Scaling the weight parameters in Markov logic networks and relational logistic regression models

Felix Q. Weitkämper
ORCID 0000-0002-3895-8279

April 15, 2020

felix.weitkaemper@lmu.de
Institut für Informatik
Ludwig-Maximilians-Universität München
Bundesrepublik Deutschland

Abstract

We consider Markov logic networks and relational logistic regression as two fundamental representation formalisms in statistical relational artificial intelligence that use weighted formulas in their specification. However, Markov logic networks are based on undirected graphs, while relational logistic regression is based on directed acyclic graphs. We show that when scaling the weight parameters with the domain size, the asymptotic behaviour of a relational logistic regression model can be described by a single Bayesian network and is transparently controlled by the provided weights. We also show using two examples that this is not true for Markov logic networks. We also discuss using several examples, mainly from the literature, how the application context can help the user to decide when such scaling is appropriate and when using the raw unscaled parameters might be preferable. We highlight random sampling as a particularly promising area of application for scaled models and expound possible avenues for further research.

Keywords

Markov logic networks, Relational logistic regression, Scaling by domain size, Bayesian networks

1 Introduction

In the last 20 years, Statistical Relational Artificial Intelligence (StarAI) has developed into a promising approach for combining the reasoning skills of classic symbolic AI with the adaptivity of modern statistical AI. It is not immediately clear, however, how StarAI behaves when transitioning between domains of different sizes. This is a particularly pertinent issue for StarAI since the template-like design of statistical relational formalisms, that are presented independently of a grounding to a concrete domain, is one of their main attractions. Furthermore, scalability has proven a key barrier to their widespread deployment in applications.

The topic of scaling across domain sizes has therefore received some attention from the literature, and general patterns of behaviour have become clear. On the one hand, Jaeger and Schulte (2018) have provided very limiting necessary conditions under which domain size does not affect inference in different StarAI approaches. On the other hand, Poole et al. (2014) have characterised the scaling behaviour of both Markov Logic Networks (MLN) and Relational Logistic Regression (RLR) on a small class of formulas on which the inferences turn out to be asymptotically independent of the learned or supplied parameters. This characterisation was extended and partly corrected by Mittal et al. (2019) using considerable analytic and numerical effort. They also present a proposal to mitigate the domain-size dependence in MLN by scaling the weights associated with formulas according to the size of the domain, calling the resulting formalism Domain-size Aware Markov Logic Networks (DA-MLN). With similar computational effort, they prove that asymptotic probabilities in DA-MLN are dependent on the supplied parameters for some example cases. However, a general and systematic investigation of this dependence is still lacking.

1.1 Aims of the paper

This paper has three main objectives: First, we introduce a representation of the probability of an atom in a grounding of an MLN as the integral of a function sigmoid$(\delta^{T,n}_{R(\vec{x})}(\mathfrak{X}))$ directly derived from the weight parameters of the model. We then use this tool to evaluate the asymptotic probabilities (with increasing domain size) of two examples, and we see that neither MLN nor the scaled DA-MLN prove adequate to ensure an asymptotic behaviour that actually depends on the parameters of the model: $R(x)$ in the (DA-)MLN given by $P \Rightarrow R(x) : w$, and $Q(x)$ in the (DA-)MLN given by $P \wedge Q(x) \wedge R(x, y)$.

Then, we adapt domain-size dependent scaling of weights to a directed analogue of MLN, the RLR introduced by Kazemi et al. (2014a, 2014b), to obtain a formalism that we call “Domain-size Aware Relational Logistic Regression” (DA-RLR) in analogy to DA-MLN. We show that the weight-independent asymptotic behaviour that was exemplified above for MLN and DA-MLN does not occur in DA-RLR; in fact, we provide a representation of the asymptotic probabilities by a single Bayesian network, which we call a Proportional Relational Bayesian Network. To the best of our knowledge, this is the first StarAI formalism for which such a complete characterisation of the asymptotic probabilities from the supplied parameters is known.

Furthermore, the natural interpretation of scaled weight parameters as weighting proportions, rather than raw numbers, of influencing factors, allows us to investigate whether possible scenarios of changing domain sizes are more adequately covered by a scaled or an unscaled model. We will discuss with reference to examples of both real and toy models from the existing literature how the context of a use case can help a user to make that decision, and we will see that a particularly important case of changing domain size, that of a model trained on a random sample of a full data set, can be ideally represented by a scaled model.

1.2 Current Research on Scalability

The issues around changing domain sizes were noticed very early on in the development of Statistical Relational AI. Already in (1998), Jaeger discussed the existence of asymptotic probabilities in relational Bayesian networks (another directed networks approach), making a connection with finite model theory and infinite models of probabilistic theories. This is indeed a very interesting connection, and it would be an exciting direction for further work to investigate the implications the current work on asymptotic probabilities has for infinite models of probabilitic logics (See Section 6).

However, he did not characterise the probabilities that relational Bayesian networks would converge to, nor the conditions under which those probabilities depend on the weights. This was first explicitly isolated as a problem by Jain et al. (2010), who introduced *Adaptive Markov Logic Networks* (AMLN) as a proposed solution. This was the first suggestion to vary the weights given to the formulas of an MLN as the domain size increases. While our approach can be seen as a special case of a putative RLR-translation of AMLN, the focus of this work differs from (Jain et al., 2010) in at least two important ways. Firstly, we isolate one particular scaling function and investigate its technical properties and its asymptotic behaviour in depth. Secondly, while Jain et al. advocate learning the function from data in different domains, we suggest considering the choice of scaling a semantic problem that has as much do with how the data was obtained or is interpreted as with the data itself.

A detailed analysis of the behaviour of the uncorrected models was undertaken by Poole et al. (2014) with reference to MLN and the new framework of RLR introduced by Kasemi et al. (2014). He undertook a study of the asymptotic behaviour of MLN and proved 0-1 laws for a certain class of MLN (corrected by Mittal et al. (2019)). On the other hand, Jaeger and Schulte (2018) showed that a narrow class of MLN (corresponding to the σ-determinate MLN that define infinite models in (Singla and Domingos, 2007)) are well-behaved and indeed invariant under an increase in domain-size.

Most recently, Mittal et al. (2019) refined the classification of Poole et al. (2014) and suggested a putative solution to the problem of probabilities changing with domain size. In their model of DA-MLN, which will be formally introduced in Section 2.2, weights are changing according to a formula that relies explicitly on the number of connections a formula could possibly induce. They show that for certain combinations of MLN and queries, the asymptotic probabilities depend in a non-trivial way on the weights if they are considered as a DA-MLN, but not, if they are considered as a straightforward MLN. We will see in Section 3 in some examples that this does not generalise across DA-MLN and queries, and will suggest a reason why this is unlikely to be solved by simply changing the way the weights are recalibrated depending on the induced connections.

Instead, we believe that a transition towards directed models will overcome what we consider the main technical weakness of the DA-MLN, the aggregation function from the numbers of connections of different literals in the same formula, and we can show that when applied to the RLR approach, the scaling of the weighting with the domain size has both a very natural interpretation and guarantees weight-dependent behaviour as domain size increases.

2 Markov Logic Networks and Relational Logistic Regression

In this section we will briefly present the syntax and semantics of Markov Logic Networks and Relational Logistic Regression. Both of these approaches combine statistical with logical information, and both use weights to achieve this. However, MLN are based on *undirected* graphical models (Markov Networks) while RLR is based on *directed* graphical models (Bayesian Networks). An insightful discussion on this distinction and its importance in Statistical Relational AI can be found in (de Raedt et al., 2016).

2.1 Markov Logic Networks

As Markov Logic Networks have been extensively discussed in the literature, we will use this subsection mainly to fix some notation. A more complete discussion can be found in the original paper by Richardson and Domingos (2006) or in (de Raedt et al., 2016).

Therefore we will also restrict the definition of MLN to a setting large enough to accommodate all the examples in this work and in the papers mentioned in Section 1.2

Definition 1 Let $\mathcal{L}$ be a (potentially multi-sorted) relational signature. A *Markov Logic Network* T over $\mathcal{L}$ is a collection of pairs (φ, w) where φ is a quantifier-free $\mathcal{L}$-formula and $w \in \mathbb{R}$. We call w the *weight* of φ in T.

