Bases Formales de la Computación: Redes de Bayes (segunda parte)

Prof. Gloria Inés Alvarez V.

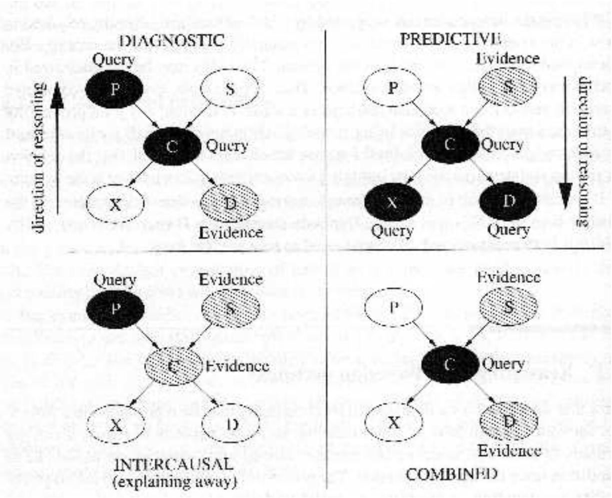
Departamento de Ciencias e Ingeniería de la Computación
Pontificia Universidad Javeriana Cali

Periodo 2008-2

Razonamiento en Redes de Bayes

- Las RB ofrecen una representación completa de las distribuciones de probabilidad sobre sus variables.
- Se pueden plantear condicionales sobre cualquier subconjunto de variables.
- Soporta razonamiento en cualquier dirección.
- Considerar el efecto de cierta evidencia sobre la red, consiste en calcular la distribución de probabilidad a posteriori de un conjunto de nodos de consulta dados los valores de dicha evidencia.

Diagrama sobre tipos de razonamiento[1]

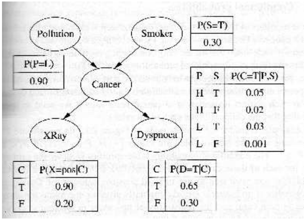


[1] Tomado de Bayesian Artificial Intelligence

Tipos de Evidencia

- Definida: cuando se conoce el valor preciso de una variable
- No definida: cuando se conoce que el valor de una variable puede ser uno de varios posibles descartando el resto de los valores permitidos para la variable.
- Negativa: se sabe que una variable no puede tomar determinado valor.
- Virtual: se conocen indicios que modifican la distribución a posteriori de la variable.

Ejemplo de razonamiento numérico[2]



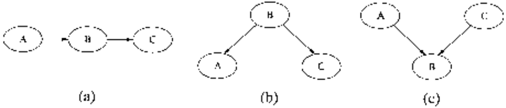
Node P(S)=0.3	No Evidence	Reasoning Case Diagnostic D=T	Predictive S=T	Intercausal C=T	C=T S=T	Combined D=T S=T
Bel(P=high)	0.100	0.102	0.100	0.249	0.156	0.102
Bel(S=T)	0.300	0.307	1	0.825	1	1
Bel(C=T)	0.011	0.025	0.032	1	1	0.067
Bel(X=pos)	0.208	0.217	0.222	0.900	0.900	0.247
Bel(D=T)	0.304	1	0.311	0.650	0.650	1
P(S)=0.5						
Bel(P=high)	0.100	0.102	0.100	0.201	0.156	0.102
Bel(S=T)	0.500	0.508	1	0.917	1	1
Bel(C=T)	0.174	0.037	0.032	1	1	0.067
Bel(X=pos)	0.212	0.226	0.311	0.900	0.900	0.247
Bel(D=T)	0.306	1	0.222	0.650	0.650	1

[2] Tomado de Bayesian Artificial Intelligence

Independencia Condicional

- Una red de Bayes, que cumple la propiedad de Markov, se puede considerar un mapa de independencia (I-maps) entre las variables, ya que la ausencia de arco entre dos variables corresponde a la independencia real entre esas variables en el sistema.
- La aparición de evidencia modifica la relación de dependencia/independencia que tienen inicialmente las variables. Calcular si dos variables, o dos conjuntos de variables son independientes entre sí, dada una evidencia no es un problema trivial. En principio se pueden dar tres casos generales.

