Bayesian Networks for Data Integration in the Absence of Foreign Keys

Bohan Zhang, Scott Sanner, and Shagun Gupta

Abstract—In the era of open data, a single data source rarely contains all of the attributes we need for inference in specific applications. For example, a marketing department may aim to integrate retailer-specific purchase data with separate demographic data for purposes of targeted advertising – a capability not possible with either dataset alone. In this work, we address two key desiderata of an automated framework for probabilistic data integration over multiple data sources: (1) we require that each relational data source share at least one attribute with another relational data source, but we do not require these attributes to be foreign keys (e.g., attributes such as gender, age, and postal code are not foreign keys because they do not uniquely identify individuals in a data source) and (2) we require inference to be probabilistic to reflect inherent uncertainty in population-level predictions given the absence of foreign keys. While some frameworks such as Probabilistic Relational Models (PRMs) address point (2), they do not address point (1) since they rely on foreign keys to link tables. To achieve both desiderata simultaneously, we develop an automated framework to construct Bayesian networks for data integration capable of answering any probabilistic query spanning the attributes of multiple relational data sources. We demonstrate that our framework is able to closely approximate the inference of a global Bayesian network over a single relation that has been projected onto multiple local relations and further investigate properties of local relations such as the number of shared attributes and their cardinality to understand how these properties affect the quality of inference.

Index Terms—Bayesian networks, probabilistic data integration

1 Introduction

Open databases provide unprecedented access to a range of data, but rarely does a single data source contain all of the attributes that we need for specific applications. For example, consider the case of targeted marketing, where a company has data on purchases, gender, and age for a set of consumers, but wants to target advertising based on consumer education level. To do this, we would like to integrate external survey or market research data containing education level into our consumer behavior model. Specifically, let us consider the case that we have two datasets to integrate: one that relates consumer behavior with gender and age, and one that relates gender and age with education level. Our ultimate goal is to predict consumer behavior from education level, which requires reasoning jointly over both data sources. However, this is not a standard data integration problem for two key reasons: First, no foreign keys link the two data sources (i.e., privacy concerns may require anonymization and gender and age do not uniquely select an entity in either relation) meaning we cannot simply perform inference by using a global schema mapping as done by [1]–[4]. (2) Second, unlike standard data integration, which focuses on instance-level inference [5], the predictions will have a high degree of uncertainty and therefore we would like to assign probabilities to predicted behavior.

In this work, we attack both problems (1) and (2) by leveraging a novel Bayesian network methodology for probabilistic reasoning over multiple data sources. While Probabilistic Relational Models (PRMs) [6], [7] have been previously proposed as a formalism for leveraging Bayesian network inference for probabilistic reasoning over databases, PRMs never explicitly focused on data integration and further, they require foreign keys for inference. In a different vein, work on Probabilistic Data Integration (PDI) [8]–[10] also focused on data integration under data uncertainty — often explicitly represented by probabilities in the data storage representation. In contrast to PDI, this work does not assume an explicit representation of probabilistic uncertainty *within* the database relations nor does it require linked data. While it is possible to discover foreign key or inclusion dependencies [11]–[13] that are not explicitly annotated in the data schema, we focus on the setting where overlapping relation attributes cannot act as (implicit) foreign keys, but can instead be characterized through probability distributions conditioned on other attributes (e.g., gender and age in our previous example, or even *partial* foreign keys such as a postal code that induces a distribution over other attributes).

The critical insight behind our method is that we can adapt existing Bayesian network structure learning methodologies to the case of data integration to build a global Bayesian network from individual local relations. We note that *if all local relations could be joined into a single global relation via foreign keys, we could simply apply standard Bayesian network structure learning methodology*. However, when we are unable to join local relations, we need to instead introduce special constraints on the structure of the Bayesian network to ensure it can be learned from the individual local relations. Once learned, this Bayesian network then permits general probabilistic queries over the attributes of *all* local relations. In our marketing example, this allows us to infer the probability of a person buying an item given that the

- *B. Zhang, S. Sanner, and S. Gupta are with the Department of Mechanical and Industrial Engineering, 27 King's College Cir, Toronto, ON M5S 3H7, Canada.*
E-mail: ssanner@mie.utoronto.ca (corresponding author)

Manuscript received XXX YY, ZZZZ; revised XXX YY, ZZZZ.

person has a Masters degree, even though the local relations with these attributes are not linked by any foreign keys.

Using the restricted search and learning methodology for Bayesian network learning over local relations without foreign keys that we contribute in this article, we demonstrate important properties of our approach: (a) Under conditions that we elucidate in the paper, it is actually possible to recover the same Bayesian network structure from local relations that we would have learned if the original global relation was explicitly given. (b) Second, in cases that do not meet the previous conditions, we empirically find that we are still able to recover Bayesian networks that provide data models and probabilistic inference comparable to Bayesian networks learned directly from the global relation.

2 Bayesian Network Preliminaries

Before we can proceed to define the Bayesian network modeling methodology in this paper, we first briefly review critical Bayesian network concepts that will be used later. The following content is explained with more detail in [14].

Model Definition: A Bayesian network provides a compact representation of a probability distribution that exploits conditional independence of all child nodes in the Bayesian network conditioned on their parents. Formally, for a set of discrete random variables $\mathbf{X} = \{X_1, \ldots, X_n\}$, a Bayesian network factorizes their joint distribution as follows:

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | Parents(X_i)), \quad (1)$$

where $Parents$ of X_i are determined according to a Directed Acyclic Graph (DAG) over $\mathbf{X}$ (see the example in Fig. 3).

Inference: Given this joint distribution, a variety of algorithms permit us to exploit the DAG stucture of the Bayesian network to efficiently infer any probability $P(\mathbf{Q}|\mathbf{E})$ given query $\mathbf{Q} \subseteq \mathbf{X}$ and evidence $\mathbf{E} \subset \mathbf{X}$, where $\mathbf{Q} \cap \mathbf{E} = \emptyset$. When all random variables are discrete, Conditional Probability Distributions (CPDs) $P(X_i|Parents(X_i))$ can be represented in a tabular form enumerating all possible combinations of variable assignments with maximum likelihood parameters estimated from their empirical distribution.

Conditional Independence: Every Bayesian network DAG implies a set of (conditional) independences among its variables. As illustrated in Fig. 1, for a Bayesian network involving three variables A, B and C with A being a shared variable, there are four possible edge orientations.