Markov Logic Networks have been introduced by Richardson and Domingos (2006) and have since then been highly influential in the field of Statistical Relational AI. Their semantics is based on undirected networks, which means that any literal in a formula can influence any other in a dynamic way. We can obtain such a semantics for an MLN T by choosing a (finite) domain for each sort of $\mathcal{L}$. Given such a choice, we will define the semantics as a probability distribution over all $\mathcal{L}$-structures on the chosen domains as follows:

Definition 2 Given a choice of domains for the sorts of $\mathcal{L}$, an MLN T over $\mathcal{L}$ defines a probability distribution on the possible $\mathcal{L}$-structures on the chosen domains as follows: let $\mathfrak{X}$ be an $\mathcal{L}$-structure on the given domains. Then

$$\mathcal{P}_{T,D}(X = \mathfrak{X}) = \frac{1}{Z}\exp(\sum_i w_i n_i(\mathfrak{X}))$$

where $n_i(\mathfrak{X})$ is the number of true groundings of φ_i in $\mathfrak{X}$, w_i is the weight of φ_i and Z is a normalisation constant to ensure that all probabilities sum to 1.

As the probabilities only depend on the sizes of the domains, we can also write $\mathcal{P}_{T,n}$ for domains of size n.

We refer to $\sum\limits_i w_i n_i(\mathfrak{X})$ as the *weight of* $\mathfrak{X}$ and write it as $w_T(\mathfrak{X})$.

2.2 Domain-size Aware Markov Logic Networks

Mittal et al. (2019) have introduced weight scaling to MLN in order to compensate for the effects of variable domain sizes. They call the resulting formalism *Domain-size Aware Markov Logic Networks (DA-MLN)* and we will rehearse the main definitions from their paper here.

Definition 3 A *Domain-size Aware Markov Logic Network (DA-MLN)* is given by the same data as a regular MLN (see Definition 1).

In order to adapt the semantic to changing domain size, Mittal et al. (2019) use the concept of a *connection vector.*

Definition 4 Let φ be a formula. Let Ψ be the set of literals of φ and for every $\psi \in \Psi$, let V_ψ be the set of free variables in φ not occurring in ψ. For every variable x, let D_x signify its domain. Then the *connection vector* of φ is the set $\{\prod\limits_{x \in V_\psi} |D_x| | \psi \in \Psi\}$.

The connection vector records how many tuples each literal could possibly connect to; in the formula $P(x) \wedge Q(x,y) \wedge R(z)$, for instance, the literal $P(x)$ could connect to $|D_y| * |D_z|$ many tuples, the literal $Q(x,y)$ could connect to $|D_z|$ many tuples and the literal $R(z)$ could connect to $|D_x| * |D_z|$ many tuples.

The problem now is to aggregate this information to a single scaling factor. Mittal et al. (2019) use the maximum of the entries of the connection vector, but suggest investigating other options as well (see Subsection 3.4 below).

Definition 5 Given a choice of domains for the sorts of $\mathcal{L}$, a DA-MLN T over $\mathcal{L}$ defines a probability distribution on the possible $\mathcal{L}$-structures on the chosen domains as follows: let $\mathfrak{X}$ be an $\mathcal{L}$-structure on the given domains. Then

$$\mathcal{P}_{T,D}(X = \mathfrak{X}) = \frac{1}{Z}\exp(\sum_i \frac{w_i}{C_{i,D}} n_i(\mathfrak{X}))$$

where $n_i(\mathfrak{X})$ is the number of true groundings of φ_i in $\mathfrak{X}$, w_i is the weight of φ_i, $C_{i,D}$ is the maximum of the entries of the connection vector of φ_i, and Z is a normalisation constant to ensure that all probabilities sum to 1.

Mittal et al. (2019) give several examples of how moving from ordinary to DA-MLN changes the asymptotic behaviour; we will see further examples in Section 3 below.

2.3 Relational Logistic Regression

Relational Logistic Regression differs from MLN in that it is based on directed rather than undirected models. Thus rather than allowing arbitrary weighted formulas, one first designates a *Relational Belief Network (RBN).* This is a directed acyclic graph whose nodes are labelled with atoms from $\mathcal{L}$. An RLR-model over this RBN consists of an additional labelling of each node as follows:

Definition 6 A *Relational Logistic Regression* T over a relational signature $\mathcal{L}$ consists of a directed acyclic graph G whose nodes O are labelled with a pair $(\varphi, (V, \psi, w)_i)$, where φ is an $\mathcal{L}$-atom and $(V, \psi, w)_i$ a set of triples such that V_i is a finite set of variable symbols, ψ_i a quantifier-free formula and $w_i \in \mathbb{R}$. Furthermore, the following are required to hold:

1. Every $\mathcal{L}$-atom appears as the first label of exactly one node.
2. None of the variable symbols appearing in φ are in a V_i

3. ψ_i is a boolean combination of the first labels of the parent nodes of O, and every variable in ψ_i appears either in φ or is in V_i. If O is a leaf, then there is only 1 triplet, with $\psi =$ "$x = x$" for a variable x.

While the semantics of the MLN in Definition 2 was given as a single probability distribution, we will give the semantics of the RLR by induction over the underlying RBN. More precisely, we will use induction over the longest path from a leaf to a given node, which we will call the *index* of a node. To facilitate writing, we will introduce some notation before defining the semantics of an RLR:

Definition 7 The sigmoid function is defined by $\mathrm{sigmoid}(k) := \frac{\exp(k)}{\exp(k)+1}$. We will furthermore write $\vec{a} \in D$ for a tuple from D (rather than $\vec{a} \in D^n$) and use the expression 1_ψ for the function that returns 1 whenever ψ holds and 0 otherwise. Finally, we will use the convention of denoting with $\psi(\vec{a}/\vec{x})$ the grounding of ψ where $\vec{a}$ is substiuted for the variables $\vec{x}$.

Let $\mathcal{L}_n$ be defined as the subsignature of $\mathcal{L}$ consisting of all those relations that label a node of index $\leq n$. Then we will define a probability distribution on the possible possible $\mathcal{L}_n$-structures on a given domain D for $\mathcal{L}$ by induction over n and as the product of the probabilities for every grounding of the atom:

$n = 0$: The probability of any given grounding of the atom in $\mathcal{L}_0$ at node O is given by $\mathrm{sigmoid}(w)$, the probability of its negation thus by $1 - \mathrm{sigmoid}(w)$.

Assume now that a probability distribution on $\mathcal{L}_n$-structures has been defined. We will now it to $\mathcal{L}_{n+1}$-structures as follows: The probability of an $\mathcal{L}_{n+1}$-structure is given by the probability of its $\mathcal{L}_n$-reduct multiplied with the conditional probability of the groundings of the atoms in $\mathcal{L}_{n+1} \backslash \mathcal{L}_n$. These are given by

$$\mathcal{P}(q(\vec{x})) = \mathrm{sigmoid}(\sum_i w_i \sum_{\vec{a} \in D} 1_{\psi_i(\vec{a}/V_i))})$$

Note that this latter expression only depends on the $\mathcal{L}_n$-reduct because of the conditions on the ψ_i from the definition of an RLR above.

For a short and informal exposition to MLN and RLR and their relationship to each other, see Kazemi et al. (2014b).

3 Asymptotic probabilities in MLN and DA-MLN

In this section we will build on the work of Mittal et al. (2019) and give two examples of dependencies for which an asymptotic behaviour that depends on the weight is achieved neither in MLN nor in DA-MLN. In order to give a clear and rigorous structure to these derivations, we will first introduce probability kernels for Markov logic and prove some basic facts about their behaviour under limits.

3.1 Characterising probabilities in MLN and DA-MLN

In order to determine the probability of a certain atomic formula being true, we compute the probability measure of the set of all worlds in which this formula holds. This can also be conveniently written as an integral

$$\mathcal{P}(R(\vec{x})) = \int\limits_{\mu_{T,n}} 1_{R(\vec{x})}$$

We would like to replace the indicator function in the integral with a function that we can express analytically in terms of the weights of the MLN model. We can achieve that by considering the weighted mean of the indicator functions between two models that only differ in the value of $R(\vec{x})$.

