

# Revisão de Teorias Relacionais Probabilísticas através de Exemplos com Invenção de Predicados

Kate Revoredo, Gerson Zaverucha

<sup>1</sup>Departamento de Engenharia de Sistemas e Computação - COPPE  
Universidade Federal do Rio de Janeiro (UFRJ)

katerevored@uniriotec.br, gerson@cos.ufrj.br

**Abstract.** *The majority of machine learning algorithms use only the vocabulary explicitly supplied in the data for model construction. However, automatic extension of this vocabulary, with new structures that represent implicit information on objects of the domain, can enrich the learning. On the other hand, an initial model approximately correct can be provided to the learning algorithms. Therefore, a particular case of learning, called theory revision, which minimally modifies the structure of the model to correctly reflect the database, can be applied. This article investigates the benefits of vocabulary extension when applying theory revision techniques to probabilistic models. This proposal was applied successfully to artificial and realistic databases.*

**Resumo.** *A maioria dos algoritmos de aprendizado de máquina utiliza apenas o vocabulário fornecido explicitamente nos dados para construir os modelos. Entretanto, a extensão automática desse vocabulário com novas estruturas que representam informações implícitas sobre objetos do domínio pode enriquecer o aprendizado. Por outro lado, em alguns casos, já existe um modelo que é aproximadamente correto podendo a sua estrutura ser alterada o mínimo possível de forma a refletir corretamente a base de dados, caracterizando um caso particular de aprendizado denominado revisão de teoria. Este artigo, investiga os benefícios da extensão do vocabulário quando revisando modelos probabilísticos, tendo sido aplicada com sucesso em bases de dados artificiais e reais.*

## 1. Introdução

Aprendizado de máquina indutivo [Mitchell 1997] tem por objetivo o desenvolvimento de ferramentas e técnicas para induzir modelos a partir de observações (exemplos). Diferentes mecanismos de representação dos modelos, como (i) rede Bayesiana (RB), (ii) teoria de primeira-ordem e (iii) teoria probabilística de primeira-ordem (TPPO) (veja Seção 3), originam diferentes algoritmos de aprendizado. A maioria deles usa apenas o vocabulário fornecido nos dados para construir os modelos. Por exemplo, se o *dataset* apenas contém informações sobre o gosto dos usuários por filmes, então a única relação que será representada no modelo final, quando considerando lógica de primeira-ordem, é a relação

*gosta(Usuario, Filme)*. No entanto, informações sobre os indivíduos podem enriquecer o processo de aprendizado, muitas vezes tornando o modelo aprendido mais eficiente e/ou compacto. Por exemplo, definir se um usuário  $x$  gosta de um filme  $y$ , indica a importância de conhecimento individual, tais quais a idade de  $x$  ou o gênero de  $y$ . Na prática, entretanto, não é sempre que características como essas estão disponíveis no *dataset*. É interessante então, a utilização de técnicas para estender de forma automática o vocabulário inicial com novas estruturas que representem essas informações "escondidas", sendo estas consideradas durante o aprendizado dos modelos. Em RB essas estruturas são conhecidas como *variáveis não-observadas* [Friedman 1997], em lógica de primeira-ordem por *predicados inventados* [Kramer 1995] e em lógica de primeira-ordem probabilística vamos denominar *predicados probabilísticos inventados* [Revoredo 2009].

Geralmente, os sistemas de aprendizado aprendem modelos partindo apenas do *dataset* e de um conhecimento preliminar invariante. Entretanto, em algumas situações, já existe um modelo inicial que é aproximadamente correto, ou seja apenas alguns pontos da sua estrutura o impede de refletir corretamente o *dataset*. Neste caso, é mais vantajoso considerar este modelo como ponto de partida para o aprendizado, encontrando pontos potenciais a sofrerem modificação, denominados *pontos de revisão*. Temos, então, um caso particular do aprendizado, que é a revisão do modelo a partir dos exemplos. A técnica de revisão utiliza exemplos classificados incorretamente para guiar a busca por pontos de revisão. *Operadores de revisão* são então utilizados para propor alterações a esses pontos, como inclusão ou exclusão de dependência entre as variáveis do modelo. A melhor proposta é então escolhida, de acordo com uma função de avaliação, para ser implementada. A revisão prossegue enquanto for possível melhorar o modelo corrente. Como técnicas de revisão definem pontos de revisão, o espaço de busca de modelos é menor, tornando o aprendizado mais rápido e o modelo final tão eficiente quanto o modelo aprendido a partir do zero. Em [Paes et al. 2005b, Paes et al. 2005a, Paes et al. 2006] nós propomos o primeiro sistema de revisão de TPPOs, denominado PFORTE, incluindo um estudo da melhor função de avaliação a ser utilizada. Em [De Raedt et al. 2008, Revoredo 2009] os benefícios da utilização de técnicas de revisão também foi verificado quando revisando programas Prolog Probabilísticos (ProbLog) [De Raedt et al. 2007].

