



ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

Departamento de Engenharia de Computação e Sistemas Digitais

PCS2059

Inteligência Artificial

Redes Bayesianas e Inferência Exata

Gabriel Iseppe Porto	5174633
Raphael Petegrosso	5176451
Victor Tseimazides	5178828

Sumário

1	Introdução	3
2	Inferência por Enumeração	4
3	O Algoritmo de Eliminação de Variáveis	7
4	A Complexidade da Inferência Exata	9
5	Algoritmos de Agrupamento	9
6	Conclusões.....	10

1 Introdução

As Redes Bayesianas são grafos que tem por objetivo representar informações probabilísticas quantitativas para variáveis discretas ou contínuas.

De modo simplificado, o grafo representa as probabilidades condicionais entre nós filhos e pais, ou seja, $P(X_i | \text{Parents}(X_i))$, que quantifica o efeito dos nós pais em seus filhos.

A notação $P(a | b)$ simboliza a probabilidade condicional de $A = \text{true}$ em relação a $B = \text{true}$, ou seja, calcula a probabilidade de A ocorrer, dado que B teve sucesso. Para calcular esta probabilidade condicional, utiliza-se a seguinte equação:

$$P(a | b) = P(a \wedge b) / P(b)$$

EQUAÇÃO 1

A Figura 1 mostra um exemplo de Rede Bayesiana

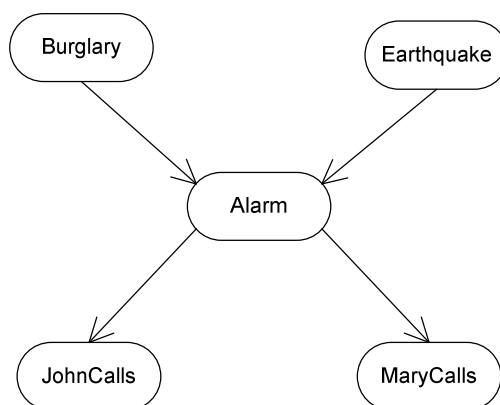


Figura 1 – Exemplo de Rede Bayesiana

Nesta rede ilustramos a seguinte situação: imagine que você comprou um alarme para sua casa que pode ser disparado em 2 situações: quando ocorre um terremoto e quando um ladrão tenta entrar na sua casa. Além disso, seus dois vizinhos John e Mary ligarão para o seu celular caso ouçam o alarme. Obviamente a situação não conta com o teor da conversa no telefone, mas sim com o fato de ser identificada uma ligação.

As probabilidades que esses eventos ocorram serão a seguinte, dados que, como visto da Figura 1, Burglary (B) e Earthquake (E) não tem probabilidade condicional, Alarm (A) é condicional a B e E, JohnCalls (J) é condicional a A e por fim MaryCalls (M) é condicional a A:

$$P(B) = 0.001; P(E) = 0.002$$

B	E	P(A)	A	P(J)	A	P(M)
true	true	0.95	true	0.90	true	0.70
true	false	0.94	false	0.05	false	0.01
false	true	0.29				
false	false	0.001				

A utilização de inferência em Redes Bayesianas se refere à tarefa de determinar a distribuição de probabilidade posterior de um grupo de variáveis X . Para isso, utilizamos um conjunto de variáveis de evidência E , ou seja, eventos observados, e um conjunto de variáveis não-

evidenciadas Y , chamadas de variáveis escondidas. Desta forma, uma inferência seria questionar qual a distribuição de probabilidade posterior $P(X | e)$.

No exemplo dado, uma inferência a ser resolvida seria calcularmos qual a distribuição de probabilidade posterior de um ladrão estar tentando assaltar a sua casa, dado que John e Mary fizeram uma ligação para você. Em outras palavras, realizarmos o seguinte cálculo:

$$P(\text{Burglary} | \text{JohnCalls} = \text{true}, \text{MaryCalls} = \text{true})$$

A seguir, discutiremos a dificuldade de calcular essa probabilidade de forma exata, ou seja, sem utilizar algoritmos numéricos de aproximação.