Let $\mathfrak{X}$ be a structure and let $\vec{x} \in \mathfrak{X}$ be a tuple. Then let $\mathfrak{X}_{R(\vec{x})}$ be the structure that potentially differs from $\mathfrak{X}$ only in that $R(\vec{x})$ holds in $\mathfrak{X}_{R(\vec{x})}$, regardless of whether it holds in $\mathfrak{X}$. $\mathfrak{X}_{\neg R(\vec{x})}$ is defined analogously. In this way the class of all structures of a given domain size divide equally in structures of the form $\mathfrak{X}_{R(\vec{x})}$ and structures of the form $\mathfrak{X}_{\neg R(\vec{x})}$, and there is a natural one-to-one correspondence betwen structures of each type. Therefore we can compute the integral above by instead considering the weighted mean of the indicator function on both $\mathfrak{X}_{R(\vec{x})}$and $\mathfrak{X}_{\neg R(\vec{x})}$. This can be described in terms of an auxiliary function that reflects the dependence of the weights on the validity of $R(\vec{x})$:

Definition 8 Let $\mathfrak{X}$ be a structure of domain size n and let T be a (DA-)MLN. Then $\delta^{T,n}_{R(\vec{x})}(\mathfrak{X}) := \sum\limits_i w_i n_i(\mathfrak{X}_{R(\vec{x})}) - \sum\limits_i w_i n_i(\mathfrak{X}_{\neg R(\vec{x})})$.

The weighted mean of $1_{R(\vec{x})}$ on $\mathfrak{X}_{R(\vec{x})}$ and $\mathfrak{X}_{\neg R(\vec{x})}$ can now be computed as follows:

$$\frac{1\cdot\mu(\mathfrak{X}_{R(\vec{x})})}{\mu(\mathfrak{X}_{R(\vec{x})})+\mu(\mathfrak{X}_{\neg R(\vec{x})})} = \frac{1\cdot\left[\frac{\mu(\mathfrak{X}_{R(\vec{x})})}{\mu(\mathfrak{X}_{\neg R(\vec{x})})}\right]}{\left[\frac{\mu(\mathfrak{X}_{R(\vec{x})})}{\mu(\mathfrak{X}_{\neg R(\vec{x})})}\right]+1} = \frac{\left[\frac{\exp(\sum\limits_i w_i n_i(\mathfrak{X}_{R(\vec{x})}))}{\exp(\sum\limits_i w_i n_i(\mathfrak{X}_{\neg R(\vec{x})}))}\right]}{\left[\frac{\exp(\sum\limits_i w_i n_i(\mathfrak{X}_{R(\vec{x})}))}{\exp(\sum\limits_i w_i n_i(\mathfrak{X}_{\neg R(\vec{x})}))}\right]+1} =$$

$$= \frac{\exp(\delta^{T,n}_{R(\vec{x})}(\mathfrak{X}))}{\exp(\delta^{T,n}_{R(\vec{x})}(\mathfrak{X}))+1} = \mathrm{sigmoid}(\delta^{T,n}_{R(\vec{x})}(\mathfrak{X}))$$

and therefore we can express the probability of $R(\vec{x})$ as an integral over δ, which is directly connected to the weights in the (DA-)MLN:

$$\mathcal{P}(R(\vec{x})) = \int\limits_{\mu_{T,n}} \mathrm{sigmoid}(\delta^{T,n}_{R(\vec{x})})$$

Before moving on to concrete calculations, we will briefly record a lemma which we will need at several points:

Lemma 1 *Let φ be a formula with at least one free variable. If, for all $\vec{x} \in D$ for any D of any size, $\mathcal{P}(\varphi(\vec{x})) \geq k \in (0,1)$, then for all $N \in \mathbb{N}$ and every $\varepsilon > 0$ there is an $n \in \mathbb{N}$ such that for any domain D larger than n the probability that a structure has less than N tuples $\vec{x}$ with $\varphi(\vec{x})$ is less than ε.*

Proof This is an immediate consequence of the weak law of large numbers, see e. g. (Bauer, 1996).

Later we will use a much sharper version of the same idea in our exact treatment of asymptotic probabilities in domain-size aware directed models.

3.2 The formulaP $\Rightarrow R(x)$

The formula $P \Rightarrow R(x)$ is the converse of $R(x) \Rightarrow P$, possibly the most studied formula in discussions on scaling and asymptotic probability. Poole et al. (2014) give a detailed analysis of the behaviour of P in this configuration, but we will focus here on the asymptotic behaviour of $R(x)$.

Proposition 1 *Let T_w be the MLN given by the formula $P \Rightarrow R(x) : w$, $w > 0$, $|\Delta_x| = n$. Then the asymptotic probability of P is 0 and the asymptotic probability of $R(x)$ is $\frac{1}{2}$, independent of the value of w.*

Proof We will use our results from the preceding subsection.

$$\mathcal{P}_{T_w,n}(R(x)) = \int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(\delta^{T_w,n}_{R(x)}) = \int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(w\cdot 1_P) \leq \int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(w) = \mathrm{sigmoid}(w)$$

Thus, Lemma 1 holds and for all $N \in \mathbb{N}$ there is an $n \in \mathbb{N}$ such that $\mathcal{P}(|\neg R(\mathfrak{X})| > N) \geq \frac{1}{2}$. Therefore we see that, for sufficiently large n, $\int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(-w\cdot|\neg R(\mathfrak{X})|) \leq \frac{1}{2}\mathrm{sigmoid}(-w\cdot N)$, which converges to 0 as n grows to infinity. Thus,

$$\lim_{n\to\infty}\mathcal{P}_{T_w,n}(P) = \lim_{n\to\infty}\int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(\delta^{T_w,n}_{P}) = \lim_{n\to\infty}\int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(-w\cdot|\neg R(\mathfrak{X})|) = 0$$

This proves the first part of the proposition. We will now use this to prove the second part. Since the asymptotic probability of P is 0, there is for every $\varepsilon > 0$ an $N \in \mathbb{N}$ such that $\mathcal{P}_{T_w,n}(P) < \varepsilon$. Thus

$$\lim_{n\to\infty}\mathcal{P}_{T_w,n}(R(x)) = \lim_{n\to\infty}\int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(\delta^{T_w,n}_{R(x)}) = \lim_{n\to\infty}\int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(w\cdot 1_P) = \mathrm{sigmoid}(0) = \frac{1}{2}$$

As $P \Rightarrow R(x)$ is logically equivalent to $\neg R(x) \Rightarrow \neg P$, we obtain analogous results for $R(x) \Rightarrow P$:

Corollary 1 *Let T_w be the MLN given by the formula $R(x) \Rightarrow P : w$, $w > 0$, $|\Delta_x| = |\Delta_y| = n$. Then the asymptotic probability of $R(x)$ is $\frac{1}{2}$ and the asymptotic probability of P is 1, independent of the value of w.*

Of course, it is well known that MLN do not generally behave well asymptotically. Usually, however, this phenomenon has been observed in atoms that have an unbounded number of connections themselves. Here, the issue stems from the fact that the atom with which it is connected has an unbounded number of connections and therefore degenerates.

Since moving to DA-MLN will regulate the asymptotic behaviour of P, one might expect that this will also make the probability of P weight-dependent. However, unfortunately, scaling the weights with the domain size will itself impact $R(x)$ in the same way:

Proposition 2 *Let T_w be the DA-MLN given by the formula $P \Rightarrow R(x) : w$, $w > 0$, $|\Delta_x| = n$. Then the asymptotic probability of $R(x)$ is $\frac{1}{2}$, independent of the value of w.*

Proof The connection vector for the formula $P \Rightarrow R(x)$ is $(n, 1)$, and so weights will be scaled using the factor n.

$$\lim_{n\to\infty} \mathcal{P}_{T_w,n}(R(x)) = \lim_{n\to\infty} \int\limits_{\mu_{T_w,n}} \text{sigmoid}(\delta^{T_w,n}_{R(x)}) = \lim_{n\to\infty} \int\limits_{\mu_{T_w,n}} \text{sigmoid}(\frac{w}{n} \cdot 1_P)$$
$$\leq \lim_{n\to\infty} \text{sigmoid}(\frac{w}{n}) = \text{sigmoid}(0) = \frac{1}{2}$$

Again, we obtain a corollary on $R(x) \Rightarrow P : w$.

