Обобщение Правил Взаимодействия в Вероятностной Логике

Arjen Hommersom и Peter J. F. Lucas

Институт вычислительных и информационных наук

Университет Немейгена

Немейген, Нидерланды

{arjenh,peterl}@cs.ru.nl

Перевод с английского: Подтынный С.Д.

Аннотация

В последние два десятилетия наблюдается появление многих различных вероятностных логик, использующих логические языки для определения, а иногда и рассуждения, с помощью вероятностных распределений. Вероятностные логика, что поддерживает рассуждения с помощью вероятностных распределений, таких как ProbLog, использует неявное определение из правила взаимодействия, чтобы объединить вероятностные доказательства об атомах. В этой статье мы покажем, что это правило взаимодействия является примером более общего класса взаимодействий, которые могут быть описаны немонотонной логикой. Кроме того, мы покажем, что такие локальные взаимодействия о вероятности атома могут быть описаны с помощью свертки. Полученная расширенная вероятностная логика поддерживает немонотонные рассуждения с вероятностной информацией.

1 Введение

В последние два десятилетия мы стали свидетелями появления множества различных вероятностных логик, такие как логика Маркова [Richardson and Domingos, 2006], вероятностная логика дизъюнкта Хорна [Poole,1993], и ProbLog [Kimmig et al., 2010], который использует логические языки для определения, а иногда и рассуждения, с помощью вероятностного распределения. Вероятностные логики базируются на логике первого порядка, и они разделяют преимущество логики первого порядка в сравнении с логикой высказываний в том, что они могут рассматриваться как более выразительные языки представления знаний со значительной мощностью моделирования, которое для вероятностных логик распространяется на те области, где сталкиваются неопределенности.

Тем не менее, несмотря на их общности, все эти языки основаны на различных семантических принципах. Некоторые из принципов, за исключением само собой разумеющиеся, фактически должны быть рассмотрены как часть проектных решений. Когда различные решения были сделаны, разичные вероятностные логики появились из-за тесного взаимодействия между логическими аспектами и вероятностными аспектами вероятностной логики.

В исследовании, описанном в этой статье, мы вводим теоретические основы целого класса вероятностной логики, начиная от очень общей и неограниченной вероятностной логики - вышеупомянутый ProbLog, и показать, что из нее может получиться новая, но связанная, вероятностная логика с помощью изменения способа вероятностного доказательства, скомбинированного между собой. Мы делаем это с помощью реорганизации вероятностного и логического базисов вероятностной логики, так что в результате логика может поддерживать двоичный, логический и вероятностный вид. Полученный класс вероятностных логик имеет преимущество в том, что его можно использовать как орудие логики для решения прикладных задач.

В следующем разделе мы сначала рассмотрим основы ProbLog, который сопровождается развитием основных методов, используемых в обобщении вероятностно-логического взаимодействия, и, наконец, кратко обсуждается изначальная логика. Мы будем использовать вероятностно-логические взаимодействия, такие как смысловой и общий, алгебраических способ объединения неопределенных доказательств, в то время как по умолчанию логика будет использоваться в нашем языке, чтобы осуществлять взаимодействие операторов, снова отражающиих эту двойную перспективу вероятностной логики. Новая вероятностно-логическая структура описана в разделе 3, и ее сравнение с другими подходами в разделе 4. Достижения этого исследования отражены в разделе 5.

2 Вступление

В этом разделе описыпается вероятностная логика и метод объединения вероятностного доказательства. Мы также кратко опишем понятия из изначальной логики.

2.1 Вероятностная логика: синтаксис и семантика

В научной литературе существует много различных определений вероятностной логики. В этой статье мы сосредоточимся на классе логик, где вероятностные и логические рассуждения идут рука об руку.ProbLog является типичным примером такого языка [Kimming et al,, 2010]. ProbLog был специально разработан в качестве основного языка, на котором могли быть скомпилированы программы на других языках вероятностной логики [De Raedt et al., 2008]; Таким образом результаты этой статьи относятся к целому ряду языков.

Предполагается, что читатель знаком с терминологией логического программирования, где программа содержит дизъюнкт Хорна в виде:

$$B<-A_1,\dots,A_n.$$

где B, A_i - *атомы* в виде $p(t_1,\dots,t_m)$ с t_k количеством термов. Если *n = 0,* то дизъюнкт Хорна называется *фактом,* если B – ложно ($\perp$), дизъюнкт называется *вопросом (запросом);* в противном случае дизъюнкт называется *правилом*.