2 Inferência por Enumeração

Inicialmente é importante termos em mente a seguinte Equação 2, que representa a probabilidade de uma série de eventos ocorrer, dados seus condicionais, ou seja, seus parents no grafo da Rede Bayesiana:

$$P(x_1, \dots, x_n) = \prod_{i=1}^n P(x_i | \text{parents}(X_i)) ,$$

EQUAÇÃO 2

Além disso, para calcularmos $P(X | e)$ podemos utilizar a seguinte equação:

$$P(X|e) = a P(X, e) = a \sum_Y P(X, e, y)$$

EQUAÇÃO 3

Nesta equação vemos que podemos transformar a probabilidade condicional ao evento e em uma somatória de probabilidades que leva em conta todos os valores da variável escondida y , ou seja, $a \cdot P(X, e, y) + a \cdot P(X, e, \sim y)$, onde a é uma constante de normalização.

Voltando ao exemplo dado e considerando a Equação 3 citada, podemos calcular a probabilidade $P(B | j, m)$ da seguinte forma:

$$P(B|j, m) = \alpha P(B, j, m) = \alpha \sum_e \sum_a P(B, e, a, j, m) .$$

Onde as variáveis escondidas, neste caso, são e e a , uma vez que tanto John quanto Mary dependem do Alarm e o Alarm, por sua vez, depende do Earthquake e do Burglary, estando tanto o Alarm quando o Earthquake escondidos no problema. Em outras palavras, não importa se houve terremoto ou não e não importa se o alarme tocou ou não. Simplesmente queremos saber se há um ladrão em casa, dado que John e Mary fizeram uma ligação. Utilizando a Equação 2 na probabilidade final temos o seguinte:

$$P(b|j, m) = \alpha \sum_e \sum_a P(b)P(e)P(a|b, e)P(j|a)P(m|a)$$

Por fim, isolando as constantes que não fazem parte das somatória temos o seguinte:

$$P(b|j, m) = \alpha P(b) \sum_e P(e) \sum_a P(a|b, e)P(j|a)P(m|a) .$$

Assim, chegamos a uma equação que mostra que a probabilidade posterior desejada pode ser calculada através de um produtório que tem como fatores somatórias em relação as variáveis escondidas. Para facilitar, podemos dizer que a equação pode ser expandida para o seguinte:

$$P(b | j, m) = \alpha (P(b) \cdot (P(e) \cdot (P(a | b, e) \cdot P(j | a) \cdot P(m | a) + P(\neg a | b, e) \cdot P(j | \neg a) \cdot P(m | \neg a)) + P(\neg e) \cdot (P(a | b, \neg e) \cdot P(j | a) \cdot P(m | a) + P(\neg a | b, \neg e) \cdot P(j | \neg a) \cdot P(m | \neg a))))$$

Podemos também simbolizar esta expressão através da seguinte árvore, que facilita a visualização, onde as probabilidades de cada ramo foram determinadas através das tabelas de probabilidades mostradas na definição do problema:

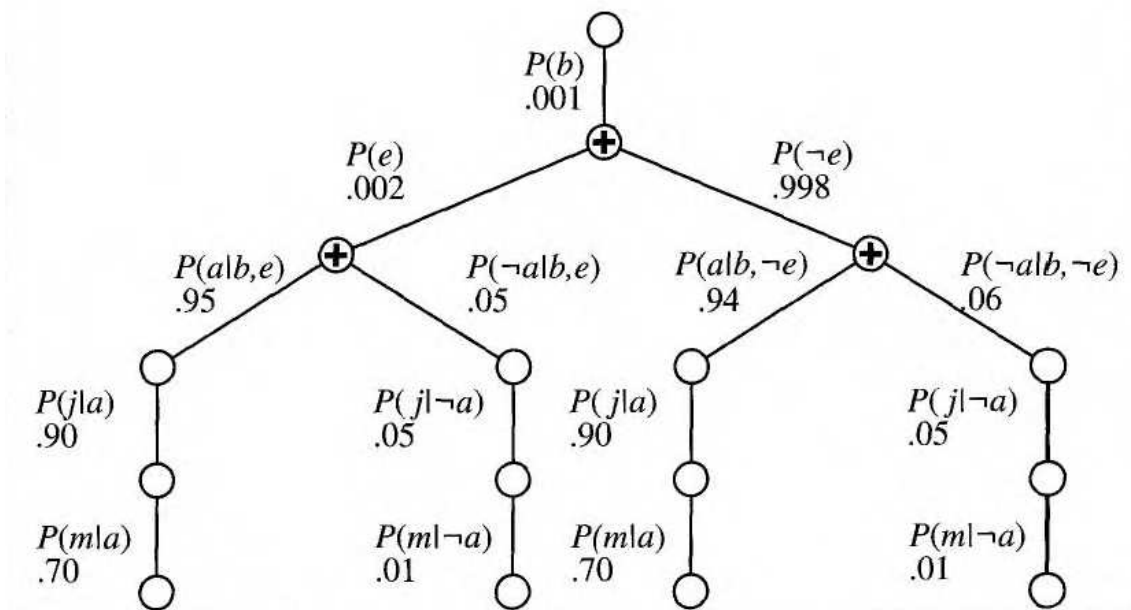


Figura 2 – Árvore de Cálculo da Inferência Exata

É importante termos em mente que as variáveis que estamos tratando são booleanas e, dessa forma, a expansão da somatória seria maior que esta. Além disso, estamos considerando poucas variáveis.

Realizando o cálculo teremos que $P(b | j, m) = \alpha \cdot 0,00059224$. Para normalizar o valor, fazemos o mesmo cálculo para $P(\neg b | j, m)$, que nos dará $P(\neg b | j, m) = \alpha \cdot 0,0014919$.

Assim, normalizando os valores temos:

$$P(b | j, m) = 0,00059224 / (0,00059224 + 0,0014919) = 0,284 \text{ e}$$

$$P(\neg b | j, m) = 0,0014919 / (0,00059224 + 0,0014919) = 0,716.$$

Concluimos, portanto, que há 28,4% de chance de haver um ladrão em casa, dado que John e Mary fizeram uma ligação.

Um algoritmo capaz de percorrer uma Árvore de Cálculo de Inferência Exata, como a do exemplo anterior, é o ENUMERATION – ASK , mostrado abaixo.

```

function ENUMERATION-AsK(X, e, bn) returns a distribution over X
  inputs: X, the query variable
           e, observed values for variables E
           bn, a Bayes net with variables  $\{X\} \cup E \cup Y$  / * Y = hidden variables */
  Q(X) <- a distribution over X, initially empty
  for each value xi of X do
    extend e with value xi for X
    Q(xi) <- ENUMERATE-ALL(VARS[bn], e)
  return NORMALIZE(Q(X))

function ENUMERATE-ALL(vars, e) returns a real number
  if EMPTY?(vars) then return 1.0
  Y <- First(vars)
  if Y has value y in e
    then return  $P(y / \text{Parents}(Y)) \times \text{ENUMERATE-ALL}(\text{REST}(\text{vars}), e)$ 
    else return  $\sum_y P(y / \text{Parents}(Y)) \times \text{ENUMERATE-ALL}(\text{REST}(\text{vars}), e)$ 
    where ey is e extended with Y = y

```

Figura 3 – Algoritmo ENUMERATION - ASK

Neste algoritmo a complexidade de processamento cresce linearmente de acordo com as variáveis acrescentadas à Rede Bayesiana, porém, a complexidade em tempo de processamento cresce na ordem de 2^n onde n é o número de variáveis. Na figura abaixo nota-se facilmente que para o cálculo de, por exemplo, $P(b \mid j, m)$, o algoritmo calcula duas vezes o valor de $P(j \mid \neg a) \cdot P(m \mid \neg a)$, o que representa desperdício de tempo.