Proposition 3 *Let T_w be the DA-MLN given by the formula $R(x) \Rightarrow P : w$, $w > 0$, $|\Delta_x| = n$. Then the asymptotic probability of $R(x)$ is $\frac{1}{2}$, independent of the value of w.*

3.3 The formula $P \wedge Q(x) \wedge R(x, y)$

In this section we will discuss the formula $P \wedge Q(x) \wedge R(x, y)$ as an example in which DA-MLN "overcompensate" for the amount of connections of one literal, namely $Q(x)$.

In ordinary MLN, the asymptotic probability of $Q(x)$ is 1, regardless of the weight.

Proposition 4 *Let T_w be the MLN given by the formula $P \wedge Q(x) \wedge R(x, y) : w$, $w > 0$, $|\Delta_x| = |\Delta_y| = n$. Then the asymptotic probability of $Q(x)$ is 1, independent of the value of w.*

Proof First observe that the probability of P will be at least $\frac{1}{2}$:

$$\mathcal{P}_{T_w,n}(P) = \int\limits_{\mu_{T_w,n}} \text{sigmoid}(\delta^{T_w,n}_P) = \int\limits_{\mu_{T_w,n}} \text{sigmoid}(w \cdot |Q(x) \wedge R(x, y)|) \geq \text{sigmoid}(0) = \frac{1}{2}$$

We will now establish that, for any fixed x, the probability of $P \wedge R(x, y)$ is always at least $\frac{1}{4}$. This is a consequence of the Bayesian formula, which implies that $P \wedge R(x, y) = \mathcal{P}_{T_w,n}(P) * \mathcal{P}_{T_w,n}(R(x, y)|P)$. Since the analysis of Subsection 3.1 is just as valid for conditional probabilities, we can derive

$$\mathcal{P}_{T_w,n}(R(x, y)|P) = \frac{\int_{\mu_{T_w,n}} \text{sigmoid}(\delta^{T_w,n}_{R(x,y)}) * 1_P}{\mathcal{P}_{T_w,n}(P)} = \frac{\int_{\mu_{T_w,n}} \text{sigmoid}(w \cdot 1_{Q(x)}) * 1_P}{\mathcal{P}_{T_w,n}(P)} \geq$$
$$\geq \frac{\int_{\mu_{T_w,n}} \text{sigmoid}(0) * 1_P}{\mathcal{P}_{T_w,n}(P)} = \frac{\frac{1}{2}\mathcal{P}_{T_w,n}(P)}{\mathcal{P}_{T_w,n}(P)} = \frac{1}{2}$$

and therefore $\mathcal{P}_{T_w,n}(P \wedge R(x, y)) \geq \frac{1}{2} * \frac{1}{2} = \frac{1}{4}$. Thus we can once again apply Lemma 1 and conclude that, asymptotically, there will be arbitrarily many pairs (x, y) with $P \wedge R(x, y)$ with arbitrarily high probability. As in the proof of Proposition 1,

$$\lim_{n\to\infty} \mathcal{P}_{T_w,n}(Q(x)) = \lim_{n\to\infty} \int\limits_{\mu_{T_w,n}} \text{sigmoid}(\delta^{T_w,n}_{Q(x)}) = \lim_{n\to\infty} \int\limits_{\mu_{T_w,n}} \text{sigmoid}(w \cdot |P \wedge R(x, \mathfrak{X})|) = 1$$

In DA-MLN, the situation is reversed, and in fact, the probability of $Q(x)$ will always tend to $\frac{1}{2}$:

Proposition 5 *Let T_w be the DA-MLN given by the formula $P \wedge Q(x) \wedge R(x, y) : w$, $w > 0$, $|\Delta_x| = |\Delta_y| = n$. Then the asymptotic probability of $Q(x)$ is $\frac{1}{2}$, independent of the value of w.*

Proof The connection vector for the formula $P \wedge Q(x) \wedge R(x, y)$ is $(n^2, n, 1)$, and so weights will be scaled using the factor n^2.

$$\lim_{n\to\infty} \mathcal{P}_{T_w,n}(Q(x)) = \lim_{n\to\infty} \int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(\delta^{T_w,n}_{Q(x)}) = \lim_{n\to\infty} \int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(\frac{w}{n^2} \cdot |P \wedge R(x, \mathfrak{X})|) \leq$$

$$\leq \lim_{n\to\infty} \int\limits_{\mu_{T_w,n}} \mathrm{sigmoid}(\frac{w}{n^2} \cdot n) = \lim_{n\to\infty} \mathrm{sigmoid}(\frac{w}{n}) = \mathrm{sigmoid}(0) = \frac{1}{2}$$

3.4 Discussion

When introducing DA-MLN, Mittal et al. (2019) use the aggregation function max as a pragmatic choice with good formal properties that also proved to work well in practice. However, they also point out that investigating different choices could be an avenue for further research, specifically mentioning the function sum. Therefore, we would like to discuss briefly to what extent the issues raised in this section depend on the precise aggregation function used.

It seems clear that, since the asymptotic behaviour is degenerate in those cases in which the δ limits to either 0 or ∞, the order of the scaling coefficient is more important than the precise number. When using the maximum function as an aggregation function, the order will be the same as the highest order among the entries of the connection vector. This scaling will be unsuitable though for investigating the behaviour of the other literals in the formula, since the scaling will overcompensate for the number of connections. This is exactly what happens in the examples discussed here. Therefore, changing the aggregation function to summation would rather exacerbate than mitigate the issues, since it would then be scaled down even further.

Instead, it might seem plausible to use a concept of mean. Since we are dealing with multiplicative scaling, the arithmetic mean would not be a natural choice, and would be of the same order as the maximum. Instead, one might try the geometric mean as an aggregation function. This will not regulate the asymptotic behaviour of the literals at the extremes - in fact, if one considers the standard example of $R(x) \Rightarrow P : w$, one would overcompensate for the connections of the $R(x)$ and undercompensate for the connections of the P. Therefore, the geometric mean would be far from theoretically optimal, and in fact, since the orders of the connection numbers of the literals are different, no single aggregation function will be adequate for all literals. However, the convergence to the degenerate probabilities would be slowed for all literals, and this might be more relevant in practical applications.

We will now continue along a different line, and switch our representation formalism from MLN and undirected models, where all literals influence each other and connections will need to be aggregated, to RLRs and directed models, where only a single literal is being influenced and one can scale directly along the possible connections of that single literal.

4 Domain-size-Aware RLR and Proportional Relational Belief Networks

We have seen in the preceding sections that while DA-MLN enhance the dependence of limit behaviour on the weights of the formulas, there are cases where they cannot solve the issue with regards to all queries. More precisely, we see that the issue comes from the aggregation function and the need to choose a single weight for formulas whose literals have different numbers of connections. While we have also seen that MLN can have weight-independent limit behaviour even when there is only at most one connection from the literal concerned, the example given is dependent on the other literal in the formula having infinitely many connections and thus limiting to probability 1.

4.1 Definition of Domain-size-Aware RLR

In this section, we will adapt the principle of DA-MLN to the weighted directed model approach of RLR, and we will see that since there is only one literal that is directly affected by any connection (the child literal of the edge), we can avoid the problem of aggregating a connection vector and can scale directly by the domain sizes of the free variables in the corresponding variable set. Formally, we define:

Definition 9 A *DA-RLR T is given by the same data as an RLR (see Definition 6). However, the semantics differ as follows from the semantics given in Definition 7:*

Let D be a multi-sorted domain and let D_x be the sort of a variable x. For any set of variable symbols V let $|D|_V := \prod_{x \in V} |D_x|$.

