B}, S)$ to be arbitrarily large while $Err(\mathcal{B}, S) = 0$. (For example, suppose Q consists of only the single query *"What is P(A)?"*, to which $\mathcal{B}$ and S provide the same answer. However, $\mathcal{B}$ and S can be completely different with respect to every other feature, correlation, etc.)

There are some challenges to using this $Err_Q(\cdot, \cdot)$ measure. One issue is obtaining (at least an estimate of) the distribution over queries Q; see [GGS97] for a discussion of this issue. For this report, however, we will simply assume this distribution is given.

Another problem occurs if there are too many possible queries to examine each. Here, of course, we can *sample* from this space of queries, and replace Equation 1 with its empirical approximation; again see [GGS97].

2.3 Related Work

As our task often involves using a set of samples to produce an accurate belief net, our results appear related to the work on *learning* belief nets [Hec95, Bun96], and perhaps even more similar to *revising* a given belief net [LB94, HGC95]. Note, however, that almost all of the learning/revising systems assume that the given training sample (here I_A) is completely correct. This assumption does not hold in our situation, as we know there will be systematic errors in the data. (See [PDar, Bee57] for further motivation.) This means we should *not* necessarily incorporate (parts of) I_A into our emerging $\mathcal{B}_E$. Our goal, instead, is to simply inform the expert of possible discrepancies, here by identifying "components" of $\mathcal{B}_E$ that do not correspond to I_A. The expert can then decide on the proper response.

As another comparison: Every BN-learning algorithm must have some criterion to decide whether one BN is "better" than another. One component of this criterion is typically fit to the training data. Of course, a larger BN can only improve this empirical fit. Unfortunately, as a better empirical fit does not guarantee a better fit to the underlying distribution, most learners impose some "penalty" for each additional parameter. For example, given training data S, many MDL-based learners score a belief net $\mathcal{B}$ with m parameters as [FY96]

$$\log(P(S \,|\, \mathcal{B})) \;-\; \tfrac{m}{2} \log(|S|)$$

This means, when comparing $\mathcal{B}_1$ to a larger $\mathcal{B}_2$ that uses k more parameters, such MDL systems will prefer the simpler $\mathcal{B}_1$ unless

$$\log \left(\frac{P(S \,|\, \mathcal{B}_2)}{P(S \,|\, \mathcal{B}_1)} \right) \quad > \quad \tfrac{k}{2} \log(|S|)$$

Section 4.1 uses a similar expression when comparing two similar nets, to see which is better supported by the training data. However, rather than impose an absolute measure, this system first asks the user to specify some confidence bound, say $\alpha = 0.05$. It then advocates the simpler $\mathcal{B}_1$ unless

$$\log \left(\frac{P(S \,|\, \mathcal{B}_2)}{P(S \,|\, \mathcal{B}_1)} \right) \quad > \quad \tfrac{1}{2} \times t_\alpha$$

where t_α is the α quantile for the $\chi^2_k(\lambda)$ distribution (defined below). Here, the user can specify different α values in different contexts, depending on how confident he needs to be for each individual decision.

This issue is clearly related to the confidence measured used by the BN-learners that use conditional-independence tests to decide whether to include some link [GSSK87, CBL97]. Our test, however, is more fine-grained (at the level of a single modification, rather than dealing with the entire cascade of decisions made by general learning algorithms), and, again, is under the control of the user.

Also, some of our analyses assume we know how the eventual belief net will be used — *i.e.*, we know the distribution of queries that will be posed (see Section 2.2.2). This allows us to focus attention on producing a system that will perform well on these queries. This model, first expressed in [GGS97], differs from both of the dominant approaches, based on maximizing likelihood [Hec95] or finding independencies [GSSK87, CBL97].

Section 3 shows how the error-bars for an inference from a belief net varies with the sample size; this relates to the work on determining the sample complexity for learning belief nets [Höf93, FY96, Das97]. Of course, those papers dealt with the problem of identifying which of a given class of belief nets is best; by contrast, we are given a single belief net to evaluate. Also, while those other analyses are in terms of the "Maximize Likelihood"-based approaches, some of our analyses are in terms of the queries that will be addressed.

Section 4.2 considers situations where we may want to remove some features (aka attributes, variables). While this objective is superficially similar to the many feature-selection algorithms [KJ97, BL97], note that our goals are different: While many of those other systems try to remove *redundant* features (*i.e.*, features whose values can basically be determined from the other features), we are trying to identify those features where the expert and the training sample appear to *disagree*.

3 Is the Difference Significant?

Motivation: Suppose we are considering a particular belief net structure, and are using a set of samples S_A to fill-in the CPtables [Hec95, CH92]. We can then use the resulting belief net, $\mathcal{B}_A = \mathcal{B}(S_A)$, to answer questions; *i.e.*, to obtain a value $P_A(X = x \,|\, Y = y)$ for each *"What is $P(X = x \,|\, Y = y)$?"* query posed. Of course, the expert will also have an answer to each such question, written $P_E(X = x \,|\, Y = y)$, determined by his $\mathcal{B}_E$ belief net. Finding that $P_A(X = x \,|\, Y = y) = P_E(X = x \,|\, Y = y)$ suggests that $\mathcal{B}_A$ and $\mathcal{B}_E$ agree, wrt this query. If these values are different, however, the expert may be tempted to modify his $\mathcal{B}_E$. Note, however, that $\mathcal{B}(S_A)$'s CPtables depend on the actual S_A samples used, meaning the value of $P_A(X = x \,|\, Y = y)$, in turn, also depends on this sample. It is therefore possible that the difference between the value returned by $\mathcal{B}_E$ and by $\mathcal{B}(S_A)$ is due *only* to the sampling "error" in producing this S_A, which would mean that $\mathcal{B}_E$ and the source of S_A do in fact agree. If so, then this "$P_A(X = x \,|\, Y = y) \neq P_E(X = x \,|\, Y = y)$" discrepancy should *not* suggest modifying $\mathcal{B}_E$.