Motivados, então, pela possibilidade de (i) aprender modelos melhores a partir da extensão da linguagem com novas estruturas e (ii) reduzir o espaço de busca a partir da utilização de técnicas de revisão de teoria, nós propomos neste artigo a revisão de modelos probabilísticos utilizando invenção e introdução de novas estruturas à linguagem, tornando o modelo final mais eficaz e/ou compacto. Nós validamos a proposta em duas formas de representação de conhecimento probabilístico, RB e TPPOs, propondo dois algoritmos de revisão: o primeiro, denominado **DAHVI** (*Discriminative Approach for Hidden Variable Introduction*), revisa RBs introduzindo variáveis não-observadas [Revoredo et al. 2009, Revoredo 2009] e o segundo, denominado **PFORTE-PI** (*Probabilistic First Order Revision of Theories from Examples with Predicate Invention*) [Revoredo et al. 2006,

Revoredo et al. 2007, Revoredo 2009], estende o nosso sistema de revisão PFORTE com operadores de revisão que introduzem predicados probabilísticos inventados .

Este artigo está organizado da seguinte forma: na Seção 1 introduzimos o nosso trabalho. Nas Seções 2 e 3 revisamos conceitos de RBs e TPPOs para então na Seção 4 apresentar a nossa proposta. Os resultados experimentais são exibidos na Seção 5 e na Seção 6 apresentamos algumas conclusões.

## 2. Redes Bayesianas

Uma RB é um grafo direcionado acíclico, onde os nós correspondem a variáveis aleatórias,  $Vars = \{X_1, \dots, X_n\}$ , e estas têm uma distribuição de probabilidade condicional (CPD) associada. As CPDs quantificam para cada combinação de valores de uma variável  $X_i$  e dos seus pais  $Pa(X_i)$  o efeito desses em  $X_i$  ( $P(X_i|Pa(X_i))$ ). Dada uma RB é possível efetuar consultas: qual a probabilidade de um conjunto de variáveis  $Y$  (*variáveis de consulta*), assumir determinados valores  $y$  e condicionadas a um conjunto de valores de outras variáveis  $Ev$  (*variáveis de evidência*) ( $P(Y_1 = y_1, \dots, Y_k = y_k | Ev_1 = ev_1, \dots, Ev_j = ev_j)$ ). Para responder a essas consultas utiliza-se um procedimento de inferência. Quando a tarefa é de classificação, as variáveis de consulta são as variáveis de classe, enquanto que para tarefas mais gerais o usuário escolhe o conjunto de variáveis para as quais ele quer inferir um valor.

Considere o conjunto de exemplos treinamento  $E = \{e^1, \dots, e^m\}$  onde cada exemplo  $e^i$  associa um valor a alguma (ou toda) variável em  $Vars$ . A tarefa de aprender uma RB é uma busca pela melhor RB que descreve  $E$ . Uma abordagem usual é avaliar cada RB existente no espaço de busca de acordo com uma função de avaliação escolhendo a de melhor avaliação. Geralmente, um algoritmo subida de encosta guloso com recomeços aleatórios é utilizado, com um arco sendo inserido/excluído/alterado por vez. Quando o conjunto de exemplos de treinamento é parcialmente observado, isto é, algum  $e^i$  não associa valor para alguma (ou toda) variável em  $Vars$ , a tarefa de aprendizado é computacionalmente e conceitualmente muito mais difícil. Uma questão, ainda mais complicada e desafiadora é aprender uma RB identificando novas variáveis não-observadas. Essa tarefa envolve escolher o número de variáveis não-observadas, suas cardinalidades e a dependência entre elas e as outras entidades do domínio, ou seja a determinação de onde a variável será inserida. Estas decisões são cruciais para encontrar uma boa generalização. Em particular, o algoritmo *SEM* [Friedman 1998] é capaz de aprender considerando a inserção de variáveis não-observadas, entretanto é preciso informar a priori quantas são inseridas. Além disso, ele não atinge bons resultados, a não ser que algum conhecimento a priori da localização e da cardinalidade delas seja conhecida (pelo menos aproximada).