Then the induction step of Definition 7 is replaced as follows:

The probability of an $\mathcal{L}_{n+1}$-structure is given by the probability of its $\mathcal{L}_n$-reduct multiplied with the probability of the groundings of the atoms in $\mathcal{L}_{n+1} \backslash \mathcal{L}_n$. These are given by

$$\mathcal{P}(q(\vec{x})) = \text{sigmoid}(\sum_i \frac{w_i}{|D|_{V_i}} \sum_{\vec{a} \in D} 1_{\psi_i(\vec{a}/V_i))})$$

The definition now explicitly depends on the domain sizes just as the definition of DA-MLN depended on domain sizes. However, the number $|D|_{V_i}$ in the denominator now represents the possible domains of connection for just the atom $q(\vec{x})$ - since this is the only child of that edge relation, there is no connection vector and no aggregation function. For this representation, we can prove that any cluster point of the probability of any formula will vary with the weights chosen as domain sizes approach infinity. Furthermore, we can describe the limit behaviour in terms of a special variant of relational belief networks, which we call Proportional Relational Belief Network (PRBN).

4.2 Interpretation of scaled weights and asymptotic behaviour

There is a very intuitive interpretation of the $\frac{w_i}{|D|_{V_i}} \sum_{\vec{a} \in D} 1_{\psi_i(\vec{a}/V_i))}$ from Definition 9, as this expression is clearly equivalent to $w_i \frac{\sum_{\vec{a} \in D} 1_{\psi_i(\vec{a}/V_i))}}{|D|_{V_i}}$. The latter expression shows that this is just the weighted proportion of tuples for which ψ_i holds. Therefore, a DA-RLR has two types of conditions - those depending on proportions (of formulas with free variables) and those depending just on a Boolean true/false value (of propositions or formulas made up from relations and constants only). We will see in this section that asymptotically, these behave very differently. To see why, consider the language consisting of two relations P and R, where P is a 0-ary proposition and R a unary predicate. Now consider the independent distribution, where both P and $R(x)$ for any x have probability $\frac{1}{2}$. Now consider a domain size limiting to infinity. In this scenario, the probability of P will still be $\frac{1}{2}$ - in half of the worlds it will be true, in half of the worlds it will not. The same goes for $R(x_0)$ for an element x_0 that is contained in every domain. Contrast this with the proportion of x for which R holds. Asymptotically, the probability to choose a sequence of domains in which this proportion limits to $\frac{1}{2}$ has probability 1. This is exactly the statement of the strong law of large numbers (see e. g. Chapter III in (Bauer 1996)) . In other words, it is a sure event - the proportion will definitely be asymptotically $\frac{1}{2}$, but you have no idea whether in a given large domain from your sequence P will hold. That is random and therefore not sure at all.

4.3 RBN that are polytrees

As we will discuss in the next section, the representation of this in a Bayesian Network is technically much more involved when dealing with loops in the Relational Belief Network. Therefore, we assume in this section that the RBN has no such loops. Pearl (1988) has coined the term *polytree* for such a network, as loops also add additional difficulty in evaluating belief propagation in general Bayesian networks.

Therefore, when constructing an appropriate graphical representation for the limit behaviour of DA-RLR, we will to have adapt the calculation of probabilities and the mechanism of belief propagation in Bayesian networks to reflect this distinction.

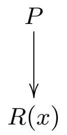
Definition 10 A polytree is called a *Proportional Relational Belief Network (PRBN)* over a language $\mathcal{L}$ if it is a relational belief network over the language $\mathcal{L}_P$ consisting of the signature of $\mathcal{L}$ enriched with sufficiently many *generic constants* for any arity on any domain occurring in a relation symbol in $\mathcal{L}$, with the following property:

If $R(\vec{x}, y)$ is an atom of $\mathcal{L}_P$ and y_0 is an appropriate generic constant, then the parents of the node $R(\vec{x}, y_0)$ are obtained by substituting y_0 for y in all the parents of $R(\vec{x}, y)$.

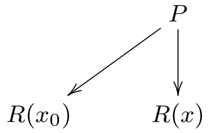
Remark 1 The property in Definition 10 uniquely determines a PRBN extending a given RBN over the language $\mathcal{L}$.

Let's consider the examples of Subsections 3.2 and 3.3.

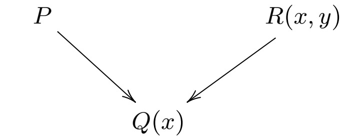
Example 1 As an RLR, the formula of Subsection 3.2 could be written as the RBN



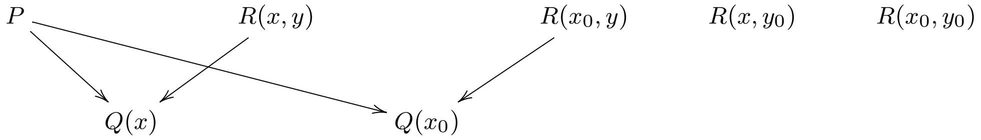
with the formula $P : w$ attached to the $R(x)$ node. Since substituting x_0 for x into P leaves P unchanged, the associated PRBN is



Example 2 There are different configurations for representing $P \wedge Q(x) \wedge R(x, y)$ as an RLR. Since we are interested in the behaviour of $Q(x)$, we will consider this to be the dependent formula, leaving us with the formula $P \wedge R(x, y) : w$ for $Q(x)$ and this RBN:



Creating the associated PRBN involves making substitutions of generic constants at every point. $Q(x_0)$ will have parents P and $R(x_0, y)$:



Remark 2 The resulting PRBN is a polytree if and only if the RBN is.

We will build a semantics for the asymptotic probabilities induced by a DA-RLR based on the corresponding PRBN, first informally and then formally as Theorem 1. As suggested at the beginning of this section, we will consider closed formulas as binary variables just as they are used in the semantics of RLR for a fixed n, with probabilistic dependencies on the values of its parent nodes. Formulas with free variables, in contrast, are considered as proportions, which are in principle continuous variables taking values between 0 and 1. In the limit case that we are modelling, however, the law of large numbers tells us that only a finite number of discrete states are taken, and that precisely which one it will be is a deterministic consequence of its parent states (it can thus rather be considered discrete than continuous, though usually not binary). The rules for computing either the proportion of an open formula, or the probability of a closed formula being true can be read off directly from the formulas and weights associated with the DA-RLR.

Theorem 1 *The asymptotic validity of closed atoms and the proportion of tuples for which non-closed atoms hold in a DA-RLR T whose underlying RBN is a (possibly disconnected) polytree are represented by the following Bayesian Network:*

1. *The underlying DAG is the unique PRBN obtained from the RBN of T.*
2. *The value of a child atom is derived from the weighted formulas of the DA-RLR:*
   (a) *If the child atom q is closed, then it is a boolean variable, and the probability of it being true is given by*

$$\mathcal{P}(q) = \mathrm{sigmoid}(w + \sum_{i \in I} w_i * 1_{\psi_i} + \sum_{j \in J} w_j * \mathrm{Prop}(\psi_j))$$

where I is the set of closed parent atoms and J is the set of non-closed parent atoms; the formulas and their weights are taken from the data of the DA-RLR, where generic constants are just replaced by their respective variables.

(b) *If the child atom* $q(\vec{x})$ *is not closed, then it is a discrete variable* $\mathrm{Prop}(q(\vec{x}))$ *whose values are a deterministic function of the values of its parents:*

$$\mathrm{Prop}(q(\vec{x})) = \mathcal{P}(q(\vec{x_0})|\mathbf{parents})$$

3. *The probability that a closed formula is true is derived from its atoms as the boolean combination of boolean variables.*
4. *The value of a formula that is not closed is a deterministic function of the values of the ancestors of its atoms:*

$$\mathrm{Prop}(\varphi(\vec{x})) = \mathcal{P}(\varphi(\vec{x})|\mathbf{atoms})$$

Remark 3 At first glance, it might seem that the definition of the value of $\mathrm{Prop}(q(\vec{x}))$ could be circular since it invokes a probability computed in the same network. However, observe that the index of $q(\vec{x})$ is the same as its generic instantiations. So, since the value of a node does not depend on the rules for computing the values of any node of the same or higher index than itself, we are not required to use the rule given in clause 2b) in order to evaluate it.