Therefore, before questioning whether $\mathcal{B}_E$ is wrong, we should first determine the "error-bars" around $\mathcal{B}(S_A)$'s answers to the *"What is $P(X = x \,|\, Y = y)$?"* queries. Then, for each query, we should only consider changing $\mathcal{B}_E$ if $|P_A(X = x \,|\, Y = y) \,-\, P_E(X = x \,|\, Y = y)|$ exceeds those error bars. This section therefore describes how to determine the error-bars around the answers returned by a completed belief net, as a function of the samples used to compute its CPtables.

Analysis: For now, we will consider only a single query $P(X \,|\, Y)$, and implicitly use the L_1 measure (the simple difference between the correct value and the estimated value); it is obvious how to generalize these results to handle an arbitrary distribution of queries, and to use the L_2 norm.

Assume there are K CPtable entries that can affect the $P(X \mid Y)$ query — *i.e.*, that are not d-separated from X, Y. (In general, this K will be significantly smaller than the total number of CPtable entries in the full belief net.[3]) Consider a single such CPtable entry $e_{q|\mathbf{r}}$, which specifies the probability that $Q = q$ given the assignment $\mathbf{R} = \mathbf{r}$. Assume we have collected N tuples S, and of these, $n_{\mathbf{r}} = |\,S[\mathbf{R} = \mathbf{r}]\,|$ have the variables $\mathbf{R}$ set to the values $\mathbf{r}$. We define $\hat{e}_{q|\mathbf{r}} = \frac{|\,S[Q=q,\mathbf{R}=\mathbf{r}]\,|}{n_{\mathbf{r}}}$ to be the empirical estimate, obtained from these $n_{\mathbf{r}}$ samples. We can use Hoeffding's Inequality [Hoe63] to see that, given these $n_{\mathbf{r}}$ samples, we can be at least $1 - \delta/K$ confident that our empirical estimate $\hat{e}_{q|\mathbf{r}}$ will be within

$$\lambda_{\mathbf{r}} \quad = \quad \sqrt{\tfrac{1}{2n_{\mathbf{r}}} \ln \tfrac{2K}{\delta}}$$

of the corresponding $e_{q|\mathbf{r}}$; *i.e.*,

$$P(\,|\hat{e}_{q|\mathbf{r}} - e_{q|\mathbf{r}}| > \lambda_{\mathbf{r}}\,) \quad < \quad \tfrac{\delta}{K}$$

Therefore, with probability at least $1 - \delta$, we can assume each of the K $\hat{e}_{q|\mathbf{r}}$ values will be within $\lambda_{\mathbf{r}}$ of the true $e_{q|\mathbf{r}}$ values. (If $n_{\mathbf{r}} = 0$, we can set $\lambda_{\mathbf{r}}$ to 1.)

How will that affect the value of our estimate $\hat{P}(X|Y)$ of $P(X \mid Y)$? Using

$$\frac{d\,P(X \mid Y)}{d\,e_{q|\mathbf{r}}} \quad = \quad \frac{P(\mathbf{r})}{P(Y)}\,[P(X,Y \mid q,\mathbf{r}) - P(X \mid Y)P(Y \mid q,\mathbf{r})] \qquad (2)$$

(derived in [GGS97]) we see that

$$\begin{aligned}
|\hat{P}(X|Y) - P(X \mid Y)| \quad &\leq \quad \sum_{\langle q,\mathbf{r}\rangle} \lambda_{\mathbf{r}} \times \frac{d\,P(X \mid Y)}{d\,e_{q|\mathbf{r}}} \\
= \quad & \sum_{\langle q,\mathbf{r}\rangle} \sqrt{\frac{1}{2\,n_{\mathbf{r}}} \ln \frac{2K}{\delta}} \times \frac{P(\mathbf{r})}{P(Y)} \times \\
& \qquad [P(X,Y \mid q,\mathbf{r}) - P(X \mid Y)P(Y \mid q,\mathbf{r})] \\
\approx \quad & \sqrt{\frac{1}{2\,N} \ln \frac{2K}{\delta}} \times \frac{1}{P(Y)} \times \\
& \quad \sum_{\langle q,\mathbf{r}\rangle} \sqrt{P(\mathbf{r})}\,[P(X,Y \mid q,\mathbf{r}) - P(X \mid Y)P(Y \mid q,\mathbf{r})]
\end{aligned}$$

using the observation that $\frac{n_{\mathbf{r}}}{N} = \hat{P}(\mathbf{r}) \approx P(\mathbf{r})$. As we are assuming that $\hat{P}$ is close to P, we can use the estimated $\hat{P}(\cdot)$ terms above.

Comments: (1) We see immediately that this error decreases in an $O(1/\sqrt{N})$ fashion.