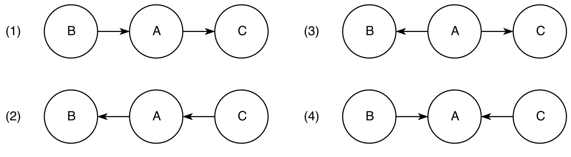

Fig. 1. All Orientations of 3 Nodes, with A Being the Shared Variable

Orientation (1), (2) and (3) are called *I-equivalent* as they represent the same independence relationship, $B \perp C|A$. That is, when A is observed, B is independent of C so C's influence cannot flow to B, and as such, we say there is no active trail (dependence) between B and C. When A is not observed, $B \not\perp C$ so one can estimate C by using B as evidence or vice versa, and we say that there is an active trail between B and C. Orientation (4) represents a different independence relationship, $B \not\perp C|A$, which is also called a "V-structure". There exists an active trail between B and C when A is observed, but the active trail is blocked when A is not observed. We will need to leverage these properties later when defining legal Bayesian networks for data integration.

Structure Learning: When learning the Bayesian network structure from a single relation over variables $\mathbf{X}$, we start with isolated nodes (no edges) and we greedily select actions to modify the Bayes net structure to maximize a structure scoring function (e.g., *K2* score [14]) while maintaining the DAG property. There are three types of legal actions we can perform when learning the edges $\mathbf{E} = \{E_1, E_2, ..., E_m\}$:

- Add a directed edge, $E_i = X_a \rightarrow X_b$ where $E_i \notin \mathbf{E}$ and $X_a, X_b \in \mathbf{X}$
- Reverse a directed edge, E_i where $E_i \in \mathbf{E}$
- Delete a directed edge, E_i where $E_i \in \mathbf{E}$

We keep performing actions that yield the highest score until we can no longer improve the overall score of the structure. Having the Bayesian network structure, we use maximum likelihood estimation as outlined previously to learn the parameters of the CPD's. We refer to this well-known Bayes net structure learning algorithm as BNLearn [14].

Since learning the structure of a Bayesian network is NP-hard, we perform Hill Climbing search using K2 score as a heuristic [14]. While other scores could be used with our approach, K2 conveniently consists of a log likelihood term under a Dirichlet prior with unit hyperpriors plus a log prior over the network structure itself, both of which can be computed using the same time and information it takes to compute the structure's maximum likelihood parameters.

3 Formal Problem Definition

In this article, we assume that we have a *global relation* (table) that we want to model. However, we are only provided with projections of that global relation, which we call *local relations* (tables). Formally, we use $R_G(\mathbf{X})$ to represent the global relation over attribute set $\mathbf{X} = \{X_1, X_2, \ldots, X_n\}$, where in a probabilistic sense, we can also think of each X_i as a random variable. Also, we use $R_{L_j}(\mathbf{X_j})$ for $j = 1 \ldots k$ to represent the jth local relation generated from $R_G(\mathbf{X})$ by projecting to the subset of local attributes $\mathbf{X_j} \subseteq \mathbf{X}$. Critically, we note that while an attribute X_i may be shared between local relations, in this paper we consider the case where X_i cannot be considered as a foreign key but rather induces a distribution over other attributes, e.g., X_i may represent gender, age, or postal code, but none of these necessarily independently selects for a unique row in any local relation.

Our objective in this paper is to find the best Bayesian network structure over attributes $\mathbf{X}$ that: (i) can be used to answer a probabilistic query $P(\mathbf{Q}|\mathbf{E})$, (ii) can be exactly learned from only the data in local relations $R_{L_j}(\mathbf{X_j})$ (i.e., tables), and (iii) optimizes the *K2* score, making it the best possible Bayesian network structure according to this metric. The key technical contribution in this article is in defining a set of constraints regarding dependences (edges) achieving (i), (ii), and (iii), that we will elucidate shortly after we discuss a motivating example for our methodology.

4 Methodology

Our goal is to define an extension of BNLearn called LR-BNLearn that learns a Bayesian network over the local relations $R_{L_j}(\mathbf{X_j})$ and permits probabilistic inference $P(\mathbf{Q}|\mathbf{E})$ over $\mathbf{Q}$ and $\mathbf{E}$ containing any attributes from the local relations. The key idea is that all attributes (random variables) shared between local relations serve as conduits that permit information sharing across the LR-BNLearn'ed Bayesian network (termed the *LR Bayesian network*) and hence should allow for effective inference of any $P(\mathbf{Q}|\mathbf{E})$. Our ultimate objective in this construction is for inference in the LR Bayesian network to match or closely approximate inference in the *Global Bayesian network* that is BNLearn'ed from the global relation.[1]

For example, as illustrated in Fig. 2, assume we have a global relation R_G over variables (relation attributes) A, B, and C projected onto two local relations $R_{L_1}(A, B)$ and $R_{L_2}(B, C)$. Assuming our goal is to query C given A as evidence, we can first learn the Bayesian network with $(B \to A)$ and $(B \to C)$ as shown for BN_1 in Fig. 2. Then the query $P(C|A)$ exploits the active trail between C and A.

4.1 Local Relation Constrained Bayes Net Learning

In the example above, we saw that learning Bayesian networks over local relations involves a search for DAGs that link variables (attributes) shared among relations. However, in this section, we observe that not all legal DAG structures can be learned from local relations, specifically shared variables cannot participate in V-structures in the Bayes net. Below, we formally define and discuss this constraint:

Definition 4.1 LR-Learnable Node: The conditional probability distribution (CPD) for a Bayesian network node X_i is LR-learnable if $\exists j \in \{1 \ldots k\}$ s.t. $\{X_i\} \cup \text{Parents}(X_i) \subseteq \mathbf{X_j}$ for some local relation R_{L_j}.

In short, because maximum likelihood CPD learning for a node requires empirical frequency counts of data over all joint assignments to the node and its parents in the Bayes net, we require all of these variables to be present in at least one local relation. LR-Learnability then implies that a shared variable cannot be at the vertex in a V-structure with parents from two different local relations. It further implies that two variables that do not appear in the same local relation cannot be connected by any edge in the Bayesian network DAG (cf. BN$_2$ in Fig. 2 and the explanation in the caption).

Definition 4.2 LR-Learnable Model: A Bayes net model is LR-Learnable if all nodes are LR-learnable.

This provides an easily checked LR-Learnability sufficiency constraint on Bayesian Network DAG learning over local relations that we use next in defining LR-BNLearn.

4.2 LR-BNLearn

In order to construct a Bayesian network DAG to answer a probabilistic query $P(\mathbf{Q}|\mathbf{E})$ w.r.t. a local relation decomposition of a global relation, we need to modify the original BNLearn algorithm from Section 2 (Structure Learning) to take into account LR-Learnability constraints during DAG structure search. We call this modified algorithm **LR-BNLearn**, which at each step of the BNLearn DAG modification process (where edges are added, reversed, or deleted) ensures that modifications are only considered if the resulting DAG is LR-Learnable w.r.t. available local relations.