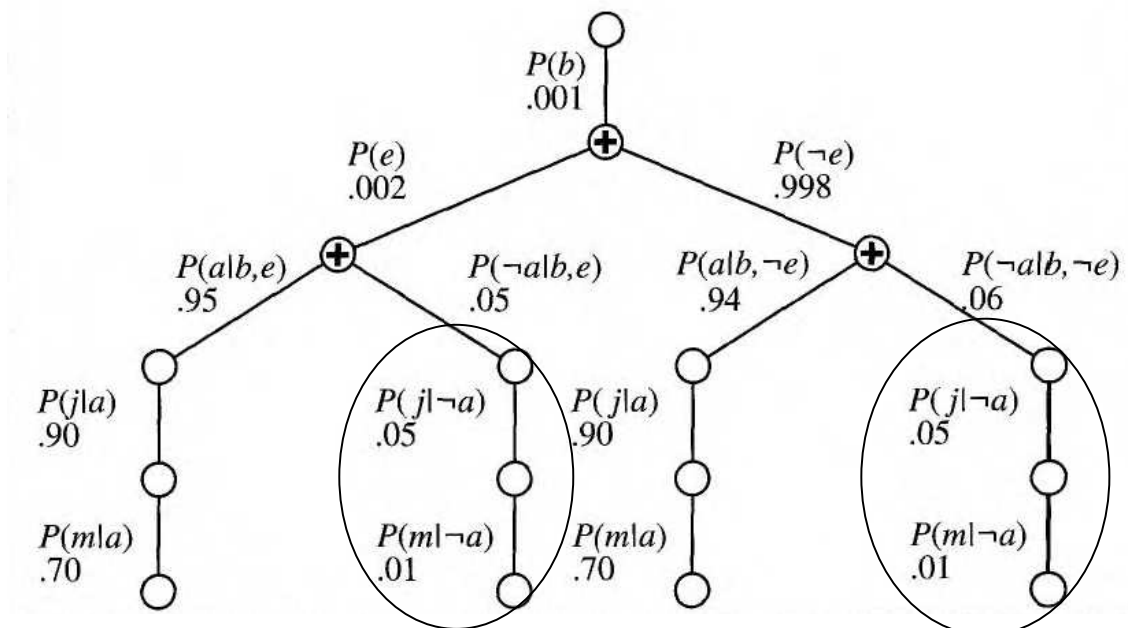


Figura 4 – Cálculo repetido no Enumeration-Ask

Uma solução intuitiva para este problema seria calcular o valor de uma determinada subexpressão e em seguida salvar o valor calculado para ser utilizado posteriormente. Uma forma de se fazer isso é através do algoritmo de Eliminação de Variáveis, explicado adiante.

3 O Algoritmo de Eliminação de Variáveis

Olhando-se para a equação abaixo, retirada do último exemplo, a maneira convencional de se avaliá-la é da esquerda para a direita, como se partíssemos do nó raiz até atingirmos a base. Tal procedimento leva ao problema citado anteriormente da múltipla inferência de subexpressões, que pode ser evitado se avaliarmos a expressão da direita para esquerda (*bottom – up*), armazenando os valores intermediários e eliminando da soma os valores com independem da variável analisada.

$$P(b|j, m) = \alpha P(b) \sum_e P(e) \sum_a P(a|b, e) P(j|a) P(m|a) .$$

Mostraremos agora, passo a passo, como o algoritmo funciona para o caso onde queremos saber qual a probabilidade de uma invasão ter ocorrido dado que Mary e John ligaram.

- Representamos cada parte da equação acima por letras maiúsculas, por exemplo:

$$M = P(m | a)$$

$$J = P(j | a)$$

$$A = P(a | b, e)$$

$$E = P(e)$$

$$B = P(b)$$

- Para cada letra associamos um fator que armazenará a probabilidade da variável representada através de uma matriz como feito a seguir:

$$f_M(A) = \begin{pmatrix} P(m|a) \\ P(m|\neg a) \end{pmatrix}$$

Repare que a probabilidade de M ocorrer depende somente de tocar ou não o alarme, o que nos leva a um vetor de dois elementos, sendo o primeiro a probabilidade de m sendo que o alarme tocou, e o segundo a probabilidade de m sendo que o alarme não tocou. O mesmo ocorre para J.

No caso de $f_A(A, B, E)$ teremos uma matriz $2 \times 2 \times 2$.

- Sabendo $f_A(A, B, E)$, $f_J(A)$ e $f_M(A)$ devemos agora calcular a soma sobre a desses três termos:

$$\begin{aligned}
\mathbf{f}_{\bar{A}JM}(B, E) &= \sum_a \mathbf{f}_A(a, B, E) \times \mathbf{f}_J(a) \times \mathbf{f}_M(a) \\
&= \mathbf{f}_A(a, B, E) \times \mathbf{f}_J(a) \times \mathbf{f}_M(a) \\
&\quad + \mathbf{f}_A(\neg a, B, E) \times \mathbf{f}_J(\neg a) \times \mathbf{f}_M(\neg a)
\end{aligned}$$

Observação: Para multiplicar os fatores não se utiliza a multiplicação de matrizes, e sim um método chamado *pointwise pruduct* que será explicado mais para a frente

- Realizamos o mesmo procedimento em E resultado em:

$$\begin{aligned}
\mathbf{f}_{\bar{E}\bar{A}JM}(B) &= \mathbf{f}_E(e) \times \mathbf{f}_{\bar{A}JM}(B, e) \\
&\quad + \mathbf{f}_E(\neg e) \times \mathbf{f}_{\bar{A}JM}(B, \neg e)
\end{aligned}$$

- Finalmente podemos calcular $P(b | j, m)$ apenas multiplicando $P(b)$ pelo fator $\mathbf{f}_{\bar{E}\bar{A}JM}(B)$.

$$\mathbf{P}(B|j, m) = \alpha \mathbf{f}_B(B) \times \mathbf{f}_{\bar{E}\bar{A}JM}(B)$$

Explicaremos agora o método, já citado, para multiplicar os fatores, chamado de *pointwise pruduct*.