[3] In one short empirical study, using the ALARM network and a reasonable distribution of queries [HC91], we found that, on average, only 10% of the CPtable entries were actually involved with any computation.

(2) This expression also suggests which type of new tuples may be most useful, towards reducing the error bars. In particular, if we are running a simulator and so are able to select the samples, we should focus on the particular instances that "hit" the particular CPtable entries $e_{q|\mathbf{r}}$ that most effect (*i.e.*, have the highest derivative wrt) the relevant queries "$P(X \mid Y)$" — for example, the queries that still have the highest variance.

(3) We can use this analysis to help identify which queries are "unambiguous" — *i.e.*, where $\mathcal{B}_E$ and S_A agree. We may then choose to believe these queries, but be skeptical of the others.

(4) Here, we used a set of samples S_A to obtain the error-bars. However, all that we used (beyond the empirical average) was the "amount of evidence": how many examples matched a certain assignment of the parent nodes. We can sometimes obtain this information in other contexts; *e.g.*, it is implicitly provided by a Dirichlet distribution [Hec95]. Therefore, we can also obtain error-bars associated with the expert's $\mathcal{B}_E$ belief net if we know such statistics about its nodes — perhaps because the value of the $\mathcal{B}_E$'s CPtables were also filled using samples, or because the expert who filled in these values could also provide "equivalent sample size" (*e.g.*, "the probability that `Temp = high` given `Cancer = true` is 0.45, and this is based on having seen 200 `Cancer` instances"). Here, we could of course combine the two error-bars, from $\mathcal{B}_E$ and $\mathcal{B}(S_A)$, in a statistically appropriate manner [BD77].

(5) We can use the finding that $P_E(X \mid Y)$ appears statistically different from $P_A(X \mid Y)$ to help identify "problematic" parts of the $\mathcal{B}_E$; *i.e.*, we know that $\mathcal{B}_E$ and S_A differ wrt some of the links that are involved with (read "not d-separated from") such "significantly different" queries. Given a set of queries, we can further focus the search for problematic links by emphasizing the links that appear to be involved in many "problematic queries". We may then be able to use some of non-problematic queries to prune these options, as finding that I_A *agrees* with $\mathcal{B}_E$ on some query suggests that the links involved are likely to be *correct*. As a simple example, suppose $\mathcal{B}_E$ contains the links $\boxed{A \to B \to C}$. If I_A and $\mathcal{B}_E$ agree on the $P(C \mid B)$ query but disagree on $P(C \mid A)$, then we may suspect the $B \to C$ link is correct, but the $A \to B$ link is not. (Section 4 will provide better mechanisms for isolating these differences.)

(6) Of course, the general problem requires dealing with a distribution of queries; here we would have to weight each such error by the probability of that particular query. (And if using L_2 norm, we need to square this difference.) If there are too many queries

to evaluate the error bars of each, we can instead estimate this quantity, by sampling from the distribution of queries.

4 Isolating the Differences

The previous section discussed ways to decide whether two distributions (perhaps one obtained from an expert and the other obtained from a sample) are significantly different. If so, the next step is to isolate this difference; in our situation, this means pin-pointing just where the belief net differs from the other distribution. Section 4.1 considers the situation where this difference is possibly due to the *links* within the belief nets, in that the belief net may better match the sample if it includes a new link, or excludes an existing one. In particular, it provides a statistically sound technique (*viz.*, a non-central χ^2 test) for determining whether a net formed by adding, or by deleting, a few specified links is a better model of a given set of tuples. It is based on the "query-free (aka "distribution-only") approach — *i.e.*, in terms of the likelihood of the data, given the belief net.

Removing a single arc is a relatively small modification to a network; sometimes we may want to perform the bigger modification of removing an entire feature. To motivate this, imagine the person building the network was truly an expert, but only in one subdomain — perhaps in hydraulics. Hence we may expect the sub-network dealing with hydraulics to be essentially perfect. However, he may also add in nodes that represent the electrical system. It is possible that the expert will get these features (and associated sub-net) seriously wrong.

Section 4.2 therefore considers ways to identify which features give rise to differences between $\mathcal{B}_E$ and I_A, as this may point the expert to "regions" where his opinion, as encapsulated within $\mathcal{B}_E$, is questionable.[4] This tool also uses the "distributional approach"; we explain below why this task is not interesting in the query-based approach.

4.1 Problematic Links

The section provides a technique to determine whether the expert's $\mathcal{B}_E$ belief net omits some links that are sanctioned by the alternative distribution I_A (which was perhaps induced by a set of samples S_A). That is, form $\mathcal{B}'$ by adding some new links to the given $\mathcal{B}_E$ network. Clearly $\mathcal{B}'$, with strictly more degrees of freedom, cannot be a worse fit to any distribution (and so it cannot be a worse fit to I_A). However, if $\mathcal{B}_E$ is as good a fit, then the extra links in "$\mathcal{B}' - \mathcal{B}_E$" are superfluous.

Here, we provide a tool to address this issue: to determine whether $\mathcal{B}'$ is a *significantly* better fit to the I_A distribution, within the "query-free" approach. If so, this would suggest adding the extra links in $\mathcal{B}' - \mathcal{B}_E$. We can also use this same analytic tool to go the other way: Suppose the expert initially proposed the larger $\mathcal{B}_E = \mathcal{B}'$, and then observed that the reduced $\mathcal{B}''$, which omitted some links from $\mathcal{B}_E$, was still comparable to the larger $\mathcal{B}_E$. This would suggest deleting these apparently-superfluous links.