TABLE 1
Global Relation

A	B	C
1	3	2
2	2	3
3	1	1
2	1	1
2	2	2

TABLE 2
Local Relation 1

A	B
1	3
2	2
3	1
2	1
2	2

TABLE 3
Local Relation 2

B	C
3	2
2	3
1	1
1	1
2	2

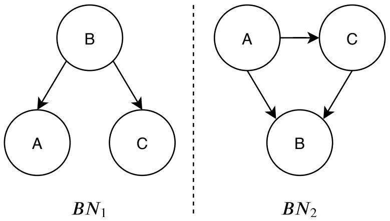

Fig. 2. An illustration of LR-Learnability for a global relation in Table 1 decomposed into two location relations in Tables 2 and 3. While the DAG for BN$_1$ (bottom left) is LR-Learnable from the local relations, BN$_2$ (bottom right) is *not* LR-Learnable because there is no local relation that allows learning of the CPD for edge $A \to C$.

Since LR-BNLearn is simply a restriction of the DAG modification search process in the existing BNLearn algorithm, the time complexity of LR-BNLearn is on the same order as BNLearn — $O(RC^2)$ for R rows, C column attributes, and assuming a constant upper bound on the number of parents of any node. Furthermore, we note that since the K2 score uses the same information required to compute the maximum likelihood parameters of the Bayesian network, K2 extends easily to LR-BNLearn, which ensures that all parameters can be learned exclusively from local relations.

5 Experiments

In our experiments, there are four key questions we want to answer in order to validate the correctness and effectiveness of our data integration framework, LR-BNLearn [2]:

1) How much do the probability distributions inferred from the LR-Learned model differ from the ground truth probabilities (e.g., in terms of absolute error or KL-divergence)?
2) If we have a known ground truth Bayesian network that is not LR-Learnable w.r.t. the given local relations, how does this impact the quality of probabilistic inference (e.g., in terms of absolute error or KL-divergence) in the LR-Learned Bayesian network compared to the ground truth Bayesian network?

[1] In practice we do not have access to the global relation since we would not need to reason over local relations if the global relation was available. However, our experimental design assumes knowledge of the global relation in order to compare inference in the global Bayesian network with the LR-BNLearn'ed Bayesian network.

[2] https://github.com/bohan-zhang/autopgm

3) How closely does the probability distribution of the LR-Learned Bayesian network approximate the ground truth probability as the quantity of data increases?
4) How does the number (and cardinality) of shared variables affect the error of probabilistic inference of models learned by LR-BNLearn in comparison to the ground truth values?

Experiments that further address the last question are reported in Appendix A and summarized in Section 5.3.4.

5.1 Data Sets

We constructed four experiments to answer the questions above, using both synthetic and real-world datasets with discrete and integer variables described in the following subsections. Each dataset is randomly divided into a training set (80%) and a test set (20%), and the training set is then projected onto local relations with different but overlapping columns.[3] LR-BNLearn treats these projected tables as different data sources and automatically trains an LR-Learned Bayesian network from these local relations. Then, the LR-Learned BN is evaluated against the test set joint distribution considered to be the "ground truth".

5.1.1 Experiment 1: Student SAT (Synthetic)

The *Student SAT* model [14], as illustrated in Fig. 3, describes the relationship between 5 variables:

- Intelligence of the student (`I`)
- Difficulty of the course (`D`)
- Grade of the student (`G`)
- Student receiving a recommendation letter (`L`)
- Student's SAT score (`S`)

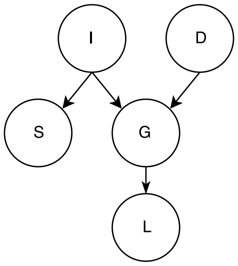

Fig. 3. Student SAT Experiment Ground Truth Bayesian Network

A synthetic dataset of $1,000,000$ rows is generated based on this model; we focus on synthetically sampled data since we aim to test if the LR-Learned Bayesian network converges to the true distribution represented by the model.

The five variables are projected into 3 local relations with the following columns: (1) `I`, `S`, (2) `G`, `L`, and (3) `I`, `D`, `G`.

5.1.2 Experiment 2: Shared Variables (Synthetic)

This experiment is designed to investigate the effect of the number of shared variables. Since real-world datasets with a large number of shared variables are largely unavailable, $1,000,000$ rows of synthetic data are generated using the Bayesian network shown in Fig. 4, with 6 random variables, and the following projection onto two local relations: (1) `A`, `B`, `C`, `D`, `E` and (2) `B`, `C`, `D`, `E`, `F`. To reduce the number of shared variables in experiments, we simply remove `B`, `C`, `D`, `E` to achieve the desired amount of sharing. Also, to investigate the behavior of learning non-BN-Learnable structure, we intentionally constructed two V-structures at node `D` and `E`.

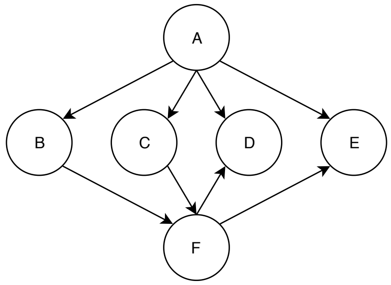

Fig. 4. Shared Variables Ground Truth Bayesian Network

5.1.3 Experiment 3: 2016 American Jobs Survey

The dataset, *2016 State of American Jobs Survey*, is acquired from *Pew Research Center* [4]. Since this is non-synthetic data, there is no ground truth Bayesian network. 5,006 data points were collected in this survey and a subset of columns are included in our experiment:

- Income level (`income` / In)
- Own or rent an apartment (`ownrent` / Ow)
- Employment status (`em` / Em)
- Financial status (`financial` / Fi)
- Level of happiness (`happy` / Ha)

These variables are projected onto two separate tables with overlapping columns, with each containing: (1) `income`, `em`, `ownrent`, and (2) `income`, `em`, `happy`, `financial`.

We perform this projection because the data collected in the local relation (1) and the local relation (2) could come from different sources in reality requiring the use of techniques introduced in this work. For example, a real-estate firm might be interested in knowing a customer's preference of renting or purchasing a property based on their financial status. In the meantime, an individual might also be motivated to know whether renting or buying an apartment would lead to an increased level of happiness.

5.1.4 Experiment 4: HackerRank Survey

HackerRank Developer Survey 2018, acquired from *Kaggle* [5], contains 25,090 responses from students and developers. Since this is non-synthetic, there is no ground truth Bayesian network. Among 50+ columns, we include the following:

[3] In this paper, we assume local relations are projections of the global relation and hence the marginal distributions of all shared variables match. In practice, however, it may be the case that local relation projections are subsampled according to different distributions, especially for open data. While it is beyond the scope of this article to address sample bias mismatch in its full generality, we note with an example in Appendix B that this sample bias *may* be addressed in certain cases without major changes to the methodology proposed here.