Язык ProbLog расширяет логическое программирование, позволяя прикреплять вероятности к фактам. Такие факты называются *помеченными*. Пусть F обозначает множество помеченных фактов, тогда каждый элемент F имеет вид:

$$p::f$$

также сокращенно p_f, где p_f *= P(f) = ∈ [0,1]* является вероятностью и F соответствует синтаксису атома. Смысл атома в терминах теории вероятности — набор случайных величий. Например,

$$0{,}4::parent(X)$$

определяет набор случайных величин *parent(x)* для каждого экземпляра *parent(X),* полученные путем применения замены Θ к *parent(X)*. Например, *parent(X)θ = parent(*john*),* где *θ* = {john/X}, причем *θ* ∈ Θ. Результирующие атомы назваются *логическими фактами*.

В дополнение к помеченным фактам, программа ProbLog состоит из правил, которые являются базовыми для программы B. Теперь, пусть T = F ∪ B является программой ProbLog, и пусть Θ_f является набором всех возможных замен, связанных с логическим фактом *f.* Затем L_T определяется как *максимальное подмножество* множества всех логических фактов, которые могут быть добавлены к *B,* применением множества замен Θ_f к факту *f,* для каждого *f* ∈ *F.* Затем программа ProbLog определяет совместное вероятностное распределение для логических фактов L_T. Пусть $L \subseteq L_T$, тогда:

$$P_T(L)=\prod_{f\theta\in L} pf \prod_{\theta' f'\in L_{T/L}} (1-pf')$$

Теперь пусть *q* – любой запрос к программе ProbLog, тогда

$$P_T(q)=\sum_{L\subseteq L_T} P(q\vee L)\,P_T(L), \qquad (1)$$

где

$$P(q \vee L)=\begin{cases} 1, если \quad \theta : B \cup L\, СЛЕДУЕТ\, q\theta \\ 0, если\, ИНАЧЕ \end{cases}$$

Каждое L и P (q | L) = 1 называется *объяснением* q.

Семантика ProbLog называется семантикой распределения; это название было заимствовано из PRISM [Sato, 1995].В основном, в семантике распределения все факты, как предполагается, взаимно независимы.Тем не менее, это не означает, что ее невозможно закодировать вероятностными зависимостями: все зависимости определены на логическом уровне. Это позволяет определить любое совместное распределение вероятностей. Семантика распределения также имеет особые следствия для получения вероятностного взаимодействия между фактами, как показано в следующем примере.

Пример 1. Рассмотрим следующую (тривиальную) программу ProbLog **T**, которая представляет собой некоторые медицинские знания:

```
0.7 :: flu.
0.2 :: pneumonia.

fever ← flu.
fever ← pneumonia.
```

В этой программе *flu* и *pneumonia* - две независимые случайные переменные в соответствии с семантикой распределения. Теперь мы хотим, чтобы вычислить вероятность P_T (*fever*). Обращаем внимание, что *fever* может объясняться либо из *flu*, *pneumonia*, или из *flu* и *pneumonia* вместе взятых. Таким образом, в соответствии с уравнением (1), мы получаем:

$$P_T(fever)=P(flu)P(pneumonia)+P(flu)(1-P(pneumonia))+(1-P(flu))P(pneumonia)=¿$$
$$0{,}7*0{,}2+0{,}3*0{,}2+0{,}7*0{,}8=0{,}14+0{,}06+0{,}56=0{,}76$$

Как хорошо известно, этот вероятностный результат идентичен тому, что был бы получены с помощью одного из наиболее популярных способов моделирования взаимодействия (условного) независимых событий: так называемый *«зашумленный»-ИЛИ* [Pearl, 1988].

Так как *«зашумленное»-ИЛИ* основано на логической дизъюнкции, одном из 16 двоичных логических операторов, можно предположить, что могут быть и другие, схожие по смыслу, способы сочетания вероятностного доказательства. Тем не менее, для того, чтобы иметь возможность объединить доказательства, нужен алгебраический метод объединения информации. Вообще, алгебраической способ объединения вероятностной информация доступен из основной теории вероятностей, хотя он редко используется для моделирования логического взаимодействия. Ниже мы кратко рассмотрим необходимые основы из теории вероятности, со сверткой в качестве особого случая, а затем исследуем, как идеи из вероятностной логики, логического взаимодействия и общей логики могут быть объединены, чтобы получить более выразительную вероятностную логику, которую мы называем логикой вероятностного взаимодействия, или ProbIL, если кратко.