Given this tool, we could consider various ways of exploring the space of "new links to consider adding" and "existing links to consider deleting". One obvious approach is to focus on the links involved with problematic queries, as determined by the analysis in Section 3. We could also consider adding in new links, which have the potential to correct the discrepancies found by that analysis. Note also that our analysis can handle adding/removing arbitrary *sets* of links, as well as single links.

Finally, before providing our approach, it is worth quickly mentioning how this improves on the obvious idea of directly comparing the $\mathcal{B}_E$'s CPtable entries with the I_A distribution. That is, each $e_{q|\mathbf{r}}$ entry in $\mathcal{B}_E$ provides the expert's estimate of $P(\,Q = q \,|\, \mathbf{R} = \mathbf{r}\,)$ — *i.e.*, the probability that $Q = q$ given that Q's parents $\mathbf{R}$ have the values $\mathbf{R} = \mathbf{r}$. We could also get I_A's opinion of this quantity, and note when these two quantities are different. There are two problems with this naïve approach: First, it is too "*local*", as it would only deal with a single $\langle\, q, \mathbf{r}\,\rangle$ entry within Q's full CPtable $\{\langle\, Q = q_i \,|\, \mathbf{R} = \mathbf{r}_j\,\rangle\}$. For example, suppose we find that the expert's $P_E(\texttt{Fever} = 1 \,|\, \texttt{Cancer} = 1)$ value was very different from the alternative $P_A(\texttt{Fever} = 1 \,|\, \texttt{Cancer} = 1)$ value, but there is agreement for the other value $P_E(\texttt{Fever} = 1 \,|\, \texttt{Cancer} = 0) = P_A(\texttt{Fever} = 1 \,|\, \texttt{Cancer} = 0)$. Here, it is not clear whether we should delete this link, based on this one problem. Second, as noted above, we should only consider any modification if the differences are *significant*. The following analysis deal with both issues.

Analysis: Let G_s and G_b be two belief nets, both using the same random variables $X_1, \ldots, X_m$ as nodes, such that the links in G_s form a subset of those in G_b. Let n be the number of additional free parameters in a minimal representation of the CPtables of G_b as compared to G_s. To simplify the presentation, assume each random variable (rv) take values in the (finite) set

[4] It is still possible that the expert is correct in this situation, but the training sample was wrong. Recall that we are not assigning blame, but are instead simply identifying differences.

V, which includes the distinguished element $v^* \in V$. Thus a variable X_a which has j parents in G_s and $k > j$ parents in G_b, contributes $(|V|^k - |V|^j)(|V|-1)$ to n.

Definition 1 *For any sequence S of iid draws of $X_1, \ldots, X_m$, let P_s and P_b be the probability measures of the maximum-likelihood belief nets corresponding to G_s and G_b. Then we define the* likelihood ratio statistic *as $l = P_s(S)/P_b(S)$. Note that $0 \leq l \leq 1$.*

Note we can compute this Likelihood Ratio relatively efficiently: Let P_G be the probability distribution of a specific belief network with graph G. Then for samples $S = \{s^{(j)}\}_i$ (where each $s_j = \langle s_1^{(j)}, \ldots, s_m^{(j)} \rangle$), $P_G(D) = \prod_i P_G(s^{(j)}) = \prod_j \prod_i P_G(X_i = s_i^{(j)} \,|\, \mathrm{Pa}(X_i) = s^{(j)}_{\mathrm{Pa}(X_i)})$, where $\mathrm{Pa}(X)$ refers to the parents of the node X. Hence, when calculating l, we can cancel the terms corresponding to nodes that have the same set of parents in G_b that they have in G_s.

Consider a parameterization (ϕ, θ) of the G_b belief net such that, if we constrain each element of θ to be zero, ϕ is a valid parameterization of G_s. Then:

Proposition 2 ([SO91, Roy57]) *Let H_s (resp., H_b) be the hypothesis that the data was generated by a distribution representable by G_s (resp., by G_b, but not by G_s).*
When H_s holds, $-2\ln l$ asymptotically has a χ^2 distribution with n degrees of freedom.
When H_b holds, $-2\ln l$ asymptotically has a non-central *χ^2 distribution with n degrees of freedom and non-central parameter having the form*

$$\lambda \;=\; \theta^T \, M \, \theta \tag{3}$$

where $M_{ij} = -E_{X_1,\ldots,X_m}\left(\frac{\partial^2 \ln P_b(D)}{\partial\theta_i \partial\theta_j}\right)$

Section 4.1.1 below gives an example that illustrates how to produce this (ϕ, θ) parameterization, and then how to compute this λ; Section 4.1.2 then shows that this test works in practice. Appendix A provides the general description.

Use as a Statistical Test: We expect l to be close to 1 when H_s holds, and close to 0 otherwise; hence $-2\ln l \approx -2\ln 1 = 0$ when H_s holds, and will blow up towards infinity otherwise. Therefore, we can establish a threshold t_α, measure $1 - \alpha = P(\chi^2_n(\lambda) < t_\alpha)$, and reject H_b with probability $p \geq \alpha$ if $l \leq t_\alpha$. (A potential weakness of this test is that its exact meaning is unclear when *neither* H_s nor H_b holds; however the heuristic argument as to the behavior of l is still valid).