[4] `http://assets.pewresearch.org/wp-content/uploads/sites/3/2017/10/09160742/May16-data-release.zip`

[5] `https://www.kaggle.com/hackerrank/developer-survey-2018`

- Whether the survey taker is willing to recommend HackerRank to a friend (`recommend` / Re)
- Whether they have received a HackerRank challenge before (`hr_challenge`/ Hr)
- Age (`age` / Ag)
- Gender (`gender` / Ge)
- Student or not a student (`stu` / St)
- Degree type (`degree` / De)
- Level of education (`edu` / Ed)

These seven variables are projected onto two local relations in the following manner: (1) `age`, `gender`, `stu`, `recommend`, `hr_challenge`, and (2) `age`, `gender`, `stu`, `edu`, `degree`. The rationale is that, HackerRank, as a company, might be interested to know whether a person is likely to become a HackerRank user (`hr_challenge`) or even recommend HackerRank to their friends (`recommend`), based on their degree type or educational background. Knowing this information, HackerRank can optimize its targeted marketing campaign and thus improve its profitability.

5.2 Metrics

To evaluate the performance of the LR-Learned Bayesian networks, we use two different metrics: *KL Divergence* and *Mean Absolute Deviation* defined below.

Kullback-Leibler divergence (KL divergence), $D_{KL}(P||Q)$, measures how the LR-Learned Bayesian network's distribution Q diverges from the test data's joint distribution (ground truth) P and is defined as follows:

$$D_{KL}(P||Q) = -\sum_i P(i) \log_2 \frac{Q(i)}{P(i)}.$$

While *KL divergence* estimates the divergence between two distributions, an alternative more interpretable metric for individual query probabilities would be absolute deviation. Hence, letting $P(X = i|E)$ be the probability inferred from the LR-learned model and p_i^* be the ground truth value, we define *Mean Absolute Deviation (MAD)* as follows:

$$MAD = \frac{1}{n} \sum_{i=1}^{n} |p_i^* - P(X = i|\mathbf{E})|$$

5.3 Experimental Evaluation

5.3.1 *Performance*

Here we recall that the *Global BN* is the Bayesian network learned using BNLEARN on the global relation. We now compare inference in this Global BN to the LR-BN learned from projected local relations using LR-BNLEARN, with results shown in Table 4. Here we see that the KL divergence of the Global BN's joint distribution from the data it was learned from is very small and LR-BN does almost as well indicated by the last column showing a small % difference.

TABLE 4
KL Divergence of All Experiments

Experiment	KL (Global BN)	KL (LR-BN)	% Δ
SAT	0.000221	0.000221	0.00%
Shared Variable	0.315862	0.315862	0.00%
American Jobs	0.497224	0.494564	-0.53%
HackerRank	0.216894	0.224685	3.59%

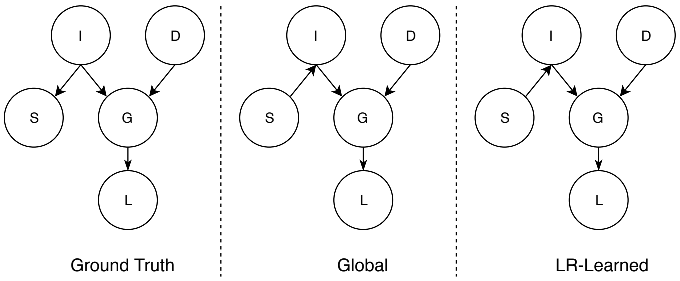

Fig. 5. Bayesian Networks, Student SAT Experiment

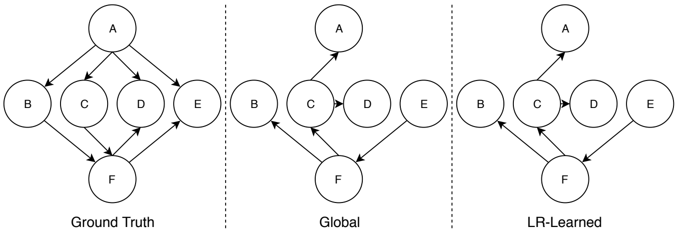

Fig. 6. Bayesian Networks, Shared Variable Experiment

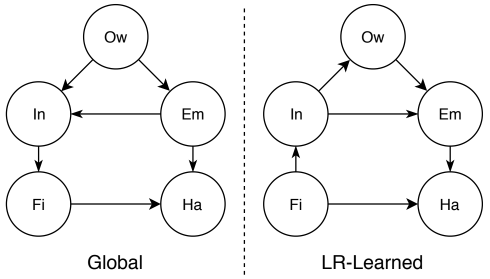

Fig. 7. Bayesian Networks, American Jobs Experiment

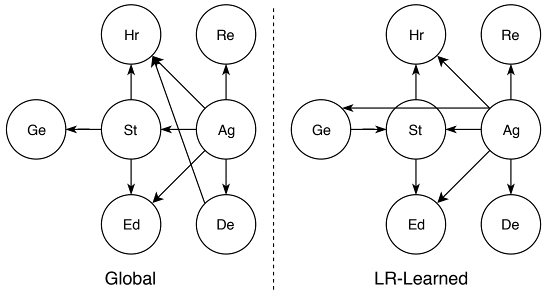

Fig. 8. Bayesian Networks, HackerRank Experiment

For *Student SAT*, the KL divergence approaches 0 (within the bound of statistical noise) since an *I-equivalent* structure to Ground Truth is recovered (Fig. 5). For *Shared Variable*, where a structure I-equivalent to Ground Truth is not LR-LEARN-able (Fig. 6), a 0 KL divergence is more difficult to achieve, but structurally matches the Global model. For *American Jobs Survey*, a significantly higher KL divergence value is obtained because there are only 5,006 data points available, and the variables have high cardinalities yielding many parameters to learn. The LR-Learned Bayesian network only differs from Global due to non-deterministic tie-breaks in (LR-)BNLearn. For *HackerRank*, we see a 3.59% difference between the LR-Learned Bayesian network and

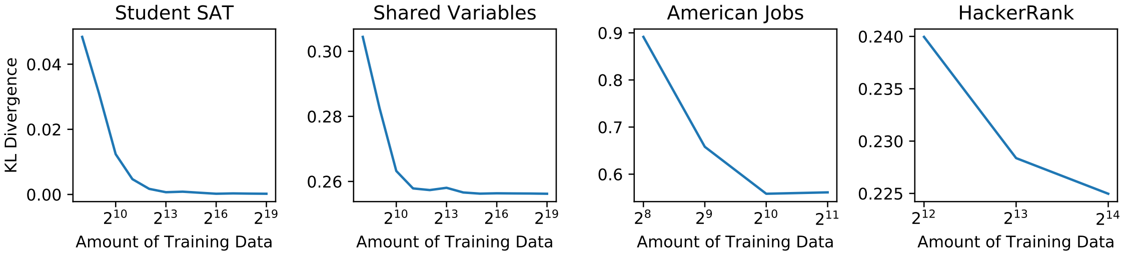

Fig. 9. KL Divergence of the LR-learned Bayesian Network from the Test Data Distribution vs. Amount of Training Data

the original Bayesian network. This is explained by Fig. 8: the edge $(De \rightarrow Hr)$ spans different local relations and cannot be recovered. Overall, we note the restricted search of LR-BNLearn yields identical or similar models to BNLearn.