O *pointwise pruduct* representa nada mais do que a união de dois fatores, por exemplo, sendo $f_1(A, B)$ e $f_2(B, C)$, para $f_1 \times f_2$ teremos $f_3(A, B, C)$, como podemos notar na figura a seguir:

A	B	$\mathbf{f}_1(A, B)$	B	C	$\mathbf{f}_2(B, C)$	A	B	C	$\mathbf{f}_3(A, B, C)$
T	T	.3	T	T	.2	T	T	T	$.3 \times .2$
T	F	.7	T	F	.8	T	T	F	$.3 \times .8$
F	T	.9	F	T	.6	T	F	T	$.7 \times .6$
F	F	.1	F	F	.4	T	F	F	$.7 \times .4$
						F	T	T	$.9 \times .2$
						F	T	F	$.9 \times .8$
						F	F	T	$.1 \times .6$
						F	F	F	$.1 \times .4$

Figura 5 – Tabela de Pointwise Product

Com rotinas para realizar de adaptação de somatórias e de *Pointwise Product* o algoritmo de eliminação de variáveis pode ser implementado como a seguir:


```

function ELIMINATION-AsK(X, e, bn) returns a distribution over X
    inputs: X, the query variable
             e, evidence specified as an event
             bn, a Bayesian network specifying joint distribution  $P(X_1, \dots, X_n)$ 

    factors  $\leftarrow []$ ; vars  $\leftarrow \text{REVERSE}(\text{VARS}[bn])$ 
    for each var in vars do
        factors  $\leftarrow [ \text{MAKE-FACTOR}(var, e)(factors) ]$ 
        if var is a hidden variable then factors  $\leftarrow \text{SUM-OUT}(Var, factors)$ 
    return NORMALIZE(POINTWISE-PRODUCT(factors))

```

Figura 6 – Algoritmo de Eliminação de Variáveis

4 A Complexidade da Inferência Exata

O algoritmo de Eliminação de Variáveis é mais eficiente que a inferência por enumeração por evitar computações repetidas, além de descartar variáveis irrelevantes. Os requisitos de tempo e espaço da eliminação de variáveis são dominados pelo tamanho do maior fator construído durante a operação do algoritmo. Este, por sua vez, é determinado pela ordem de eliminação de variáveis e pela estrutura da rede.

A rede bayesiana do exemplo mostrada na Figura 1 contém apenas um caminho não-direcionado entre quaisquer dois nós da rede. Redes deste tipo são chamadas de **singularmente conectadas** (singly connected networks) ou **poli-árvores** (polytrees), e possuem uma propriedade particular: a complexidade espacial e temporal da inferência exata nelas é linear no tamanho da rede. Aqui, o tamanho é definido como o número de entradas na tabela de probabilidades condicionais (CPT); se o número de pais de cada nó é delimitado por uma constante, então a complexidade também será linear no número de nós. Estes resultados valem para qualquer ordenação coerente com a ordenação topológica da rede.

Para redes **multiplamente conectadas** (multiply connected networks), a eliminação de variáveis pode apresentar complexidade exponencial de tempo e espaço no pior caso, mesmo quando o número de pais por nó é delimitado.

Existe uma estreita ligação entre a complexidade da rede bayesiana de inferência e a complexidade dos CSPs (problemas de satisfação de restrições). A dificuldade de resolver um CSP discreto está relacionada ao quanto a forma de seu grafo de restrições se parece com uma árvore. Medidas como **hypertree width**, que limitam a complexidade de resolver um CSP, também podem ser aplicadas diretamente a redes Bayesianas. Além disso, o algoritmo de eliminação de variáveis pode ser generalizado para resolver CSPs, bem como redes bayesianas.