4.1.1 Example of Likelihood Ratio Test

Consider comparing the two-node belief net $G_b \equiv \boxed{A \to B}$ with the simpler net G_s that contains only the nodes $\{A, B\}$ but no arcs. For G_s, there are two CPtable entries: $P(A = t)$ and $P(B = t)$. For G_b there are three: $P(A = t)$, $P(B = t \,|\, A = f)$, and $P(B = t \,|\, A = t)$. We can represent the latter two G_b CPtable entries as

$$\begin{aligned} P(B = t \,|\, A = f) &= P(B = t) + \theta \\ P(B = t \,|\, A = t) &= P(B = t) + \left(1 - \tfrac{1}{P(A=t)}\right)\theta \end{aligned}$$

where θ is a new parameter. Note that the probability distribution represented by the G_b network reduces to the one represented by G_s when $\theta = 0$. (This parameterization follows immediately from the constraint that $P(B = t) = P(B = t \,|\, A = f)P(A = f) + P(B = t \,|\, A = t)P(A = t)$.)

Defining $d_{ij} = \delta(B, i)\,\delta(A, j)$, we can write

$$\begin{aligned} P(B|A) &= d_{TF}[P(B = t) + \theta] \\ &+ d_{FF}[1 - P(B = t) - \theta] \\ &+ d_{TT}\left[P(B = t) + \left(1 - \tfrac{1}{P(A=t)}\right)\theta\right] \\ &+ d_{FT}\left[1 - P(B = t) - \left(1 - \tfrac{1}{P(A=t)}\right)\theta\right] \end{aligned}$$

Further defining $P_T = P(A = t)$, $P_{TT} = P(B = t|A = t)$, and $P_{TF} = P(B = t|A = f)$, we have

$$\frac{\partial \ln P(B,A)}{\partial\theta} = \frac{\partial \ln P(B\,|\,A)}{\partial\theta} = \frac{1}{P(B|A)}\left[d_{TF} - d_{FF} + d_{TT}\left(1 - \tfrac{1}{P_T}\right) - d_{FT}\left(1 - \tfrac{1}{P_T}\right)\right]$$

(the first equality uses the observations that $P(B, A) = P(B\,|\,A)P(A)$ and $P(A)$ does not depend on θ) and so

$$\frac{\partial^2 \ln P(B,A)}{\partial\theta^2} = \frac{\partial^2 \ln P(B\,|\,A)}{\partial\theta^2} = \frac{-1}{P(B\,|\,A)^2}\left[d_{TF} + d_{FF} + d_{TT}\left(1 - \tfrac{1}{P_T}\right)^2 + d_{FT}\left(1 - \tfrac{1}{P_T}\right)^2\right]$$

(This requires the observations that exactly one of the d_{ij} is 1, and the others are 0, which means each cross-term $d_{ij}\,d_{k\ell}$ is 0, and $d_{ij}^2 = d_{ij}$.)

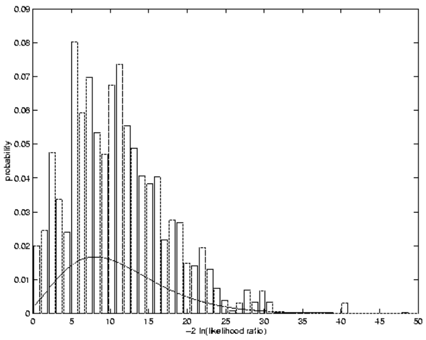
Letting S be a set of $|S|$ iid draws of $\langle A, B \rangle$,

$$\begin{aligned} \frac{M}{|S|} &= -E_{\langle A,B\rangle}\frac{\partial^2 \ln P_b(B\,|\,A)}{\partial\theta^2} = \\ &\sum_{A,B}\frac{P(A,B)}{P(B\,|\,A)^2}\left[d_{TF} + d_{FF} + (d_{TT} + d_{FT})\left(1 - \frac{1}{P_T}\right)^2\right] \\ &= \frac{1-P_T}{P_{TF}[1-P_{TF}]} + \frac{P_T}{P_{TT}[1-P_{TT}]}\left(1 - \frac{1}{P_T}\right)^2 \end{aligned}$$

Given this, together with $\theta = P_T(P_{TF} - P_{TT})$, we can use Equation 3 to compute λ.

4.1.2 Empirical Study

We implemented this tool, and investigated how well it worked, in the context of the simple 2-node nets presented above. We found it worked very reliably, in that it allowed us to correctly accept G_b (resp., G_s) where appropriate. For example, G_b is the correct model when the distribution is $P(A = t) = 0.5$, $P(B = t \mid A = t) = 0.55$ and $P(B = t \mid A = f) = 0.45$. Using the formulas shown above, this correspond to non-centrality parameter $\lambda = 10.1$, and $n = 1$ extra parameter. Below we plot the values of the $\chi_1^2(10.1)$, as well as the (differently scaled) empirical histogram over 10,000 trials, where each trial involved 1000 samples. If we base our decision on the observed empirical score, we will almost always decide on G_b, as is appropriate.