2.2 Вероятностное логическое взаимодействие

В дальнейшем, мы будем называть вероятность массового функция случайной величины X как f_x; P обозначает связанное распределение вероятностей. Классический результат из теории вероятностей, которая будет полезна при изучении сумм переменных следующая, находится по известной теореме (cf. [Grimmett и Stirzaker, 2001]).

Теорема 1. Пусть f является функцией общей совместной вероятности случайных величин X и Y , так, что X + Y = z. Затем полагаем, что

$$P(X+Y=z)=f_{x+y}(z)=\sum_x f(x, z-x).$$

Подтверждение. Смотри [Grimmett and Stirzaker, 2001].

Если X и Y являются независимыми, то отсюда вытекает следствие:

Следствие 1. Пусть X и Y две независимые случайные величины, то считая, что

$$P(X+Y=z)=\sum_x P(X=x)P(Y=z-x)=\sum_x f_x(x)f_y(z-x) \qquad (2)$$

Функция общей вероятности f_{X+Y} в этом случае называется *сверткой* f_X и f_Y, и обычно обозначаются как $f_{X+Y}=f_X*f_Y$. Теорема свертки очень полезна, так как суммы независимых случайных величин встречаются очень часто в теории вероятностей и статистике. Кроме того, ее можно применять рекурсивно, т.е.,

$$f_{X_1+\ldots+X_n}=f_{X_1}*\ldots*f_{X_n}$$

как следует из рекурсивного применения уравнения (2).

Теорема 1 содержит в себе не только сложение двух случайных переменных, но и булевых функций случайных переменных. Тем не менее, в отличие от полей вещественных чисел, где значение случайной переменной Y однозначно определяется вещественными числами x и z через $Y = z - x$, в булевой алгебре значения булевых переменных ограничено лишь значениями двугих булевых переменных. Эти ограничения могут привести к множеству значений, вместо одного значения, которое до сих пор совместимо с теоремой. В дальнейшем мы будем использовать обозначение $b(i, J) = q$ для таких ограничений, где логические значения i и q ограничивают J конкретными значениями. Например, для $(i \vee J) = q$, где вместо i,q подставляется $I=1$ (I имеет значение «ИСТИНА») и $Q = 1$ (Q имеет значение «ИСТИНА»), считая, что $J \in \{0, 1\}$. Таким образом, $f(i, (i \vee J) = q)$ является аббревиатурой для $f(i, j) + f(i, \neg j)$. Теорема может быть выражена следующим образом.

Теорема 2. Пусть f является функцией общей совместной вероятности случайных переменных I и J, считая, что $b(I, J) = q$, где b – булевая переменная. Затем, считаем, что $P(b(I, J) = q) = f_{b(I,J)}(q)=\sum_i f(i,b(i,J)=q)$.

Подтверждение. Пространство *(I, J)* определяется *b(I, J) = q,* и может быть разложено следующим образом: U_i *{ I = i }* ∩ *{ J = j* | *b(i, j) = q }*, где выражение *b (i, j) = q* следует интерпретировать как логическое ограничение на булевых значений переменной *J*. Поскольку отдельные множества *{ I = i }* ∩ *{ J = j* | *b*(*i, j*) = *q }* являются взаимоисключающими, получаем результат теоремы.

Следующее следствие для свертки получает в случае, если *I* и *J* независимы.

Следствие 2. Пусть *f* является функцией общей совместной вероятности независимых, случайных, булевых переменных *I* и *J,* и пусть *b* – булевая функция, определенная на *I* и *J,* тогда это означает

$$P(b(I,J)=q)=\sum_i f_I(i)P(b(i,J)=q)$$

Теорема 2 теперь является тем ключиком, необходимым для понимания ProbLog.