5 Algoritmos de Agrupamento

O algoritmo de eliminação de variáveis é simples e eficiente para responder a consultas individuais. Entretanto, para se calcular probabilidades posteriores para todas as variáveis em uma rede, ele pode ser menos eficiente. Por exemplo, em uma rede poli-árvore, seria preciso fazer $O(n)$ consultas custando $O(n)$ cada uma, perfazendo um total de $O(n^2)$ tempo. Usando algoritmos de

agrupamento (também conhecidos como algoritmos **joint tree**), o tempo pode ser reduzido para $O(n)$. Por esta razão, estes algoritmos são amplamente utilizados em ferramentas comerciais de redes bayesianas.

A idéia básica do agrupamento é juntar nós individuais da rede de modo a formar nós agrupados, de tal forma que o resultado é uma rede poli-árvore. Por exemplo, a rede multiplamente conectada mostrada na Figura 7(a) pode ser convertida em uma poli-árvore combinando os nós *Sprinkler* e *Rain* em um nó agrupado chamado *Sprinkler+Rain*, como mostrado na figura 7(b). Os dois nós booleanos são substituídos por um nó maior que pode assumir quatro valores: TT, TF, FT, e FF. Este nó tem apenas um pai, a variável booleana *Cloudy* e, portanto, existem dois casos condicionados.

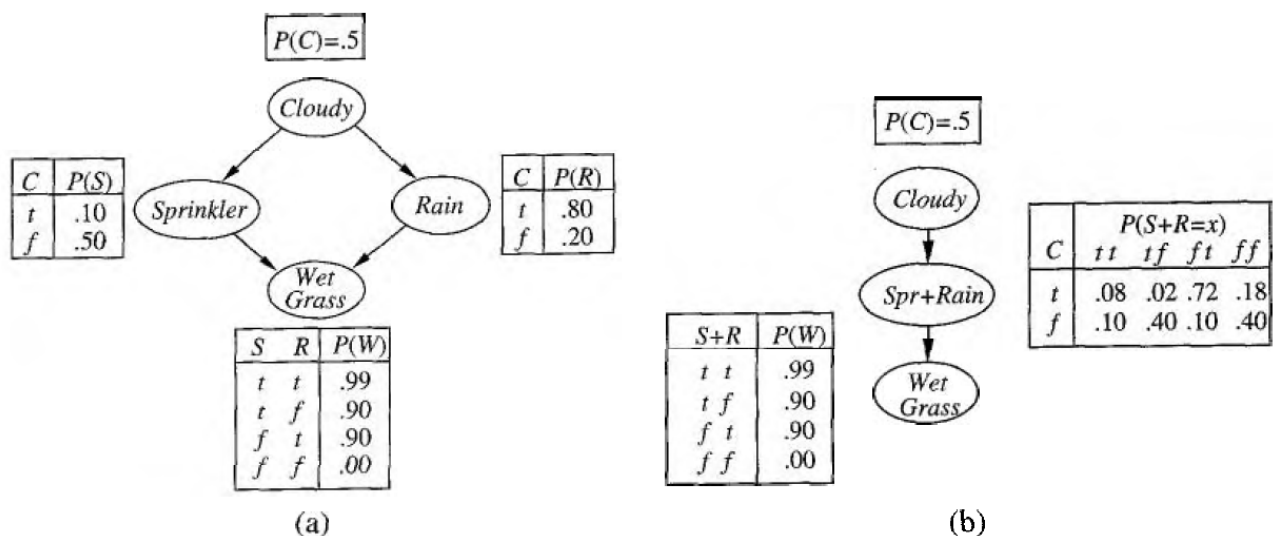


Figura 7 – (a) rede multiplamente conectada com suas CPTs. (b) equivalente agrupado da rede anterior.

Assim que a rede está na forma poli-árvores, um algoritmo específico de inferência é aplicado. Esse algoritmo é capaz de calcular probabilidades posteriores para todos os nós não-evidência na rede em tempo $O(n)$, onde n é agora o tamanho da rede modificada. No entanto, a complexidade NP do problema não desaparece: se uma rede requer tempo e espaço exponenciais com a eliminação de variáveis, então os CPTs na rede agrupada irão requerer tempo e espaço exponenciais para construir.

6 Conclusões

Redes bayesianas são úteis para resolver problemas de teoria da probabilidade tradicional. O seu uso não requer muitos números, de forma a trazer métodos eficientes de solução exata, bem como uma variedade de métodos de aproximação.