## 3. Teorias Probabilísticas de Primeira-ordem

Uma TPPO é um conjunto de cláusulas de Horn com uma CPD associada, onde esta quantifica a influencia dos antecedentes da cláusula no conseqüente, denominada *cláusula probabilística*. Os predicados de uma cláusula probabilística são denominados *predicados*

*probabilísticos*, já que eles tem um domínio e uma CPD associados. No domínio da *Familia*, por exemplo, temos o predicado probabilístico  $genero(X)$  com domínio  $D_{genero} = \{masculino, feminino\}$  e CPD igual a  $\langle P(genero(X) = masculino) = 0.6, P(genero(X) = feminino) = 0.4 \rangle$ . Da mesma forma que a partir de uma RB podemos efetuar consultas, a partir de uma TPPO também. Nesse caso, as variáveis de consulta e as variáveis de evidência são predicados probabilísticos básicos, ou seja predicados instanciados (ex:  $P(irmao(joao, jose) = verdadeiro | genero(joao) = masculino)$ ). Para responder a essas consultas primeiro é construída uma prova lógica para as variáveis de consulta. Em seguida essa prova é representada como uma RB, onde os nós dessa rede correspondem aos predicados probabilísticos utilizados na prova. Dessa forma, o domínio e a CPD dos nós são os mesmos dos predicados probabilísticos. Algum algoritmo de inferência para RB pode então ser aplicado para inferir o valor da consulta.

Aprender uma TPPO significa aprender as cláusulas probabilísticas e as CPDs correspondentes. Para aprender as cláusulas, técnicas de *Inductive Logic Programming* (ILP) [Muggleton 1992] podem ser aplicadas considerando funções de avaliação probabilísticas para seleção da melhor TPPO. As CPDs são aprendidas utilizando variações do algoritmo EM e do gradiente [Kersting and De Raedt 2002] e utilizam as RBs construídas para cada um dos exemplos como descrito no parágrafo anterior.

#### 4. Revisando modelos probabilísticos com invenção de estruturas

Nesta seção descrevemos a nossa proposta para revisar modelos probabilísticos considerando invenção e introdução de novas estruturas. O algoritmo em alto nível pode ser visto em Algorithm 1, onde os operadores de revisão são os responsáveis pela extensão da linguagem, definição do domínio da nova estrutura e localização da inserção. Diferentemente do algoritmo SEM, que necessita de informações a priori sobre a nova estrutura a ser inserida, o nosso algoritmo determina em tempo de execução a quantidade de novas estruturas, sua localização e cardinalidade. Esse algoritmo é especializado para RBs e TPPOs, onde a diferença entre eles consiste na estrutura dos pontos de revisão e no operador de revisão proposto.

---

**Algorithm 1** Revisando modelos probabilísticos através da introdução de novas estruturas

---

```

1: repete
2:   gera pontos de revisão;
3:   para todo ponto de revisão faça
4:     gera possíveis revisões
5:     aprende as CPDs
6:     avalia possíveis revisões
7:     atualiza melhor revisão encontrada
8:   se a melhor revisão melhora o modelo corrente então
9:     implementa melhor revisão
10: até não ser mais possível melhorar o modelo

```

---

O DAHVI determina os pontos de revisão como sendo variáveis de consulta e suas *Coberturas de Markov*<sup>1</sup> correspondentes, já que as variáveis que influenciam o valor inferido para as variáveis de consulta pertencem a Cobertura de Markov. A seleção dos pontos de revisão depende dos exemplos, logo se alguma variável considerada como ponto de revisão não for observada em um exemplo então a sua Cobertura de Markov também faz parte do conjunto de pontos de revisão determinado por esse exemplo e assim sucessivamente. O operador de revisão proposto define uma estrutura candidata com uma variável não-observada intermediando um ponto de revisão e seus pais. Cada variável não-observada é introduzida considerando cardinalidade 2, ou seja um domínio binário, já que assim aumentamos o número de parâmetros probabilísticos o mínimo possível. Apesar de apenas acrescentar uma variável não-observada intermediária entre 2 variáveis, a avaliação das redes candidatas pode ser alterada para melhor, já que as CPDs são outras. Além disso, dependendo do tamanho do domínio das variáveis envolvidas, ao acrescentar a variável não-observada reduz-se o número de parâmetros probabilísticos, podendo tornar a inferência e o aprendizado dos parâmetros mais rápido e preciso.