Пример 2. Пересмотрим **пример 1** и логическое соотношение *L* ∨ *P = F*, где *L* означает *«flu»* , *P* для *«pneumonia»* , и *F* для *«fever»*. Используем те же распределения вероятностей, что в примере 1: *P (l)* = 0,7 и *P (p)* = 0,2. Применяя теорему 2, получаем следующие результаты:

$$P(L \quad P=f)=\sum_l f_l(l)P(l \quad P=f)=\text{¿}$$
$$f_L(l)(f_P(p)+f_P(\quad p))+f_L(\quad l)f_P(p)=\text{¿}$$
$$f_L(l)f_P(p)+f_L(l)f_P(\quad p)+f_L(\quad l)f_P(p)=\text{¿}$$
$$0.7\cdot 0.2+0.7\cdot 0.8+0.3\cdot 0.2=0.76$$

где терм $f_P(p)+f_P(\neg p)$ *вытекает из логического ограничения l∨ P = f, т.е., P* ∈ *{0, 1}*.

Таким образом, пример показывает, что *«зашумленное»-ИЛИ* может быть описана через свертку. Для детальной информации смотри [Lucas and Hommersom, 2010].

Соответствие между двумя подходами заключается в следующем. Булева функция b соответствует детерминированному распределению вероятностей $P\ (q \mid L)$ в уравнении (1). В терминах логического вывода $P\ (q \mid L)$ определяется как $q\theta$ от $B \cup L$; в этой статье рассматривается вопрос, существуют ли способы, чтобы заменить логический вывод методом рассуждения, который включает в себя вероятностную логическое взаимодействие как способ выражения взаимодействия между правилами. С помощью логического взаимодействия можно рассматривать правила интерференции - естественный способ расширить ProbLog заменой стандартной логики исходной логикой.

2.3 Исходная логика (Логика по умолчанию)

В логике умолчанию [Reiter, 1980] добавляется еще одно специальное правило вывода (интерференции), называемое *умолчанием*, к обычной логике первого порядка предикатов. *Умолчания* имеют следующий вид:

$$\frac{\textit{предпосылка} : \textit{обоснование}}{\textit{следствие}}$$

где *«предпосылка»* является условием, если оно истинно, то может быть получено следствие *«следствие»*, однако, только тогда, когда результирующая теория совместно с предположениями, описываемых *«обоснованием»* является устойчивым. Результирующая теория обозначается $T = (D,\ W)$, где W, в этой статье, обозначает набор *фактов* и *правил* в логике Хорна , как и в ProbLog, и D является *умолчанием*. Вывод в логике по умолчанию происходит с помощью вычисления расширения выходной теории, используя оператор фиксированной точки. Мы будем писать $(D,W) \mid\!\sim \phi$, если $\phi \in E$, где E – расширение *умолчания* (D,W).

В следующем разделе будет показана логика вероятностного взаимодействия, ProbIL, используя умолчания для представления логического взаимодействия, которые в сочетании с вероятностями дает вероятностное логическое взаимодействие как в разделе 2.2.

3 Логика вероятностного взаимодействия

3.1 Общая идея

Пример 1 иллюстрирует, как вероятностные языки (например, ProbLog) неявно объединяют вероятностные доказательства с помощью логической дизъюнкции, что соответствует оператору «зашумленного»-ИЛИ. Этот выбор приводит к определенному вероятностному поведению, которое не всегда может быть обосновано. Результирующая вероятность, при использовании «зашумленного»-ИЛИ, всегда больше, чем сумма ее компонентов, т. е., вероятностное поведение монотонно возрастает: $pq + p(1 - q) + (1 - p)q \geq p, q$. Таким образом, невозможно смоделировать такие конкретные события, чтобы, когда они были взяты все вместе, и в то же время компенсировали («гасили») друг друга. В отличие от монотонных, немонотонные логики, например, логика по умолчанию, могут быть использованы для реализации такого рассуждения.

Общая идея логики вероятностного взаимодействия (ProbIL)- это заменить исходную теорию с помощью теории на языке логики по умолчанию. Эта логика имеет те же преимущества, что и логика по умолчанию, а именно: мы можем определить проблемы в исходном поведении без ущерба для обычного логического вывода (дедукции).

Формально, вероятность P (q | L) адаптировано следующим образом. Мы предполагаем, у нас есть фоновая исходная теория, состоящая из множества умолчаний D и набор стандартных логических правил B. Тогда мы имеем:

$$P(q \vee L) = \begin{cases} 1, если & \theta:(D, B\,U\,L)\,СЛЕДУЕТ\,q\theta \\ & 0, если\,ИНАЧЕ \end{cases}$$

т. е. монотонные логический вывод заменяется логическим выводом по умолчанию |~ .