4.2 Problematic Features

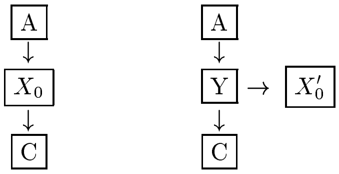
This section also deals with the situation where the two distributions do not correspond, in the query-free (distribution-only) model — *i.e.*, $P_{\mathcal{B}_E}(S_A)$ is unacceptably large. Here, however, we attempt to isolate the problem by considering the possibility that certain *features* are problematic.

One obvious approach is to compare this $P_{\mathcal{B}}(S)$ score with the scores obtains by deleting (from both $\mathcal{B} = \mathcal{B}_E$ and $S = S_A$) some feature X_0. That is, given feature X_0, let S_0 be the set of samples formed by deleting the "X_0-column" from the S data. Similarly let $\mathcal{B}_0$ be the belief net formed by marginalizing away the X_0 variable from $\mathcal{B}$ — this can involve first deleting this X_0 node and all connecting arcs, and then reconnecting each parent of X_0 to each other, and to each child of X_0, and finally filling in the appropriate marginalized value, based on all of X_0's (former) co-parents. For example, removing B from $\boxed{A \to B \to C}$ produces a network $\boxed{A \to C}$; note that just deleting the $A \to B$ and $B \to C$ arcs would instead produce a network with 3 independent features. (Hence, removing X_0 does more than just remove each of the arcs connecting X_0 to the rest of the network.)

If we see that $\mathcal{B}_0$ is significantly "closer" to S_0, than $\mathcal{B}$ is to S, that suggests that $\mathcal{B}$ and S have different "interpretations" of X_0. Here, we will have to decide which (if either) of $\mathcal{B}$ or S best "understood" X_0 — perhaps by asking an X_0-expert.

The challenge is finding an appropriate measure for this comparison. This is complicated by the observation that $P_{\mathcal{B}_0}(S_0)$ can never be smaller than $P_{\mathcal{B}}(S)$, for reasons independent of the quality of X_0's "fit". (*I.e.*, the probability assigned by the complete belief net $\mathcal{B}$ to the complete tuple $s = \langle s_0, s_1, \ldots, s_n \rangle$ is $P_{\mathcal{B}}(s) = P_{\mathcal{B}}(s_0 \mid s_1, \ldots, s_n) P_{\mathcal{B}}(s_1, \ldots, s_n) = P_{\mathcal{B}}(s_0 \mid s_1, \ldots, s_n) P_{\mathcal{B}_0}(s_{-0})$, using the observations that $\mathcal{B}_0$ and $\mathcal{B}$ will produce the same probability for any tuple that involves only $s_{-0} = \langle s_1, \ldots, s_n \rangle$. As $P_{\mathcal{B}}(s_0 \mid s_1, \ldots, s_n) \leq 1$, clearly $P_{\mathcal{B}}(s) \leq P_{\mathcal{B}_0}(s_{-0})$.)

To avoid this degeneracy, we instead first define a new structure $\mathcal{B}'$ that resembles $\mathcal{B}$ but replaces the original X_0 with a new node — call it Y — and then makes X_0' a new child of Y. For example, this would transform the left-hand structure below into the right-hand one.



The $\mathcal{B}'$ CPtables are the same as the original $\mathcal{B}$ CPtables, just substituting Y for X_0; and setting X_0''s CPtable to make $X_0' \equiv Y$. (*E.g.*, $e_{X_0'=t|Y=t} = 1$ and $e_{X_0'=t|Y=f} = 0$.) If we identify X_0' with X_0, then clearly $\mathcal{B}'$ will behave exactly like $\mathcal{B}$.

The question of whether X_0 "fits in" now reduces to the question of whether the $\mathcal{B}'$ network which includes the $Y \to X_0'$ link, is *significantly* better than the $\mathcal{B}_0'$ network, which excludes this link. We can therefore reuse the Likelihood Ratio test, described in Section 4.1, to make this decision.

Comments: (1) We earlier suggested a simple "cut-then-reconnect approach" to the task of disconnecting X_0 from $\mathcal{B}$: just remove the links connecting X_0 to other nodes, and then connect X_0's parents to each other and to X_0's children. We could not use this approach as our goal was to pull X_0 out "cleanly", in a way that maintains the same dependencies, and basic parameterization, amoung the non-X_0 nodes.

This "cut-then-reconnect" does achieve this goal when X_0 has only a single parent. Otherwise, when X_0 has more than one parent, the reconnection step (needed to maintain the parent-grandchild and parent-parent dependencies) can significantly change the parameterization: Suppose in $\mathcal{B}$, X_0 has k parents and one child, Z, which also has k other parents. For binary nodes, this would require $k + 2^k + k + 2^{k+1}$ entries. After removing X_0 and reconnecting, Z would have $2k$ parents, and so the new belief net would require $2k + 2^{2k}$

parameters, a vastly larger quantity, which means the new structure could express a much larger space of dependencies than the original $\mathcal{B}$ could.

(2) In general, we may want to consider deleting *sets* of features, rather than just a single one. While we could write a greedy incremental algorithm that considers features one-at-a-time, it is probably better to have an expert cluster together associated features (*e.g.*, all of the features associated with the electrical system, or those associated with cooling, etc.), and then consider including, versus excluding, all of this cluster of features — *i.e.*, compare the belief net that includes the complete cluster, with another that excludes it. (Of course, when this set has several parents in the remaining structure, we will need to define a set of dummy nodes.)