Como vimos na seção 3, para cada exemplo uma RB é construída e as variáveis dessa rede tem uma associação direta com as cláusulas e seus predicados na TPPO. Dessa forma, os pontos de revisão são definidos pelo PFORTE.PI como sendo as cláusulas cuja a cabeça está associada a uma variável da rede selecionada como ponto de revisão pelo algoritmo de seleção de pontos de revisão do DAHVI. Dois novos operadores de revisão são propostos pelo PFORTE.PI: *operador de compactação* e *operador de incremento*. A motivação para o uso do primeiro é a simplificação da teoria com conseqüente melhora na acurácia, e a do segundo a detecção de possíveis conexões entre variáveis aleatórias além de apenas simplificar a teoria. O operador de compactação inventa um predicado com a mesma aridade e domínio da cabeça da cláusula ponto de revisão. Em seguida, ele substituí o corpo dessa cláusula pelo predicado probabilístico inventado e cria uma cláusula para definir este novo predicado com o corpo substituído. Para ilustrar, considere a cláusula  $irmao(X, Y) : \neg genero(X), irmaos(X, Y)$ . A proposta de revisão do operador de compactação é composta das cláusulas probabilísticas:  $irmao(X, Y) : \neg novo(X, Y)$  e  $novo(X, Y) : \neg genero(X), irmaos(X, Y)$ , onde  $novo(X, Y)$  é o predicado probabilístico inventado. Já o operador de incremento substituí um dos antecedentes da cláusula ponto de revisão pelo predicado inventado probabilístico, cuja aridade e o domínio são os mesmos desse antecedente que ele substituiu. Em seguida, uma cláusula para definir o novo predicado é criada com o antecedente substituído no corpo e especializada na tentativa de encontrar uma TPPO com melhor avaliação. Ao final, para cada antecedente ( $A_i$ ) na cláusula original ( $C$ ) nós definimos um par de cláusulas ( $C_{mod}$  e  $C_{novo}$ ). A primeira é  $C$  com  $A_i$  substituído pelo predicado inventado probabilístico ( $novo_i$ ), e a segunda a cláusula definindo  $novo_i$  com  $A_i$  e talvez outros antecedentes no corpo. As CPDs são aprendidas para todos os pares. O par que fornecer a melhor avaliação será considerado como a modificação proposta pelo operador de incremento. Como exemplo

---

<sup>1</sup>Cobertura de Markov de um nó em uma RB consiste dos seus pais, filhos e dos pai dos filhos.

considere a cláusula  $irmao(X, Y) : \neg genero(X), irmaos(X, Y)$ . O operador de incremento escolhe uma proposta dentre duas possíveis, uma para o antecedente  $genero(X)$  e outra para o  $irmaos(X, Y)$ . Para este podemos ter como proposta, por exemplo, o par:  $irmao(X, Y) : \neg genero(X), novo(X, Y)$  e  $novo(X, Y) : \neg irmaos(X, Y), genero(Y)$ . Em ambos os operadores, outras cláusulas da TPPO que contenham como subconjunto do seu corpo o corpo da cláusula que define o predicado inventado também são alteradas para substituir esse grupo de antecedentes pelo novo predicado  $novo(X, Y)$ . Dessa forma, todas as cláusulas que foram alteradas têm as suas CPDs simplificadas e conseqüentemente a inferência na rede Bayesiana resultante também.