This theorem formalises the discussion that we had in the preceding paragraph. We will prove it by induction on the index.

Proof By induction on the index.

Index 0: An atom of index 0 has no parents, and therefore its probability is just given as $\mathcal{P}(q(x_0)) = \mathrm{sigmoid}(w)$. This coincides with the statement of the theorem in the closed atom case. In the case of an atom with free variables, we obtain a classic Bernoulli chain with probability sigmoid(w). For such a chain, the strong law of large numbers holds and tells us that with probability 1, the proportion of the population for which $q(x)$ holds is exactly the probability of $q(x_0)$ for almost all $x_0 \in D$.

So assume the statement true for all nodes of index $\leq n$ and let the index of a node be $n+1$. If the atom at this node is closed, then 2a) is just a reformulation of the probability in the definition of a DA-RLR. So let $q(\vec{x})$ be an atom with free variables. Then, reasoning as we did before, the law of large numbers shows that $\mathrm{Prop}(q(\vec{x}))$ is just the probability of $q(\vec{x_0})$ for a generic $\vec{x_0}$. This depends on the frequencies and probabilities of possibly boolean combinations of parents of $q(\vec{x})$. By the induction hypothesis, these are represented by the PRBN. Because the RBN and thus the PRBN is a causal polytree, however, all parents are independent (since $q(\vec{x})$ blocks the only connection between them). Thus those frequencies and probabilities can be computed purely from the values of the parent nodes themselves, and thus conditioning on the parents is sufficient to compute the true probability of $q(\vec{x_0})$.

This theorem provides a closed Bayesian Network interpretation of the asymptotic probability which directly depends on the weights provided in the model. It therefore gives a general positive answer for loop-free domain-size aware directed models to the question that has been discussed for DA-MLN by Mittal [2019a] and above: whether the asymptotic probabilities of formulas vary with the weights given in the model.

We will conclude this subsection by annotating the PRBNs from the examples above with their semantic values:

Example 3 The PRBN from Example 1 will take values as follows: P is a Boolean random variable with probability 0.5 of being true, $R(x_0)$ is a Boolean random variable which is true with probability sigmoid(w) if P holds and with probability sigmoid$(0) = 0.5$ otherwise, and $R(x)$ is a proportional variable which could take two values: If P holds, it is bound to take the value sigmoid(w), and if P does not hold, it will take the value 0.5.

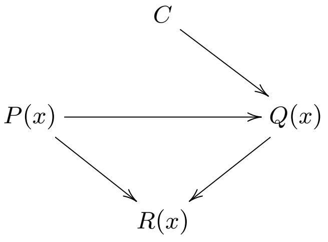
The PRBN from Example 2 looks more complex, but the calculation of the probabilities and proportions proceeds in the same manner: P is a Boolean random variable, true with probability 0.5. $R(x,y)$, $R(x_0,y)$ and $R(x,y_0)$ are proportional variables that have the fixed value 0.5 and $R(x_0,y_0)$ is a Boolean random variable that is true with probability 0.5. $Q(x_0)$ is another Boolean Random Variable, which is true with probability $\mathrm{sigmoid}(w{*}\mathrm{Prop}(R(x_0,y)) = \mathrm{sigmoid}(0.5w)$ if P holds and with probability sigmoid$(0) = 0.5$ otherwise. Finally, $Q(x)$ is a proportional random variable with two possible values: sigmoid$(0.5w)$, which it takes if P holds, and 0.5, which it takes if P does not hold.

4.4 The general case - coping with loops

This section extends the representation of asymptotic probabilities by Bayesian networks to the case of an RBN that might contain loops.

The problem with having loops is that the parents of a node are no longer necessarily independent. Therefore, it is no longer generally possible to compute the proportion of individuals that satisfy a certain boolean combination of properties from the proportions that satisfy each property individually. An example of this would be the following DA-RLR:

Example 4 Consider a DA-RLR based on the following RBN,

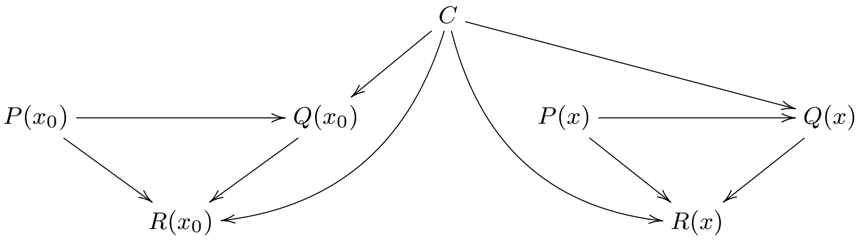


where $R(x)$ has the formula $P(x) \wedge Q(x) : w$ and $Q(x)$ has the formula $C \wedge P(x) : w$ and $\neg C \wedge \neg P(x) : w$.

Since $P(\vec{x})$ has asymptotic proportion exactly 0.5, $Q(\vec{x})$ has the same asymptotic proportion, regardless of whether C holds or not. The proportion of elements for which $Q(\vec{x}) \wedge P(\vec{x})$ and thus the proportion of elements for which $R(x)$ hold, however, depends essentially on C.

Therefore, we can no longer assume that the probability of $R(\vec{x_0})$ will depend only on the immediate parents of $R(\vec{x})$. Rather, it will depend on every factor that might influence a parent of $R(\vec{x})$. Those factors are exactly the ancestors of the node $R(\vec{x})$. This means that we have to add edges from all the ancestors of a node of the RBN as parents to a child node. Then we construct the PRBN as we did before. In the example above, the PRBN would be this:

Example 5



The formal definition of a general PRBN can therefore be given as follows:

Definition 11 A DAG which is not a polytree is a *Proportional Relational Belief Network (PRBN)* over a language $\mathcal{L}$ if it is a relational belief network over the language $\mathcal{L}_P$ consisting of the signature of $\mathcal{L}$ enriched with sufficiently many *generic constants* for any arity on any domain occurring in a relation symbol in $\mathcal{L}$, with the following properties:

1. If $R(\vec{x}, y)$ is an atom of $\mathcal{L}_P$ and y_0 is an appropriate generic constant, then the parents of the node $R(\vec{x}, y_0)$ are obtained by substituting y_0 for y in all the parents of $R(\vec{x}, y)$.
2. Every ancestor of a node is also a parent of the node.

Given this definition, the computation of probabilities and proportions can be formalised in a very similar manner to Theorem 1:

Theorem 2 *The asymptotic validity of closed formulas and the proportion of tuples for which non-closed formulas hold in a general DA-RLR T are represented by the following Bayesian Network:*

1. *The underlying DAG is the unique PRBN obtained from the RBN of T.*
2. *The value of a child atom is derived from the weighted formulas of the DA-RLR:*

(a) If the child atom q is closed, then it is a boolean variable, and the probability of it being true is given by

$$\mathcal{P}(q) = \text{sigmoid}(w + \sum_{i \in I} w_i * 1_{\psi_i} + \sum_{j \in J} w_j * \text{Prop}(\psi_j))$$

where I is the set of closed formulas and J is the set of non-closed formulas taken from the data of the RLR. Weights are assigned just as in the definition, and generic constants are just replaced by their respective variables.

(b) If the child atom $q(\vec{x})$ is not closed, then it is a discrete variable $\text{Prop}(q(\vec{x}))$ *whose values are a deterministic function of the values of its ancestors:*

$$\text{Prop}(q(\vec{x})) = \mathcal{P}(q(\vec{x_0})|\textbf{ancestors})$$

3. The probability that a closed formula is true is derived from its atoms as the boolean combination of boolean variables.

4. The value of a formula that is not closed is a deterministic function of the values of the ancestors of its atoms:

$$\text{Prop}(\varphi(\vec{x})) = \mathcal{P}(\varphi(\vec{x_0})|\textbf{ancestors of atoms})$$

Proof The proof is the same as for Theorem 1; since we are conditioning on all ancestors rather than just on the parents of a node, we do not have to assume that the atoms are independent to compute the probability of $\varphi(\vec{x_0})$.