5.3.2 Convergence

To verify whether the LR-Learned Bayesian network's distribution will converge to the ground truth given as the amount of data increases, we plot the KL divergence of every experiment versus the amount of training data in Fig. 9. Unsurprisingly, we see a consistent downward trend across all experiments as more training data is given, since it is easier for LR-BNLEARN to separate signal from noise and recover an accurate predictive model as data increases.

The LR-Learnable *Student SAT* KL divergence eventually reaches 0, as *I-equivalent* structures are learned and ample training data is given. On the other hand, the other three non-LR-LEARN-able datasets' KL divergences have not converged to 0 when given the full training set, because the amount of training data is insufficient and/or *I-equivalent* structures are not possible to be learned from local relations.

5.3.3 Inference when Global BN cannot be Recovered

In the *HackerRank* model, the edge $(De \rightarrow Hr)$ spans across two local relations and thus is not LR-Learnable. The unrepresented edge gives rise to a 3.59% difference between the KL divergence of the LR-BN and Global BN. In Table 5, we assess the KL divergence and MAD of queries that require this unrecovered edge for exact inference. Subscript *GT* indicates the comparison between the LR-Learned Bayesian network and the ground truth test data, and subscript *G* denotes the comparison between the LR-BN and the Global BN. In short, the KL divergence is small and the MAD only a few hundredths off from the data and Global BN estimates, indicating that unrecoverable edges in the LR-BN do not necessarily inhibit relatively accurate inference.

TABLE 5
Mean Absolute Deviation of Cross-Table Queries

Query	MAD_{GT}	KL_{GT}	MAD_G	KL_G
$P(Hr \mid De = \text{comsci})$	0.0229	0.0015	0.0243	0.0017
$P(Hr \mid De = \text{other})$	0.0639	0.0124	0.0458	0.0063
$P(De \mid Hr = \text{YES})$	0.0378	0.0130	0.0276	0.0080
$P(De \mid Hr = \text{NO})$	0.0211	0.0069	0.0195	0.0035

5.3.4 Number and Cardinality of Shared Variables

We studied the impact of the number of shared variables and the cardinality of a shared variable on queries spanning local relations with these shared variables. The models and results detailed in Appendix A show that the MAD worsens as we reduce the number of shared variables or we reduce a shared variable's cardinality. This suggests that (1) more shared variables promote increased accuracy in LR-BN inference and (2) lower cardinality shared variables limit the information that can be transmitted in cross-relation queries.

6 CONCLUSION

We proposed LR-BNLEARN, an automated framework to construct Bayesian networks that allows us to reason probabilistically over multiple relations not linked by foreign keys — a novel capability not available in previous probabilistic relational modeling frameworks. We showed that our framework is able to closely approximate inference w.r.t. ground truth reference data and models even when source relations do not permit optimal recovery of the true model.

REFERENCES

[1] A. Y. Levy, A. Rajaraman, and J. J. Ordille, "Querying heterogeneous information sources using source descriptions," in *Proceedings of VLDB*, 1996, pp. 251–262.

[2] R. Pottinger and A. Y. Levy, "A scalable algorithm for answering queries using views," in *Proceedings of VLDB*, 2000, pp. 484–495.

[3] O. M. Duschka and M. R. Genesereth, "Answering recursive queries using views," in *Proceedings of PODS*, 1997, pp. 109–116.

[4] X. Qian, "Query folding," in *Proceedings of the Twelfth International Conference on Data Engineering*, ser. ICDE '96, 1996, pp. 48–55.

[5] M. Lenzerini, "Data integration: A theoretical perspective," in *Proceedings of PODS*, 2002, pp. 233–246.

[6] L. Getoor, N. Friedman, D. Koller, A. Pfeffer, and B. Taskar, "Probabilistic Relational Models," in *Introduction to Statistical Relational Learning*, L. Getoor and B. Taskar, Eds. MIT Press, 2007.

[7] L. Getoor and L. Mihalkova, "Learning statistical models from relational data," in *Proc. of ACM SIGMOD Int. Conf. on Management of Data*. New York, NY, USA: ACM, 2011, pp. 1195–1198.

[8] N. Dalvi, C. Ré, and D. Suciu, "Probabilistic databases: Diamonds in the dirt," *Commun. ACM*, vol. 52, no. 7, pp. 86–94, Jul. 2009. [Online]. Available: http://doi.acm.org/10.1145/1538788.1538810

[9] M. Van Keulen, "Managing uncertainty: The road towards better data interoperability," *IT - Information Technology*, vol. 54, pp. 138–146, 2012.

[10] M. Magnani and D. Montesi, "A survey on uncertainty management in data integration," *J. Data and Information Quality*, vol. 2, no. 1, pp. 1–33, 2010.

[11] T. Papenbrock, S. Kruse, J. Quiané-Ruiz, and F. Naumann, "Divide & conquer-based inclusion dependency discovery," *Proceedings of VLDB*, pp. 774–785, 2015.

[12] M. Memari, S. Link, and G. Dobbie, "SQL data profiling of foreign keys," in *ER 2015 Conf. on Conceptual Modeling*, 2015, pp. 229–243.

[13] F. Tschirschnitz, T. Papenbrock, and F. Naumann, "Detecting inclusion dependencies on very many tables," *ACM Trans. Database Syst.*, vol. 42, no. 3, pp. 1–29, 2017.

[14] D. Koller and N. Friedman, *Probabilistic Graphical Models: Principles and Techniques*. MIT Press, 2009.

Supplementary Material for Bayesian Networks for Data Integration in the Absence of Foreign Keys

Bohan Zhang, Scott Sanner, and Shagun Gupta

Appendix

A. Additional Experiments

Experiment: Shared Variables (Synthetic)

This experiment is designed to investigate the effect of the number of shared variables. Since real-world datasets with a large number of shared variables are largely unavailable, $1,000,000$ rows of synthetic data are generated using the Bayesian network shown in Fig. 1, with 6 random variables, and the following projection onto two local relations: (1) A, B, C, D, E and (2) B, C, D, E, F. To reduce the number of shared variables in experiments, we simply remove B, C, D, E to achieve the desired amount of sharing.

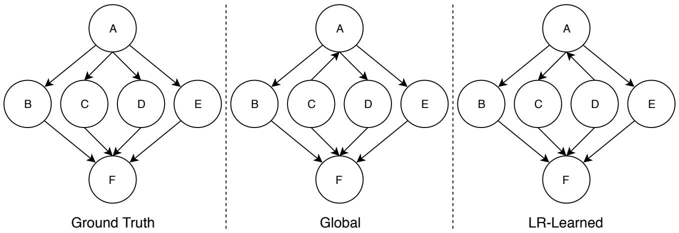

Fig. 1. Bayesian Networks, Experiment 2 (Shared Variables)

We evaluate the effect of removing shared variables in Fig. 1 by measuring the mean absolute deviation and KL divergence of two cross-table queries, $P(F|A = 0)$ and $P(F|A = 1)$. In brief, the performance worsens in both cases as we reduce the number of shared variables from 4 to 1 indicating that *more shared variables promote increased accuracy in LR-BN inference* since there are more paths (i.e., effectively more bandwidth) for information flow.