## 5. Resultados Experimentais

Para validar a nossa proposta, consideramos *datasets* discretos e cuja tarefa é classificação: 10 para o DAHVI (*Audiology, Breast Cancer, Breast Cancer Wisconsin, Car evaluation, Lymphography, Nursery, Post-operative Patient, Primary Tumor, Tic-tac-toe Endgame* e *Zoo* (obtidas aleatoriamente do repositório do UCI)) e 2 para o PFORTE.PI (*Alzheimer* e *Mutagenesis*) (veja detalhes sobre as bases em [Revoredo 2009]). Utilizamos 10-fold validação cruzada, separando os dados em conjuntos disjuntos de treinamento e teste, com exceção dos experimentos com a base Amine, onde foi considerado 5-fold validação cruzada, por questão de tempo (a execução de 1 fold demora aproximadamente 8 dias). Além disso, para o DAHVI, em cada conjunto de treinamento, para evitar overfitting, foi utilizado 5-fold validação cruzada, separando o conjunto de treinamento em conjuntos disjuntos de treinamento e validação [Kohavi 1995], onde a melhor RB, considerando o conjunto de validação, é avaliada no conjunto de teste. Utilizamos como função de avaliação a log-verossimilhança condicional (CLL). Para a avaliação do modelo final consideramos a média da acurácia (ACC) das variáveis de consulta. Utilizamos t-test corrigido [Nadeau and Bengio 2003] com 95% de confiança para verificar a significância das diferenças entre as médias.

O objetivo dos experimentos é verificar se a nossa proposta é capaz de encontrar modelos melhores ao revisar um modelo inicial considerando a invenção e introdução de novas estruturas. No caso do DAHVI os modelos iniciais são aprendidos utilizando os algoritmos SEM, *hill-climbing* (HC) e *Naive Bayes* (NB) e no caso do PFORTE.PI, o Aleph [Srinivasan 2001].

O DAHVI melhorou significativamente a ACC da rede SEM em metade dos *datasets*, não piorando em nenhum. Já revisando as redes HC, DAHVI obtém melhoras, mas não tantas quanto as obtidas com as redes SEM. A pior performance foi para as redes NB, como esperado, já que para essa estrutura a Cobertura de Markov da variável de consulta inclui todas as variáveis. Verificamos também que o DAHVI reduziu em mais de 50% o espaço de busca considerado para inclusão de variáveis não-observadas em todos os *datasets* quando revisando as redes HC e SEM, com exceção do *dataset Nursery* onde a redução foi de 33.34%, quando revisando a rede HC. Em [Revoredo et al. 2009, Revoredo 2009] outros experimentos são apresentados, como por

exemplo, (i) a avaliação da proposta para seleção de pontos de revisão, (ii) a análise da quantidade de variáveis inseridas e (iii) a aplicação em 2 *datasets* cuja a tarefa não é de classificação mostrando que a nossa abordagem é aplicável a *datasets* gerais. O PFORTE.PI revisou a teoria retornada pelo Aleph, e apesar de ter aumentado o número de cláusulas, e no *datasets* Mutagenesis também o número de literais e parâmetros probabilísticos, encontrou uma TPPO mais acurada, sendo que para base Amine a diferença em ACC não foi estatisticamente significativa. Em todos os folds do Mutagenesis o operador de incremento foi o escolhido, enquanto que nos folds do Amine foi o de compactação. Como o PFORTE.PI é uma extensão do PFORTE, nós também verificamos se os novos operadores proporcionam uma melhora significativa à TPPO de forma que eles sejam preteridos aos demais operadores de revisão do PFORTE (por exemplo adição e remoção de antecedentes). Para isto, rodamos o PFORTE.PI considerando todos os operadores de revisão e comparamos com o PFORTE. Para os dois *datasets* a TPPO retornada pelo PFORTE.PI melhorou a ACC. No caso do *datasets* Mutagenesis o operador de incremento foi escolhido em 8 dos 10 folds e no *datasets* Amine 4 dos 5 folds. Nos demais folds nenhum dos dois operadores de revisão propostos foi escolhido. O tempo médio de execução do PFORTE.PI é maior do que do PFORTE, como era de se esperar, já que no geral os pontos de revisão são os mesmos e para cada um deles o PFORTE.PI propõe duas novas modificações.

## 6. Conclusão

Nesta tese, nós propomos dois sistemas de revisão de modelos probabilísticos: DAHVI [Revoredo et al. 2009, Revoredo 2009] para revisar RBs e PFORTE.PI [Revoredo et al. 2006, Revoredo et al. 2007, Revoredo 2009] para revisar TPPOs. Em ambos os sistemas nós apresentamos (i) formas de selecionar pontos de revisão e (ii) operadores de revisão. A principal contribuição desta tese foi a definição de operadores de revisão que estendem a linguagem corrente, introduzindo novas estruturas ao modelo: (i) variáveis não-observadas no caso de RBs e (ii) predicados inventados probabilísticos no caso de TPPOs. A proposta foi avaliada com sucesso em *datasets* reais.