5 Discussion and Mixed RLR

Jain et al. (2010), who were the first to discuss adapting the weights to the population size, advocated learning the size-to-weight function from datasets of different sizes. In this section, we will use the interpretation we have given of scaled weights as proportions (rather than absolute numbers of incidences) to argue that the semantics of the use case can give an indication as to whether scaling is appropriate or not.

5.1 Interpretation of scaling

The key is that we have to assess whether the dependency we stipulate by assigning a weight to a formula shows a dependence on absolute numbers of an associated event or a dependence on the proportion of possible events that satisfy the criteria.

To see the difference, imagine the following scenario: we would like to model how much a student has learned depending on whether the teaching he has received has been good or poor. Say that we decide to use a vocabulary consisting of two domains whose individuals are “lessons” and “students” and then two predicates, a unary predicate “good lesson” $G(x)$ ranging over the first domain and a unary predicate “learning success” $L(y)$ ranging over the second. We could then frame this question in different ways. First, assume that we would like to evaluate the *effectiveness* of the teaching - has a student learnt sufficiently much *considering the amount of time he spent being taught.* In this case, it seems reasonable to assume that this will depend on the *proportion* of lessons that have been good. Whether a student is in education for 12 years or 9 years, we would still believe that the the effectiveness of teaching depends on its quality. However, we could also evaluate the sheer amount of learning a student has received - then $L(x)$ might represent a certain fixed skill level, like “can read and write”. Now it would not seem sensible to use a proportionalist model: instead, it seems rather reasonable to believe that in the limit of more and more lessons, the student will eventually have attended enough good ones to have learnt the skill.

Bearing this distinction in mind, we will evaluate how our findings relate to the different scenarios either discussed in the literature (Kazemi et al., 2014) or used as evaluation datasets (Mittal et al., 2019).

5.2 A closer look at the examples from Kazemi et al. (2014)

In their introduction, Kazemi et al. (2014) list a number of different ways in which changing population sizes might be relevant in an AI model. In the first scenario, elaborated in (Poole, 2003), the likelihood of someone having committed a crime is dependent on how many other people fit the description of the criminal. One way to implement this would be as follows: One domain of suspects, a unary predicate $C(x)$

for being a criminal and a unary predicate $D(x)$ for matching the description of the criminal. We then have a constant j for Joe. The RBN could be as given below,

$$\begin{tikzcd} C(x) \arrow[r] \arrow[d] & C(a) \arrow[d] \\ D(x) & D(a) \end{tikzcd}$$

and the formulas involved would be atomic with a positive weight attached to $C(x)$ and $C(a)$ when evaluating $D(x)$ and $D(a)$ repectively (as well as a base weighting to reflect the inherent likelihood of $D(x)$ independent of the crime) and a negative weight given to $C(x)$ when evaluating $C(a)$.

Note that we need the constant for Joe since a general model, making everyone less likely to be the criminal given that another person is the criminal, would introduce a cycle into the RBN, which is generally disallowed.

In this model, scaling of weights could only be considered at the edge $C(x)$ to $C(a)$, as this is the only one with parent variables that are not in the child. However, here scaling is counterproductive: The edge is intended to model that there is likely to be just one criminal, and that if there is one there is unlikely to be another. So here the model as it stands seems well equipped to deal with the issue of varying domain size without any scaling. If the are more people that are a priori equally likely to have committed the crime, then the likelihood for Joe having committed it is smaller. Kazemi et al. (2014) introduce a different kind of different domain size: they say that it is arbitrary which population we base our model on, whether it is the neighbourhood, the city or the whole country. However, we think that which scale to use should not be arbitrary, since any model will rely on the assumption that everyone is equally likely to have committed the crime. In a gang stabbing, the population should be restricted to the gang membership; in an internet-based case of credit card fraud, one might have to consider the whole world. Another option of dealing with this would be to have an arbitrary population but then adapt the description predicate $D(x)$ to include "lives in the area of the crime". In that case, this weight should indeed be scaled by population size, at least if we assume that the population we choose definitely encompasses at least everyone living in the area of the crime. this is, in a sense, the opposite scaling of what has been suggested here: We are scaling precisely to make the likelihood of $D(x)$ limit to 0 independent of the chosen weights since we are intending to condition on $D(a)$ anyway when evaluating the RBN later. While this is a very interesting phenomenon, exploring its technical background is outside the scope of the present work.

In the second example, Kazemi et al. (2014) mention a situation in which the population is variable, such as the population of a neighbourhood or of a school class. This is much like the example we have discussed above in relation to learning success: While the number of lessons a student takes varies between students, how we want to deal with this situation depends on exactly what sort of question we are asking.

5.3 Mixed RLR

These examples show that the decision to use or not to use proportional scaling in a model depends on exactly the sort of questions we are asking of the model and also why exacly the population is varying in the first place. It might also very well be appropriate to use proportional scaling on some of the connections but not on others. Consider for instance an application that models pollution in a lake. It has two domains, one for tributaries to the lake and one for human users of the lake. The signature has two unary predicates, $R(x)$ signifying that the water from tributary x is polluted, and $H(y)$, meaning that human y pollutes the lake, as well as a proposition P meaning that the lake is polluted. If we were to assume that pollution in the lake depends on the *proportion* of incoming water that is polluted and the *amount* of pollution added to that by humans, we could mark the formula involving $R(x)$ as proportional while keeping the formula involving $H(y)$ as absolute. Such *mixed RLR* could thus be useful to model situations in which connections have different quality, and could simply be represented by marking some formulas to be proportional and some not. As the overall probabilities are obtained from the individual weights using the logistic regression, one can simply adapt some of the individual weights as the domain size changes, while leaving others unchanged.

5.4 Random Sampling

Given this discussion, the question arises why the DA-MLN have been so successful in the tests that Mittal et al. (2019) have conducted on their training datasets. It seems unlikely that the queries they have asked all happen to be the ones that depend on proportions and that proportional scaling is the right approach for.

This brings us to what we consider the greatest practical application for fully proportional DA-RLR: training on randomly sampled subpopulations.

When Mittal et al. (2019) set up the tests on the data sets from friends and smokers, the IMDB and the Web Knowledge Base, they have created smaller sample domains by randomly selecting a subdomain of the entire database. In that case, numbers will tend to be decreased to scale, and thus what is approximately conserved is exactly the proportion of domain elements that satisfy a given relation. By passing from RLR to DA-RLR, we move from considering absolute numbers of parent atoms (which are heavily distorted by the random sampling) to proportions, which should be approximately conserved. For a concrete example, let us consider the model of teaching and learning from Section . Assume we had decided to consider absolute learning success, which usually would not suggest scaling by domain size. However, we can only observe a limited sample of lessons, and how many that is varies from school to school. Now if we were to estimate learning success here, it is natural to make it dependent on the *proportion* and not the *number* of good lessons that we are seeing.

Random sampling is prevalent throughout the natural and social sciences, however, and for instance the drug study example of Kazemi et al. (2014) also falls under this category. Moreover, random sampling could be used as a tool to reduce the size of the training dataset required.

6 Potential connections and possible further work

This work throws up several questions of both theoretical and practical interest.

6.1 Theoretical work

In the study of 0-1 laws and the existence of asymptotic probabilities in first-order theories, finite model theorists have established a close connection between infinite models of first-order theories and asymptotic probabilities in finite models (see e. g. (Fagin, 1976) or (Lynch, 1985)). In the realm of Markov Logic, one can see a similar connection between the condition of σ-determinacy for the existence of an infinite model of an MLN in (Singla and Domingos, 2007) and the condition for projectivity (which assures independence of probabilites from domain size) in (Jaeger and Schulte, 2018). Since DA-RLR provide a class of directed models with controlled asymptotic behaviour and no syntactic restriction, it would be intruguing to see whether one could use this to construct a concept of probabilistic logic with infinite models that captures proportional thinking as a foundation for probabilities.