Casos Generales de Dependencia Condicional[3]



- Cadena causal: $P(C|B) = P(C|A, B)$ si sabemos A esto afecta la probabilidad de C.
- Causas comunes: si sabemos B, A y C son independientes.
- Efectos comunes: $P(A|C, D) \neq P(A|C) \equiv \neg(A \parallel C|B)$

[3]Tomado de Bayesian Artificial Intelligence

D-separación

Sean X, Y conjuntos de variables aleatorias dentro de una red de Bayes.

Definición

Un camino entre las variables X y Y va de un nodo de X a un nodo de Y sin tomar en cuenta la dirección de las fechas y sin pasar varias veces por un mismo nodo.

D-separación

Definición

Un camino está bloqueado, dado un conjunto de variables de evidencia E, si existe un nodo Z en el camino que cumple una de las siguientes condiciones:

- $Z \in E$ *y* Z *tiene un arco que entra a él y otro que sale de él.*
- $Z \in E$ *y* Z *tiene dos arcos que salen de él.*
- $Z \notin E$ *ni ninguno de sus descendientes, ambos arcos llegan a* Z

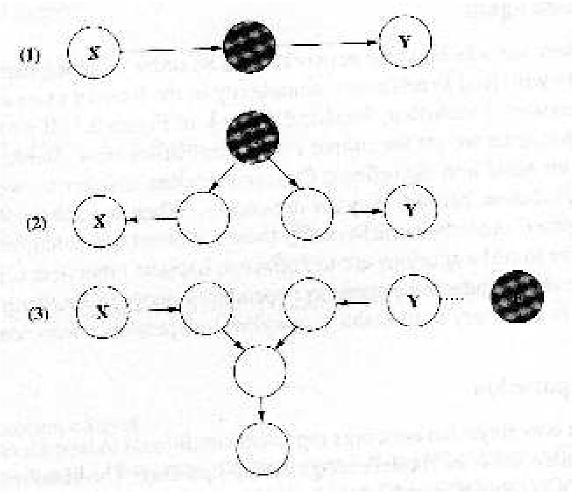
D-separación

Definición

Un conjunto de nodos E d$-$separa dos conjuntos de nodos X, Y si todo camino desde X hasta Y está bloqueado por E.

Si X, Y están d$-$separados por E, entonces X, Y son condicionalmente independientes dado E y dado que se cumple la propiedad de Markov.

D-separación[4]



[4]Tomado de Bayesian Artificial Intelligence

Algoritmos de Inferencia para Redes de Bayes

El problema de la inferencia en redes de Bayes general es NP-hard.

Los algoritmos existentes se pueden agrupar con diversos criterios.

En cuanto a la exactitud de la solución:

- Exactos: calculan completamente las distribuciones de probabilidad a posteriori para todas las variables.
- Aproximados

En cuanto a la topología de la red:

- Sobre poliárboles (hay algoritmo polinomial)
- Sobre grafos con *loops*

Algoritmo de Kim y Pearl por Paso de Mensajes

- Es un algoritmo exacto y funciona sobre árboles y poliárboles.
- Requiere mantener tres tipos de parámetros que almacenan: la fuerza de soporte predictivo de las entradas, de soporte diagnóstico de las salidas y la tabla de probabilidades condicionales de un nodo X.
- El algoritmo consta de tres pasos:
  - Actualización de las creencias: se activa por la llegada de un mensaje desde un padre o un hijo avisando que han cambiado los parámetros de creencia.
  - Propagación bottomup: el nodo computa nuevos mensajes para enviar a sus padres.
  - Propagación topdown: el nodo computa nuevos mensajes para enviar a sus hijos.

Algoritmo de Kim y Pearl por Paso de Mensajes

Definición

Definimos:

- *La constribución a la fuerza de soporte predictivo actual de cada enlace de entrada* $U_i \rightarrow X$*, como:* $\pi_X(U_i) = P(U_i|E_{U_i \backslash X})$ *(evidencia sin contar* X*)*
- *La contribución a la fuerza de soporte diagnóstico* λ *de cada enlace de salida* $X \rightarrow Y_j$*, como:* $\lambda_{Y_j}(X) = P(E_{Y_j \backslash X}|X)$
- *La tabla fija* $P(X|U_i, \ldots, U_n)$