Пример 3. Популярный пример из немонотонных логик «birds typically fly» моделируются значением по умолчанию $D = \{\frac{Bird(x): Flies(x)}{Flies(x)}\}$, и мы добавляем стандартное логическое выражение «penguins do not fly», т.е., $B = \{Penguin(x) \rightarrow \neg Flies(x)\}$. Кроме того известно, что в зоомагазинах 30% всех животных — птицы, т.е., `0,3::Bird`. Чтобы вычислить, летает ли *Tweety (T),* мы должны вычисляем *P* (Flies(T)) = *P* (Flies(T) | Bird(T)) *P* (Bird(T)) = 0.3.

Можно оформить как правило ProbLog:

Bird(*x*) ∧ Penguin(*x*) → Flies(*x*)

Однако, если, как в примере, мы не знаем, летает ли Tweety, и мы не имеем вероятность того, что Tweety пингвин, то *P* (Flies (T)) = 0. Это иллюстрирует взаимодействие между немонотонными и вероятностными рассуждениями, которые не доступны во многих существующих вероятностных логиках.

В дальнейшем мы используем этот механизм для изучения типов взаимодействий, которые могут быть смоделированы и способов представления определенных явно, результирующих общих вероятностей с помощью вероятностного логического взаимодействия и оператора свертки, в случае независимости.

3.2 Булевое взаимодействие

Рассмотрим следующие предложения языка ProbLog: {a ← c, a ← e}. Чтобы смоделировать взаимодействие между вероятностными доказательствами, способствуя вероятности a, мы должны определить сочетание функции для этих положений, то есть, мы хотим интерпретировать эти положения как $a \leftarrow b$ (c, e), с булевой функцией b. Как мы видим, в случае ProbLog, $b(c,e)$ является всегда эквивалентно $c \vee e$. Следующее предложение выражает, что все возможные интерпретации булевых взаимодействий действительно могут быть смоделированы в логике по умолчанию.

Предложение 1. Для всех булевых функций b и множества атомов $H, B_1, \ldots, B_n$ существует логика умолчанию теории D, которая не содержит b, таким образом, что:

$$\exists\theta : (D, B \cup L) \mid\sim q\theta$$

iff

$$\exists\theta : B \cup L \cup H \leftarrow b(B_1, \ldots, B_n) \quad q\theta$$

Чтобы проиллюстрировать это, мы перечислим ряд общих двоичных булевых функций на рисунке 1, с соответствующим их функциями логики по умолчанию.

Name	*Default rules D*	*Choices for L*
false	$\varnothing$	$\varnothing$
true	$\left\{\frac{\top}{H}\right\}$	$\wp(\{B_1, B_2\})$
AND	$\left\{\frac{B_1, B_2}{H}\right\}$	$\{\{B_1, B_2\}\}$
OR	$\left\{\frac{B_1}{H}, \frac{B_2}{H}\right\}$	$\wp(\{B_1, B_2\}) \setminus \{\varnothing\}$
EQ	$\left\{\frac{B_1, B_2}{H}, \frac{: \neg B_1, \neg B_2}{H}\right\}$	$\{\varnothing, \{B_1, B_2\}\}$
XOR	$\left\{\frac{B_1 : \neg B_2}{H}, \frac{B_2 : \neg B_1}{H}\right\}$	$\{\{B_1\}, \{B_2\}\}$

Рисунок 1: Представление двоичных коммутативных и ассоциативных логических функций, используя логику по умолчанию для правила

$R: H \leftarrow b(B_1, B_2)$, так что *(D,L) |~ H iff {R}* ∪ *L |= b(B1,B2).* Здесь $\wp$ (L) обозначает множетсво реальных значений L.

Следовательно, для данного набора взаимодействий, мы можем заменить соответствующие правила соответствующей теории, где взаимодействия моделируются правилами по умолчанию.

3.3 Вероятностное булевое взаимодействие

Логика умолчанию обеспечивает механизм рассуждения для логических взаимодействий. В этом разделе мы рассмотрим вероятностную перспективу, по обобщая результаты, представлене в раздел 2.2. Как уже упоминалось, теорема 2 может быть использована в качестве основы логики вероятностного взаимодействия.