(3) This task is not interesting in the query-based model: If the query explicitly mentions some specific feature X_0, then we clearly cannot answer that query if we eliminate X_0 from the belief net. Alternatively, if the query does not mention X_0, then we get the same answer whether we

- first produce a new smaller belief net $\mathcal{B}'_0$ (by marginalizing out X_0) and then answer the query from this $\mathcal{B}'_0$; or
- simply answer that query from the original $\mathcal{B}$.

This is because the computation required to produce this answer, in fact, requires marginalizing out X_0.

(4) Notice this issue — of detecting and deleting "bad" features — is orthogonal to removing a tuple from the sample. As mentioned above, it is also a more dramatic step than either removing or adding a link.

As an elaboration on both of the above points: the expert might decide to exclude some features from only a *subset* of the sample — *e.g.*, those instances which really required the single fault assumption — but leave it in for the others.

5 Conclusion

Correcting the Differences: Our tools provide ways to identify the possible differences between $\mathcal{B}_E$ and I_A. So far, we have let the expert in the loop, by allowing him to decide how to fix the belief net. Alternatively, there are sometimes ways to automate this correction process. For example, we could produce a new system that would "blur" together the answers produced from $\mathcal{B}_E$ and I_A, perhaps weighting their respective answers by quantities inversely related to their respective variances (see Section 3). Alternatively, if we had the "priors" for the two source, we could instead use these as the weights, in a Bayesian manner.

Note also that many of our tools assume human guidance, *e.g.*, to suggest which specific links to consider adding or deleting. We suggested above some heuristics to help guide this search; we are currently investigating their effectiveness.

Computing Error-Bars, in General: Section 3 presents a technique for determining the "variance" around the answer returned by a belief net, as a function of the samples used to instantiate the net's CPtable. This assumes that the net's *structure* is known and fixed. We did not consider the challenge of determining the variance around those answers when the samples were also used to learn the structure, as this variance clearly depends on the particulars of the structure-learning algorithm.

5.1 Contributions

Clearly, a DSS should work as well as possible; this become crucial when considering safety-critical situations. We should therefore use as much information as possible when building and debugging the DSS. This report considers the situation where a knowledgeable, but imperfect, expert has produced a tentative system, and wants to improve it using information from another source. As that other information is also imperfect, we therefore provide the expert with a battery of tools, to help him isolate where his system and the external source differ, and to determine when those differences are significant. In particular, we provide a way to determine whether the observed differences between the responses provided by his system, and another source, are statistically significant. If so, he may want to scrutinize his belief net. We next provide some tools to help perform this investigation, by determining where the belief net and the other source differ (*i.e.*, which nodes, or which links). We also provide preliminary empirical evidence that these tools do work effectively.

These tools used a non-central χ^2-test to compute the relative likelihood of two similar belief nets, and a sensitivity analysis that provides the "error-bars" around the answers returned by a belief net, as a function of the samples used to learn this belief net. We anticipate that these techniques will have other independent applications.

References

[BD77] P. Bickel and K. Doksum. *Mathematical Statistics: Basic Ideas and Selected Topics.* Holden-Day, Inc., Oakland, 1977.

[Bee57] Y. Beers. *Introduction to the Theory of Error.* Addison-Wesley, 1957.

[BL97] A. Blum and P. Langley. Selection of relevant features and examples in machine learning. *Artificial Intelligence*, 97(1–2):245–271, 1997.

[Bun96] W. Buntine. A guide to the literature on learning probabilistic networks from data. *IEEE KDE*, 1996.

[CBL97] J. Cheng, D. Bell, and W. Liu. Learning belief networks from data: An information theory based approach. In *CIKM-97*, pages 325–331, 1997.

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

[Das97] S. Dasgupta. The sample complexity of learning fixed-structure Bayesian networks. *Machine Learning*, 29:165–180, 1997.

[FY96] N. Friedman and Z. Yakhini. On the sample complexity of learning bayesian networks. In *UAI96*, 1996.

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

[GSSK87] C. Glymour, R. Scheines, P. Spirtes, and K. Kelly. *Discovering Causal Structure*. Academic Press, Inc., London, 1987.

[HC91] E. H. Herskovits and C.F. Cooper. Algorithms for Bayesian belief-network precomputation. In *Methods of Information in Medicine*, pages 362–370, 1991.

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

[HGC95] David Heckerman, Dan Geiger, and David M. Chickering. Learning Bayesian networks: the combination of knowledge and statistical data. *Machine Learning*, 20(3):197–243, 1995.

[Hoe63] W. Hoeffding. Probability inequalities for sums of bounded random variables. *J. American Statistical Association*, 58(301), March 1963.

[Höf93] K.-U. Höffgen. Learning and robust learning of product distributions. In *COLT-93*, pages 77–83, 1993.

[KJ97] R. Kohavi and G. John. Wrappers for feature subset selection. *Artificial Intelligence*, 97(1–2), 1997.

[KL51] S. Kullback and R. A. Leibler. On information and sufficiency. *Ann. Math. Stat.*, 22:76–86, 1951.

[KT82] D. Kahneman and A. Tversky. On the study of statistical intuitions. In *Judgment under Uncertainty: Heuristics and Biases*. Cambridge University Press, 1982.