Uma possível extensão do DAHVI é a consideração de outros domínios para as variáveis não-observadas diferentes do binário e a consideração de outros operadores de revisão, como um de exclusão de arestas. O PFORTE.PI também pode se beneficiar de novos operadores, como por exemplo, um operador que especializa a cláusula ( $C$ ) que define o predicado inventado depois de buscar na TPPO outras cláusulas que tenham no seu corpo o antecedente do corpo de  $C$ . Dessa forma, a especialização que a cláusula que define o predicado inventado sofrer será refletida para outras cláusulas na TPPO. Esse novo operador permite tanto a redução do número de parâmetros quanto a possibilidade de especialização de várias cláusulas ao mesmo tempo.

## References

De Raedt, L., Kersting, K., Kimmig, A., Revoredo, K., and Toivonen, H. (2008). Compressing probabilistic prolog programs. *Machine Learning*, 70(2-3):151–168.

- De Raedt, L., Kimmig, A., and Toivonen, H. (2007). Problog: A probabilistic prolog and its application in link discovery. In *Proc. IJCAI-2007*, pages 2462–2467.
- Friedman, N. (1997). Learning belief networks in the presence of missing values and hidden variables. In *14th Int. Conference on Machine Learning*, pages 1252–133.
- Friedman, N. (1998). The bayesian structural EM algorithm. In *UAI*, pages 129–138.
- Kersting, K. and De Raedt, L. (2002). Basic principles of learning bayesian logic programs. Technical Report 174, Univ. of Freiburg, Inst. for Computer Science, German.
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proc. of the IJCAI*, pages 1137–1145.
- Kramer, S. (1995). Predicate invention: A comprehensive view. Technical Report ÖFAI-TR-95-32, Austrian Research Institute for Artificial Intelligence.
- Mitchell, T. (1997). *Machine Learning*. McGraw-Hill, New York.
- Muggleton, S. (1992). *Inductive logic programming*. McGraw-Hill, New York.
- Nadeau, C. and Bengio, Y. (2003). Inference for the generalization error. *Machine Learning*, 52(3):239–281.
- Paes, A., Revoredo, K., Zaverucha, G., and Costa, V. S. (2005a). Further experimental results of probabilistic first-order revision of theories from examples. In *4th Workshop on Multi-Relational Data Mining / The Eleventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, page 59, Chicago, Illinois. ACM Press.
- Paes, A., Revoredo, K., Zaverucha, G., and Costa, V. S. (2005b). Probabilistic first-order theory revision from examples. In *Proceedings of the 15th International Conference on Inductive Logic Programming*, volume 3625 of *LNAI*, pages 295–311. Springer.
- Paes, A., Revoredo, K., Zaverucha, G., and Costa, V. S. (2006). Pforte: Revising probabilistic fol theories. In *Proceedings of the 18th Brazilian AI Symposium (SBIA-06)*, volume 4140 of *LNAI*, pages 441–450. Springer.
- Revoredo, K. (2009). *Revisão de Teorias Relacionais Probabilísticas através de Exemplos com Invenção de Predicados*. PhD thesis, COPPE-Sistemas/UFRJ. [www.uniriotec.br/katerevoredotese](http://www.uniriotec.br/katerevoredotese).
- Revoredo, K., Paes, A., Zaverucha, G., and Costa, V. S. (2006). Combining predicate invention and revision of probabilistic fol theories. In *Short paper proceedings of 16th International Conference on Inductive Logic Programming (ILP-06)*, pages 176–178.
- Revoredo, K., Paes, A., Zaverucha, G., and Costa, V. S. (2007). Combinando invenção de predicados e revisão de teorias de primeira-ordem probabilísticas. In *VI ENIA*.
- Revoredo, K., Paes, A., Zaverucha, G., and Costa, V. S. (2009). Revisando redes bayesianas através da introdução de variáveis não-observadas. In *VII ENIA*.
- Srinivasan, A. (2001). *The Aleph Manual*.