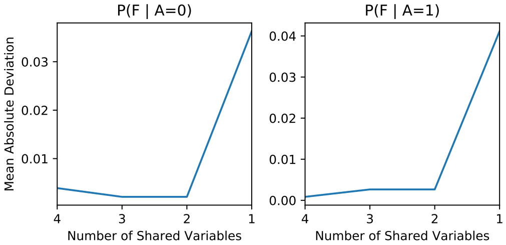

Fig. 2. Mean Absolute Deviation vs. Number of Shared Variables

Experiment: Shared Variable Cardinality (Synthetic)

In this experiment, we use a simple model, $A \rightarrow B \rightarrow C$, and $1,000,000$ rows of synthetic data, to explore the impact of shared variables' cardinality. The model is projected onto two local relations: (1) A, B and (2) B, C. Here, B is the shared variable with a cardinality of 5. During our experiment, we reduce B's cardinality by one at a time and observe the change in mean absolute deviation to determine the significance of the shared variable's cardinality.

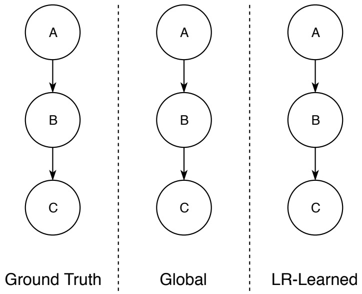

Fig. 3. Bayesian Networks, Experiment 3 (Cardinality)

The model, $A \rightarrow B \rightarrow C$, as shown in Fig. 3., is used to validate the hypothesis that accuracy will increase with shared variables' cardinality. In this model, B has a an original cardinality of 5, and we collapse its cardinality by one at a time, until it has a cardinality of 1. The mean absolute deviation of two cross-table queries, $P(C|A = 0)$ and $P(C|A = 1)$, is plotted with respect to different cardinalities in Fig. 4. Unsurprisingly, the mean absolute

- *B. Zhang, S. Sanner, and M. R. Bouadjenek are with the Department of Mechanical and Industrial Engineering, 27 King's College Cir, Toronto, ON M5S 3H7, Canada.*
  E-mail: see http://d3m.mie.utoronto.ca/

Manuscript received XXX YY, ZZZZ; revised XXX YY, ZZZZ.

deviation increases with decreasing shared-variable cardinality as suggested by the $> 12.5\%$ mean absolute deviation shown in $P(C|A=1)$. In short, fewer values in shared variables limits the information that can be transmitted through shared variables.

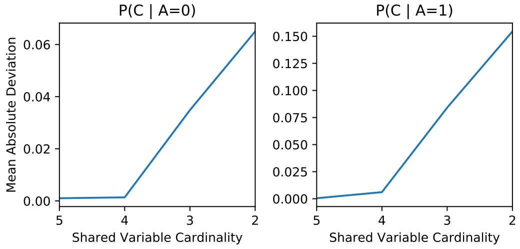

Fig. 4. Mean Absolute Deviation vs. Shared Variable's Cardinality

B. Sample Bias in Local Relations

We address here the specific case of data integration when local relations come from different distributions.

Consider a simplified version of the targeted marketing example discussed in the main article, where a company would like to target advertising based on consumer education level. Let us suppose that there are two datasets to integrate: a company dataset that relates age range $x \in X$ with consumer purchase behavior $y \in Y$ in Relation 1 (R_{L_1}) and a census dataset that relates age range $x \in X$ with education level $z \in Z$ in Relation 2 (R_{L_2}). In this case, the two datasets are drawn from different distributions and can be seen as projections of a global relation followed by a randomized subsampling procedure according to the distribution of each respective dataset.

In this Appendix, we demonstrate that when two local binary relations arise from different distributions and contain a single shared parent attribute in a Bayesian network structure, (a) one only needs to know which of the distributions represents the true (intended) prior of the shared parent attribute while learning the Bayesian network parameters, and very conveniently, (b) the distributions conditioned on this shared parent attribute remain unaffected. While this is just one case of many possible scenarios, it does suggest that there are straightforward ways to resolve issues of sample bias and mismatch in local relations within the LR-BNLearn framework proposed in this article.

Problem Setup: Given two local relations, $R_{L_1}(X,Y)$ and $R_{L_2}(X,Z)$, our goal is to query Y given Z as evidence. To this end, let us assume we wish to learn the parameters for the LR-Learnable Bayesian network with $(X \rightarrow Y)$ and $(X \rightarrow Z)$ as shown for BN_1 in Fig. 5. However, in this particular case R_{L_1} and R_{L_2} are drawn from different distributions and thus have different marginals over X:

- Table 1 (R_{L_1}) : $q(x,y) \rightarrow q(x)$
- Table 2 (R_{L_2}) : $p(x,z) \rightarrow p(x)$

How can we then learn the parameters for the CPDs of the Bayesian Network in Fig. 5?

TABLE 1
Local Relation 1

X	Y
1	0
1	0
0	1
0	0
0	1

TABLE 2
Local Relation 2

X	Z
0	1
1	0
1	0
1	0
0	1

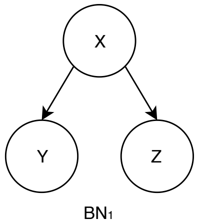

Fig. 5. An illustration showing two local relations in Tables 1 and 2 arising from different distributions. We want to learn the DAG for BN$_1$.

Solution: Suppose that we know the true marginal distribution over ages corresponding to the sample population we care about is $p(x)$ of Table 2. That is, the sample space of individuals that we are concerned with is the sample space of individuals from the census — we might consider Table 1 from a company to not be as representative of the general population distribution as Table 2 should be according to census policy. As a result, considering $q(x,y)$ to be Table 1's distribution, we want a modified version of Table 1's distribution $\tilde{q}(x,y)$ that uses the correct prior $p(x)$ of the shared variable X from Table 2:

$$q(x,y) = q(y|x)q(x) \quad \text{under Table 1's distribution}$$
$$\tilde{q}(x,y) = q(y|x)\underbrace{\tilde{q}(x)}_{p(x)} \quad \text{introduce modified } q \text{ to match Table 1's } p(x)$$

Clearly, now $\tilde{q}(x,y)$ has a marginal distribution over X of $\tilde{q}(x) = p(x)$ as intended while retaining the conditional $q(y|x)$ from $q(x,y)$:

$$\tilde{q}(x) = \sum_y \tilde{q}(x,y) = \sum_y q(y|x)p(x) = p(x)\cancelto{1}{\sum_y q(y|x)} = p(x)\,.$$

This scheme suggests the overall graphical model can be learned using empirical distributions for the following:

- $X : p(x)$
- $X \rightarrow Y : q(y|x)$
- $X \rightarrow Z : p(z|x)$

We make two key observations about this above solution:

Claim 1: Given the choice of two marginals $p(x)$ and $q(x)$, we choose the marginal $p(x)$ corresponding to the table that provides our target sampling distribution (Table 2 in this case).