[LB94] W. Lam and F. Bacchus. Learning Bayesian belief networks: An approach based on the MDL principle. *Computation Intelligence*, 10(4):269–293, 1994.

[PDar] F. Provost and A. Danyluk. Problem definition, data clearning and evaluation: A classifier learning case study. *Informatica*, to appear.

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

[Roy57] K. Roy. A note on the asymptotic distribution of likelihood ratio. *Bull. of Calcutta Statistical Assoc.*, 7:73–77, 1957.

[SO91] A. Stuart and J. Ord. *Kendall's Advanced Theory of Statistics*, volume 2. Oxford University Press, 5th edition, 1991.

A Using the Likelihood Ratio Test

As the samples are iid, $P_b(S) = \prod_{j=1}^{|S|} P_b(X_1 = s_1^{(j)}, \ldots, X_m = s_m^{(j)})$, and so $\ln P_b(S) = \sum_{j=1}^{|S|} \ln P_b(X_1 = s_1^{j}, \ldots, X_m = s_m^{j})$. We can write $P_b(X_1 = s_1, \ldots, X_m = s_m) = \prod_{i=1}^{m} P_b(X_i = s_i \mid \mathrm{Pa}(X_i) = s_{\mathrm{Pa}(X_i)})$. Focusing on one variable X_a, let its parents in G_b be $\mathrm{Pa}(X_a) = \mathbf{W} \equiv \langle W_1, \ldots, W_k, \ldots, W_p \rangle$, where $\langle W_1, \ldots, W_k \rangle$ are its parents in G_s. Let $V' = V - \{v^*\}$, where this v^* is a distinguished value. (To simplify the description, we assume this value is common to all of the variables.) We can then parameterize the $P(X_a \mid W_1, \ldots, W_p)$ term as:

$$\begin{aligned} P_b(X_a = y \mid \mathbf{W} = \mathbf{v}) &= \\ \sum_{y \in V'} \sum_{\mathbf{v} \in V^p} &\delta(X_a, y)\delta(\mathbf{W}, \mathbf{v})\left(\phi(a, y, \mathbf{v}_{1..k}) + \theta(a, y, \mathbf{v})\right) \\ + \sum_{\mathbf{v} \in V^p} &\delta(X_a, \mathbf{v}^*)\delta(\mathbf{W}, \mathbf{v})\left(1 - \sum_{y \in V'}\left(\phi(a, y, \mathbf{v}_{1..k}) + \theta(a, y, \mathbf{v})\right)\right). \end{aligned}$$

where $\mathbf{v}_{1..k} = \langle v_1, \ldots, v_k \rangle$, and $\mathbf{v} = \mathbf{v}_{1..p} = \langle v_1, \ldots, v_p \rangle$. We constrain $\theta(a, y, \mathbf{v})$ to be zero when $v_{k+1} = \cdots = v_p = v^*$. For the other θ, their maximum likelihood estimates are simply

$$\begin{aligned} \hat{P}(X_a = y|\mathbf{W} = \mathbf{v}) &- \\ &\hat{P}(X_a = y|\mathbf{W}_{1..k} = \mathbf{v}_{1..k}, W_{k+1} = v^*, \ldots, W_p = v^*), \end{aligned}$$

where hats have been used to emphasize the fact that the probabilities are estimates, obtained from the sample (or in general, from I_A). Then

$$\frac{\partial \ln P_b(X_a = y \mid \mathbf{W} = \mathbf{v})}{\partial \theta(a, y, \mathbf{v})} = \frac{\delta(\mathbf{W}, \mathbf{v})[\delta(X_a, y) - \delta(X_a, v^*)]}{P_b(X_a = y \mid \mathbf{W} = \mathbf{v})}$$

and taking a second derivative

$$\begin{aligned} \frac{\partial^2 \ln P_b(X_a = y \mid \mathbf{W} = \mathbf{v})}{\partial \theta(a, y, \mathbf{v}) \partial \theta(a, z, \mathbf{v})} &= \\ &- \frac{\delta(\mathbf{W}, \mathbf{v})[\delta(X_a, y)\delta(X_a, z) + \delta(X_a, \mathbf{v}^*)]}{P_b(X_a = y \mid \mathbf{W} = \mathbf{v})^2}. \end{aligned}$$

Second derivative terms with respect to different $\mathbf{v} \in V^p$ are zero because of the interaction of the delta functions. Reversing the sign and taking the expectation to compute the element of M corresponding to this second derivative yields

$$\begin{aligned} M(a, y, z, \mathbf{v}) &= |S| \times \\ \sum_{X_a \in V} &P_b(X_1, \ldots, X_m)\delta(\mathbf{W}, \mathbf{v})\frac{[\delta(X_a, y)\delta(X_a, z) + \delta(X_a, v^*)]}{P_b(X_a \mid \mathbf{W} = \mathbf{v})^2} \end{aligned}$$

Note that we use the iid nature of our data S to remove the sum over draws. Of course, the complexity of evaluating the expected value is reducible by capitalizing on the belief net structure of P_b and the form of the argument.

Then the contribution to λ from X_a is

$$\sum_{\mathbf{v} \in V^p} \sum_{y \in V'} \sum_{z \in V'} M(a, y, z, \mathbf{v}) \theta(a, y, \mathbf{v}) \theta(a, z, \mathbf{v})$$

and λ is the sum of the above quantity over all variables X_a.