Теорема 3. Пусть Q – атом, который представляется в теории T одиночным правилом:

$$Q \leftarrow b(I, J).$$

Для функции общей вероятности f_Q из Q имеем:

$$f_Q(q) = \sum_i f(i, b(i, J) = q)$$

Подтверждение.

$$f_Q(q) = \sum_{L \subseteq L_T} P(q \vee L) P_T(L)$$
$$\sum_{L \subseteq L_T} P(b(I, J) = q \vee L) P_T(L)$$
$$\sum_{L \subseteq L_T} \sum_i P(i, b(i, J) = q \vee L) P_T(L)$$
$$\sum_i f(i, b(i, J) = q)$$

4 Выводы

В этой статье мы предложили новый механизм для моделирования взаимодействия в вероятностных логик. Как указывается в [Poole, 1990], абдукция может рассматриваться как формализм для объяснения наблюдений, в то время как логика по умолчанию используется, чтобы сделать прогноз.

В то время как языки, такие как ProbLog, используют немонотонный подход (абдукция) для объяснения возможного запроса, прогнозы полностью монотонны. Мы использовали рассуждения по умолчанию в этом контексте прогнозирования в качестве способа для моделирования взаимодействия между вероятностными фактами.

Благодарности

Arjen Hommersom был поддержан VENI Grant 639.021.918 от Нидерландской организации Научных Исследований. Мы благодарим рецензентов за их конструктивное комментарии, которые значительно улучшили эту статью.

Ссылки

1. [Baral *et al.*, 2009] Chitta Baral, Michael Gelfond, and Nelson Rushton. Probabilistic reasoning with answer sets. *Theory and Practice of Logic Programming*, 9:57–144, 2009.

2. [De Raedt *et al.*, 2008] L. De Raedt *et al.* Towards digesting the alphabet-soup of statistical relational learning. In *NIPS 2008 Workshop on Probabilistic Programming*, 2008.

3. [Eiter *et al.*, 1997] T. Eiter, G. Gottlob, and Leone N. Semantics and complexity of abduction from default theories. *Artificial Intelligence*, 90(1-2):177–223, 1997.

4. [Grimmett and Stirzaker, 2001] G. Grimmett and D. Stirzaker. *Probability and Random Processes*. Oxford University Press, Oxford, 2001.

5. [Gutmann *et al.*, 2010] B. Gutmann, M. Jaeger, and L. De Raedt. Extending problog with continuous distributions. In *Proc ILP2010*, 2010.

6. [Kakas, 1994] A. Kakas. Default reasoning via negation as failure. In G. Lakemeyer and B. Nebel, editors, *Foundations of Knowledge Representation and Reasoning*, volume 810 of *LNCS*, pages 160–178. Springer Berlin / Heidelberg, 1994.

7. [Kimmig *et al.*, 2010] A. Kimmig, B. Demoen, L. De Raedt, V. Santos Costa, and R. Rocha. On the implementation of the probabilistic logic programming language ProbLog. *Theory and Practice of Logic Programming*, 2010.

8. [Lucas and Hommersom, 2010] P.J.F. Lucas and A.J. Hommersom. Modelling the interactions between discrete and continuous causal factors in Bayesian networks. In *Proc PGM-2010*, pages 185–192, 2010.

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

10. [Poole, 1990] D. Poole. A methodology for using a default and abductive reasoning system. *International Journal of Intelligent Systems*, 5(5):521–548, 1990.

11. [Poole, 1993] D. Poole. Probabilistic Horn abduction and Bayesian networks. *Artificial Intelligence*, 64:81–129, 1993.

12. [Poole, 1997] D. Poole. The independent choice logic for modelling multiple agents under uncertainty. *Artificial Intelligence*, 94(1-2):7–56, 1997.

13. [Reiter, 1980] R. Reiter. A logic for default reasoning. *Artificial Intelligence*, 13(1-2):81–132, 1980.

14. [Richardson and Domingos, 2006] M. Richardson and P. Domingos. Markov logic networks. *Machine Learning*, 62(1-2):107–136, 2006.

15. [Sato, 1995] T. Sato. A statistical learning method for logic programs with distribution semantics. In L. Sterling, editor, *Int Conf Logic Programming*, pages 715–729. MIT Press, 1995.

16. [Tompits, 2003] H. Tompits. Expressing default abduction problems as quantified Boolean formulas. *AI Commun.*, 16:89–105, June 2003.

17. [Zhang and Poole, 1996] N.L. Zhang and D. Poole. Exploiting causal independence in Bayesian network inference. *JAIR*, 5:301–328, 1996.