Claim 2: We use the empirical $q(y|x)$ and $p(z|x)$ to estimate their respective CPDs, which is what we would have done in the original methodology if both tables had the same sampling distribution.

Another more mathematical treatment to justify the above claims would be to estimate $q(x)$ and $q(y|x)$ under an *importance sampling* correction that corrects the biased distribution $q(x)$ to $p(x)$.

A justification of Claims 1 and 2 is given below based on this importance sampling approach, but first we briefly review the importance sampling estimator for completeness. In general, given a function $f(x)$ where x has distribution p, importance sampling allows us to estimate the expectation of $f(x)$ by sampling $\sim$ from an alternate distribution q:

$$\begin{aligned}\mathbb{E}_p\big[f(x)\big] &= \mathbb{E}_q\left[\frac{f(x)\cdot p(x)}{q(x)}\right]\\ &\approx \frac{1}{n}\sum_{i=1}^{n}\frac{f(x_i)\cdot p(x_i)}{q(x_i)}, \quad x_i \sim q(x)\\ &= \frac{1}{n}\sum_{i=1}^{n} f(x_i)\cdot w_i\,,\end{aligned}$$

where $w_i = p(x_i)/q(x_i)$ is the importance sampling weight.

Importance Sampling Justification of Claims 1 and 2: We wish to learn the maximum likelihood parameters $\boldsymbol{\theta}$ for the Bayesian network edge $X \rightarrow Y$ given n data samples $\langle x_i, y_i\rangle \sim q(x,y)$, corrected via importance sampling to have marginal $p(x)$. Because marginal constraint $p(x)$ states nothing about $p(y|x)$, we will assume $p(y|x) = q(y|x)$:

$$\begin{aligned}\arg\max_{\boldsymbol{\theta}} L(\boldsymbol{\theta} : D) &= \arg\max_{\boldsymbol{\theta}} \prod_{i=1}^{n} p(x_i, y_i : \boldsymbol{\theta})\\ &= \arg\max_{\boldsymbol{\theta}} \log\left(\prod_{i=1}^{n} p(x_i, y_i : \boldsymbol{\theta})\right)\\ &= \arg\max_{\boldsymbol{\theta}} \sum_{i=1}^{n} \log\Big(p(x_i, y_i : \boldsymbol{\theta})\Big) \qquad (1)\end{aligned}$$

We can view the main expression of (1) as a Monte Carlo estimate of the following expectation:

$$\sum_{i=1}^{n} \log p(x_i, y_i : \boldsymbol{\theta}) = \mathbb{E}_{p(x,y)}\Big[\log p(x, y : \boldsymbol{\theta})\Big]$$

Next, we can apply the importance sampling correction to reweight samples $\langle x_i, y_i\rangle \sim q(x,y)$:

$$\begin{aligned}&\mathbb{E}_{q(y,x)}\left[\frac{p(y,x)}{q(y,x)}\log p(x, y : \boldsymbol{\theta})\right]\\ &= \sum_{i=1}^{n}\frac{p(y_i,x_i)}{q(y_i,x_i)}\ \log\Big(p(x_i, y_i : \boldsymbol{\theta})\Big)\\ &= \sum_{i=1}^{n}\frac{p(y_i|x_i)p(x_i)}{q(y_i|x_i)q(x_i)}\ \log\Big(p(x_i, y_i : \boldsymbol{\theta})\Big)\\ &= \sum_{i=1}^{n}\frac{p(x_i)}{q(x_i)}\ \log\Big(p(x_i, y_i : \boldsymbol{\theta})\Big)\\ &= \sum_{i=1}^{n} w_i\ \log\Big(p(x_i, y_i : \boldsymbol{\theta})\Big)\\ &= \sum_{i=1}^{n} w_i\ \log\Big(p(x_i : \boldsymbol{\theta})\cdot p(y_i \mid x_i : \boldsymbol{\theta})\Big)\\ &= \sum_{i=1}^{n}\left[w_i\ \log\Big(p(x_i : \boldsymbol{\theta})\Big) + w_i\ \log\Big(p(y_i \mid x_i : \boldsymbol{\theta})\Big)\right]\\ &= \underbrace{\sum_{i=1}^{n}\left[w_i\ \log\Big(p(x_i : \boldsymbol{\theta})\Big)\right]}_{\text{A}} + \underbrace{\sum_{i=1}^{n}\left[w_i\ \log\Big(p(y_i \mid x_i : \boldsymbol{\theta})\Big)\right]}_{\text{B}} \qquad (2)\end{aligned}$$

We see that the likelihood decomposes into two separate terms, one for estimation of the parameters of the prior distribution $p(x)$ (A) and the other for the estimation of parameters of the conditional distribution $p(y|x)$ (B). We note that because terms A and B involve disjoint parameter sets, they can be maximized separately.

For Claim 1, we consider maximization of term A from equation (2) for the prior parameters:

$$\sum_{i=1}^{n}\left[w_i\ \log\Big(p(x_i : \boldsymbol{\theta})\Big)\right]$$

Without loss of generality, we assume that X is a Bernoulli random variable taking on two values: $x_i = 1$ with probability θ_X and $x_i = 0$ with probability $1 - \theta_X$. Below, we let $\{\cdot\}$ denote the 0-1 indicator function that takes value 1 when its argument $\cdot$ is true. We also let $w^{x=j} = p(x=j)/q(x=j)$.

$$\begin{aligned}&= \sum_{i=1}^{n}\left[w_i\ \log\left(\theta_X^{\{x_i=1\}}\cdot(1-\theta_X)^{\{x_i=0\}}\right)\right]\\ &= \sum_{i=1}^{n}\Big[w_i\{x_i=1\}\log\theta_X + w_i\{x_i=0\}\log(1-\theta_X)\Big]\\ &= \log\theta_X \sum_{\{i|x_i=1\}} w^{x=1} + \ \log(1-\theta_X)\sum_{\{i|x_i=0\}} w^{x=0}\end{aligned}$$