Estos valores se utilizan para la actualización de la crencia así:

$$Bel(x_i) = \alpha\lambda(x_i)\pi(x_i)$$

Donde α es una constante de normalización y las funciones se calculan de la siguiente manera:

$$\lambda(x_i) = \begin{cases} 1 & \text{if evidence is } X = x_i \\ 0 & \text{if evidence is for another } x_j \\ \pi_j \lambda_{Y_j}(x_i) & \text{otherwise} \end{cases}$$

$$\pi(x_i) = \sum_{u_1,\ldots,u_n} P(x_i|u_1,\ldots,u_n) \prod_i \pi_X(u_i)$$

Algoritmo de Kim y Pearl por Paso de Mensajes

En el paso de propagación bottomup, cada nodo calcula nuevos mensajes π para enviar a sus padres:

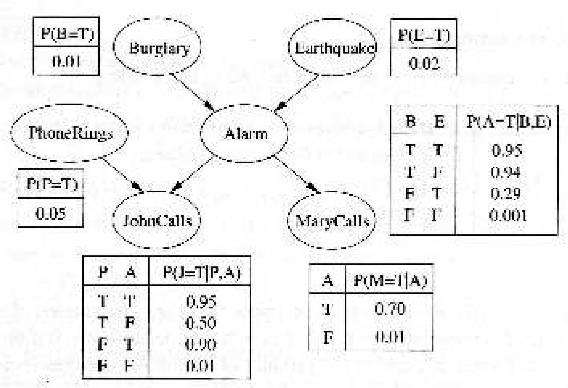
$$\lambda_X(u_i) = \sum_{x_i} \lambda(x_i) \sum_{u_k : k \neq i} P(x_i | u_1, \ldots, u_n) \prod_{k \neq i} \pi_X(u_k)$$

En el paso de propagación topdown el nodo computa nuevos mensajes λ para enviar a sus hijos:

$$\pi_{Y_j}(x_i) = \begin{cases} 1 & \text{si entra } x_i \\ 0 & \text{si entra otro } x_j \\ \alpha[\prod_{k \neq j} \lambda_{Y_k}(x_i)] & \text{sinó} \\ \sum_{u_1,\ldots,u_n} P(x_i | u_1, \ldots, u_n) \prod_i \pi_X(u_i) & \end{cases}$$

Algoritmo de Kim y Pearl por Paso de Mensajes[5]

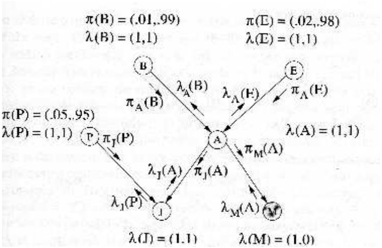
Ejemplo sobre el flujo de los mensajes:

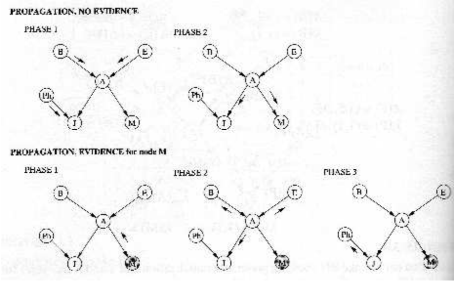


[5] Tomado de Bayesian Artificial Intelligence

Algoritmo de Kim y Pearl por Paso de Mensajes[6]

Ejemplo sobre el flujo de los mensajes:





[6]Tomado de Bayesian Artificial Intelligence

Algoritmo de Kim y Pearl por Paso de Mensajes

Características del algoritmo:

- Todos los cálculos de paso de mensaje se hacen localmente.
- El algoritmo calcula una suma sobre todas las conjuntas de los nodos padres, que es exponencial en el número de padres del nodo, así que el algoritmo no es práctico cuando crece el número de padres de un nodo.

Otras aproximaciones

- Métodos de conglomerado, como los árboles de cruce
- Métodos de simulación estocástica, como el muestreo lógico

Bibliografía

- Bayesian Artificial Intelligence. K. B. Korb, Ann E. Nicholson. Chapman & Hall/CRC. 2004.
- Inteligencia Artificial: un enfoque moderno. S. J. Russell, P. Norving. Prentice Hall. 1996.