Of course, the directed models that we use here have limitations. Most importantly, they cannot deal with cycles in the Relational Belief Network, which occur, for instance, already in the famous "Friends and Smokers" example. This contrasts with approaches based on Logic Programming, such as ProbLog, that can deal with certain cyclical programs since they are based on a Least Herbrand semantics (for a more in-depth discussion, see (de Raedt et al., 2016)). It would therefore be of great interest to port the concepts and results of this work to a logic programming context such as Problog. Another way to deal with cycles in the RBN is manually to prevent them from inducing cycles in a grounding. One way for formalising this is given by Getoor et al. in Section 5.2.5.3 of (Getoor and Taskar, 2007), who discuss colouring edges of the RBN to control dependencies in the grounding. It would be very interesting to see how this carries over to the PRBN as we have introduced it here.

Another extension to this framework which would be very useful would be incorporating multi-valued or even continuous variables. This extension has already been posited as "further work" by Kazemi et al. (2014) and would significantly enhance the applicability of our methods. We suspect that since our PRBN already includes non-binary nodes an extension of DA-RLR to multi-valued discrete RLR might not be too difficult, while encompassing continuous variables could be technically more challenging.

Finally, one could explore other types of "conditional scaling" with domain size that conserve conditional probabilities while allowing certain unconditional probabilities to limit to extremes. An example of this is the stuation in Subsection 5.2. There, the probability of a given individual having committed a crime should limit to 0 as the population we consider increases. Then, the conditional probability of Joe having committed the crime given that he satisfies the description of the suspect will converge dependending on the weights provided in the model. There is additional interest in exploring those options since arbitrary MLN can be encoded as conditional probabilities in directed models (see Section 4.3 in (de Raedt et al., 2016)), and therefore the limitations in the weight scaling method for MLN discussed in Section 3 should be reflected in limitations on capturing asymptotic behaviour for conditional probabilities.

6.2 Applied work

A first step to evaluating the practical efficacy of the DA-RLR formalism developed here would be to test it on the acyclical benchmarks used by Mittal et al. (2019), the IMDB and the WebKB. By the discussion we had in Subsection 5.4, we would expect the performance of DA-RLR to exceed the performance of RLR at least the same factor as the performance of the DA-MLN exceeded that of MLN. In fact, the performance should be almost constant with domain size. However, there would be great interest in seeing whether the perceived greater flexibility of using undirected MLN might outweigh the benefits of better scaling, and whether further theoretical advances could mitigate such downsides.

Practical application with real or simulated data would give evidential weight to the discussion in Section 5 and would be a big step towards establishing and delineating practical use cases for proportional reasoning within the wide spectrum of applications of Statistical Relational AI. In particular, finding applications in which mixed DA-RLR, which incorporate both scaled and non-scaled weights, outperform both pure RLR and pure DA-RLR would be very interesting.

Also of particular interest is the connection to random sampling discussed in Subsection 5.4. By reducing the size of the training dataset, random sampling might help overcome the high computational intensity of Statistical Relational Learning and thereby broaden its general applicability. It would thus be very interesting to formalise the connection with random sampling and to evaluate the performance of DA-RLR by training it on small randomly sampled subsets of known datasets from the StarAI literature.

Of course, a prerequisite for successful application of DA-RLR in practice are good learning algorithms for the underlying RLR formalism. A very promising approach has been developed and extensively tested by Ramanan et al. (2018), and that paper also contains datasets and benchmarks that can be used or adapted for the purposes of evaluating DA-RLR in practice.

6.3 Beyond DA-RLR

We believe that the approach we develop in Section 3 has the potential to significantly simplify the further asymptotic analysis of MLN and DA-MLN by moving it away from the heavily numerical computation exemplified by the Supplementary Material to Mittal et al. (2019) and towards a more transparent and rigorous probability-theoretic treatment. If this could be further systematicised, we believe it has the potential to deliver a single coherent treatment of asymptotic probabilities in (DA-)MLN, which will still be of considerable interest given the attractiveness of undirected models in representation formalisms. It also provides a direct link to the Gibbs measures used by Singla and Domingos when defining infinite models of MLN, suggesting that here too a connection between asymptotic probabilities and infinite models could be harnessed for the mutual benefit of both.

References

1. Bauer, H. (1996) *Probability theory.* De Gruyter Studies in Mathematics, vol. 23. Berlin: Walter de Gruyter & Co.
2. Fagin, R. (1976) Probabilities on finite models. *Journal of Symbolic Logic,* 41(1):50-58.
3. Georgii, H.-O. (2011) *Gibbs measures and phase transitions.* De Gruyter Studies in Mathematics, vol. 9, 2nd edition. Berlin: Walter de Gruyter & Co.
4. Getoor, I., Taskar, B. (2007) *Introduction to statistical relational learning.* Cambridge, MA: The MIT Press.
5. Jaeger, M. (1998) Convergence Results for Relational Bayesian Networks. In Thirteenth Annual {IEEE} *Symposium on Logic in Computer Science, Indianapolis, Indiana, USA, June 21-24, 1998.* IEEE.
6. Jaeger, M., Schulte, O. (2018) Inference, learning and population size: Projectivity for SRL models. *CoRR* abs/1807.00564.
7. Jain, D., Barthels, A., Beetz, M. (2010) Adaptive Markov logic networks: learning statistical relational models with dynamic parameters. In Coelho, H., Studer, R., Wooldridge, M. J. (Eds.) *ECAI 2010 - 19th European Conference on Artificial Intelligence, Lisbon, Portugal, August 16-20, 2010.* Frontiers in Artificial Intelligence and Applications, vol. 215:937-942. Amsterdam: IOS Press.
8. Kazemi, S. M., Buchman, D., Kersting, K., Natarajan, S., Poole, D. (2014a) Relational Logistic Regression. In Baral, C., Giacomo G. D., Eiter, T. (Eds.) *Principles of knowledge representation and reasoning: Proceedings of the Fourteenth International Conference, Vienna, Austria, July 20-24, 2014.* Palo Alto, CA: AAAI Press.

9. Kazemi, S. M., Buchman, D., Kersting, K., Natarajan, S., Poole, D. (2014b) Relational Logistic Regression: the directed analog of Markov logic networks. *AAAI Workshop on Statistical Relational Artificial Intelligence 2014, July 27-28, Quebec, Canada.* Palo Alto, CA: AAAI press.
10. Lynch, J. F. (1985) Probabilities of first-order sentences about unary functions. *Transactions of the American Mathematical Society,* 287(2):543-568.
11. Mittal, H., Bhardwaj, A., Gogate, V., Singla, P. (2019) Domain-size aware Markov logic networks. In: Chaudhuri, K., Sugiyama, M. (Eds.) *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, Naha, Japan, 16-18 April 2019.* Proceedings of machine learning research , vol. 89:3216-3224.
12. Pearl, J. (1989) *Probabilistic reasoning in intelligent systems - networks of plausible inference.* Morgan Kaufmann series in representation and reasoning. Burlington, MA: Morgan Kaufmann.
13. Poole, D. (2003) First-order probabilistic inference. In: Gottlob, G., Walsh, T. (Eds.) *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 9-15, 2003,* 985-991. Burlington, MA: Morgan Kaufmann.
14. Poole, D., Buchman, D., Kazemi, S. M., Kersting, K., Natarajan, S. (2014) Population size extrapolation in relational probabilistic modelling. In: Straccia, U., Cali, A. (Eds.) *Scalable Uncertainty Management - 8th International Conference, Oxford, UK, September 15-17, 2014.* Lecture Notes in Computer Science vol. 8720:292-305. Berlin: Springer.
15. de Raedt, L., Kersting, K., Natarajan, S., Poole, D. (2016) *Statistical relational artificial intelligence: logic, probability and computation.* Synthesis lectures on artificial intelligence and machine learning. Morgan & Claypool.
16. Ramanan, N., Kunapuli, G., Khot, T., Fatemi, B., Kazemi, S. M., Poole, D. et al. (2018) Structure learning for relational logistic regression: an ensemble approach. In: Thielscher, M., Toni, F., Wolter, F. (Eds.) *Principles of Knowledge Representation and Reasoning: Proceedings of the sixteenth international conference, Tempe, USA, October 30 - November 2, 2018.* 661-670. Palo Alto, CA: AAAI press.
17. Richardson, M. & Domingos, P. M. (2006) Markov logic networks. *Machine Learning* 62(1-2):107-132.