Since the log likelihood for the exponential family is concave, we can solve for the maximizing θ_X by differentiating w.r.t. θ_X and setting it equal to 0:

$$\begin{aligned}&\Rightarrow \frac{\sum_{\{i|x_i=1\}} w^{x=1}}{\theta_X} - \frac{\sum_{\{i|x_i=0\}} w^{x=0}}{(1-\theta_X)} = 0\\ &\Rightarrow \sum_{\{i|x_i=1\}} w^{x=1}(1-\theta_X) = \sum_{\{i|x_i=0\}} w^{x=0}\theta_X\end{aligned}$$

$$\theta_X = \frac{\sum_{\{i|x_i=1\}} w^{x=1}}{\sum_{\{i|x_i=1\}} w^{x=1} + \sum_{\{i|x_i=0\}} w^{x=0}}$$

Letting $\#[\cdot]$ denote the count of data i meeting the specified criteria of its argument $\cdot$, recalling the definition of importance weight $w^{x=j}$, and recalling that the total number of samples is n, we complete the derivation:

$$\begin{aligned}\theta_X &= \frac{\frac{p(x=1)}{q(x=1)}\#[x_i=1]}{\frac{p(x=1)}{q(x=1)}\#[x_i=1] + \frac{p(x=0)}{q(x=0)})\#[x_i=0]}\\ &= \frac{\frac{p(x=1)}{q(x=1)}\frac{\#[x_i=1]}{n}}{\frac{p(x=1)}{q(x=1)}\frac{\#[x_i=1]}{n} + \frac{p(x=0)}{q(x=0)}\frac{\#[x_i=0]}{n}}\\ &= \frac{\frac{p(x=1)}{q(x=1)}q(x=1)}{\frac{p(x=1)}{q(x=1)}q(x=1) + \frac{p(x=0)}{q(x=0)}q(x=0)}\\ &= \frac{p(x=1)}{\underbrace{p(x=1)+p(x=0)}_{1}} \quad = \quad \boxed{p(x=1)}\end{aligned}$$

Here we arrive at the intuitive result of **Claim 1** that the estimate of the maximum likelihood parameters for the prior over X using data samples $\langle x_i, y_i \rangle \sim q(x,y)$ matches $p(x)$ from Table 2 when using importance sampling to correct the marginal sample bias of Table 1 to match Table 2 on their shared variable X.

We now proceed to address Claim 2. Without loss of generality, we assume X and Y are Bernoulli random variables. Let $\theta_{Y|x^0}$ ($\theta_{Y|x^1}$) be the probability of $y_i = 1$ if conditioned on $x_i = 0$ ($x_i = 1$). Decomposing term B from equation (2) in a similar manner as for Claim 1, we arrive at a different result for the maximum likelihood parameters of the importance sampling corrected conditional:

$$\begin{aligned}
&\sum_{i=1}^{n} \left[w_i \, \log \Big(p(y_i \mid x_i : \boldsymbol{\theta}) \Big) \right] \\
&= \underbrace{\sum_{\{i|x_i=0\}} \left[w_i \, \log \Big(p(y_i \mid x_i : \theta_{Y|x^0}) \Big) \right]}_{\text{C}} \\
&\quad + \underbrace{\sum_{\{i|x_i=1\}} \left[w_i \, \log \Big(p(y_i \mid x_i : \theta_{Y|x^1}) \Big) \right]}_{\text{D}} \qquad (3)
\end{aligned}$$

Now we consider only term C from (3) since the derivation for term D is identical and independently maximized:

$$\begin{aligned}
&= \sum_{\{i|x_i=0\}} \left[w_i \, \log \Big(\theta_{Y|x^0}^{\{x_i=0,y_i=1\}} (1-\theta_{Y|x^0})^{\{x_i=0,y_i=0\}} \Big) \right] \\
&= \sum_{\{i|x_i=0\}} \Big[w_i \{x_i = 0, y_i = 1\} \, \log \theta_{Y|x^0} + \\
&\qquad\qquad w_i \{x_i = 0, y_i = 0\} \, \log(1-\theta_{Y|x^0}) \Big] \\
&= \log \theta_{Y|x^0} \sum_{\{i|y_i=1,x_i=0\}} \Big[w_i \{x_i = 0, y_i = 1\} \Big] + \\
&\quad \log(1-\theta_{Y|x^0}) \sum_{\{i|y_i=0,x_i=0\}} \Big[w_i \{x_i = 0, y_i = 0\} \Big] \\
&= \log \theta_{Y|x^0} \sum_{\{i|y_i=1,x_i=0\}} [w_i] + \log(1-\theta_{Y|x^0}) \sum_{\{i|y_i=0,x_i=0\}} [w_i]
\end{aligned}$$

Differentiating w.r.t. $\theta_{Y|x^0}$ and setting it equal to 0:

$$\begin{aligned}
&\Rightarrow \frac{\sum_{\{i|y_i=1,x_i=0\}} [w_i]}{\theta_{Y|x^0}} - \frac{\sum_{\{i|y_i=0,x_i=0\}} [w_i]}{(1-\theta_{Y|x^0})} = 0 \\
&\Rightarrow \sum_{\{i|y_i=1,x_i=0\}} [w_i](1-\theta_{Y|x^0}) = \sum_{\{i|y_i=0,x_i=0\}} [w_i]\theta_{Y|x^0}
\end{aligned}$$

$$\theta_{Y|x^0} = \frac{\sum_{\{i|y_i=1,x_i=0\}} [w_i]}{\sum_{\{i|y_i=1,x_i=0\}} [w_i] + \sum_{\{i|y_i=0,x_i=0\}} [w_i]}$$

Finally, since w_i only depends on x_i, and $x_i = 0$ throughout term C, we can factor it out of the numerator and denominator and thus arrive at a fortuitous cancellation:

$$\begin{aligned}
\theta_{Y|x^0} &= \frac{\cancelto{1}{w_i} \sum_{\{i|y_i=1,x_i=0\}} 1}{\cancelto{1}{w_i} \Big(\sum_{\{i|y_i=1,x_i=0\}} 1 + \sum_{\{i|y_i=0,x_i=0\}} 1 \Big)} \\
&= \frac{\#[x_i = 0, y_i = 1]}{\#[x_i = 0, y_i = 1] + \#[x_i = 0, y_i = 0]}
\end{aligned}$$

Noting that the bottom term is just the total count $\#[x_i = 0]$ since Y is a binary variable and both $y = 1$ and $y = 0$ are considered, we can easily identify this as an empirical estimate of $q(y = 1|x = 0)$:

$$\theta_{Y|x^0} = \frac{\#[x_i = 0, y_i = 1]}{\#[x_i = 0]} = \boxed{q(y = 1|x = 0)}$$

Here we arrive at the final result of **Claim 2** that the estimate of the maximum likelihood parameters for the edge $X \rightarrow Y$ are simply the **unweighted** empirical conditional probabilities $q(y|x)$ from Table 1 data samples $\langle x_i, y_i \rangle \sim q(x,y)$ since the importance weights cancel when conditioning on X. As for $p(z|x)$ being the correct empirical conditional distribution for $X \rightarrow Z$ — this follows trivially from the fact that Table 2 is already the target sampling distribution and requires no importance sampling correction.