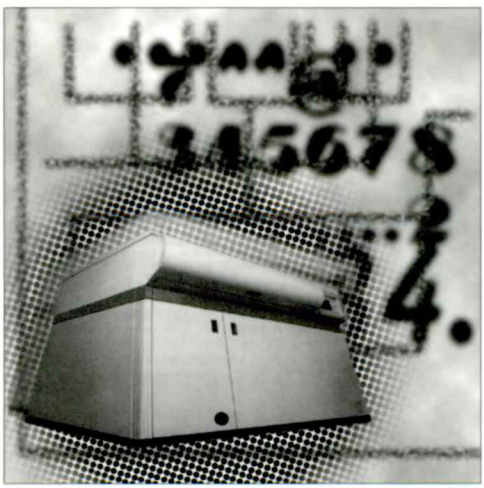
Coordinadores
José Antonio Gámez Martín
José Miguel Puerta Callejón

SISTEMAS EXPERTOS PROBABILÍSTICOS



COLECCION CIENCIA Y TECNICA

SISTEMAS EXPERTOS PROBABILÍSTICOS

SISTEMAS EXPERTOS PROBABILÍSTICOS

Coordinadores:

JOSÉ ANTONIO GÁMEZ MARTÍN
JOSÉ MIGUEL PUERTA CALLEJÓN

Ediciones de la Universidad
de Castilla-La Mancha
Cuenca 1998

SISTEMAS expertos probabilísticos / Coordinadores, José Antonio Gámez Martín, José Miguel Puerta Callejón.– Cuenca : Ediciones de la Universidad de Castilla-La Mancha, 1998
318 p. ; 22 cm.– (Ciencia y Técnica ; 20)
ISBN 84-89958-35-1
Actas del Curso de Verano de la U.C.L.M. que, con igual título, se desarrolló en Albacete en julio de 1998
1. Programas y sistemas de programación - Informática - Estudios y conferencias 2. Inteligencia artificial - Estudios y conferencias I. Gámez Martín, José Antonio, coord. II. Puerta Callejón, José Antonio, coord. III. Universidad de Castilla-La Mancha, ed. IV. Título V. Serie
681.3.06:007.52(063)

EDITA: Servicio de Publicaciones de la Universidad de Castilla-La Mancha.
Director: Pedro C. Cerrillo
Colección CIENCIA Y TÉCNICA, Nº 20.
1ª edición: junio de 1998. Tirada: 500 ejemplares.

Diseño de la colección: García Jiménez.
Diseño de la cubierta: C.I.D.I. (Universidad de Castilla-La Mancha).

Impresión y Encuadernación: Gráficas Cuenca, S.A. Avda. Juan Carlos I, 34 - 16004 Cuenca.

I.S.B.N.: 84-89958-35-1
D.L. CU - 131 - 1998

Impreso en España - *Printed in Spain*

VIII Curso de Verano de Informática

Universidad de Castilla-La Mancha

Director

D. Isidro Ramos Salavert

Organiza

Departamento de Informática de la Escuela

Universitaria Politécnica de Albacete

Coordinadores

D. Luis Miguel de Campos Ibáñez
D. Serafín Moral Callejón

Comité Organizador

D. José Antonio Gámez Martín
D. José Miguel Puerta Callejón
D. Francisco José Vigo Bustos

Presentación

Como ya es habitual por estas fechas, la sección del campus de Albacete del Departamento de Informática de la Universidad de Castilla-La Mancha hace un esfuerzo por organizar un encuentro en torno a un tema de puntera actualidad en el campo de la Informática.

Este año, el Comité Organizador ha logrado reunir en nuestra ciudad a prestigiosos investigadores nacionales que nos ilustrarán en el siempre interesante tema de los *Sistemas Expertos Probabilísticos*. La Escuela de Verano no pretende ser sólamente un foro en el que los distintos ponentes planteen su visión sobre la materia, sino que además y nos atreveríamos a decir, sobre todo, ha de ser un curso que introduzca al estudiante en el tema.

Hemos de dar nuestro sincero agradecimiento a D. Isidro Ramos Salavert por presidir un año más esta Escuela de Verano, a los ponentes por el trabajo realizado para hacer que la misma sea hoy una realidad, y como no, a D. Serafín Moral Callejón y a D. Luis M. de Campos Ibáñez por su trabajo de coordinación, así como por el interés mostrado en el proyecto desde un principio.

Por último, y no por ello menos importante agradecer la colaboración prestada por las instituciones de nuestra ciudad.

Comité Organizador EVI98

Prólogo

La incorporación de tareas rutinarias en el ordenador marcó el desarrollo inicial de la Informática. Lo algorítmico encontraba campo abonado en ella.

Pronto se advirtió que la frontera entre lo creativo y lo rutinario no era nítida y que muchas actividades consideradas como específicamente humanas eran soportables sobre un ordenador usando modelos adecuados. La frontera entre lo creativo y lo rutinario se fue inexorablemente desplazando en el sentido de convertir en rutinarias actividades consideradas como creativas previamente.

Los sistemas expertos intentan capturar el "saber hacer"de un experto (un médico por ejemplo) en una máquina que se convierte así en una magnífica ayuda en la toma de decisiones. El tipo de conocimiento de un experto suele ser impreciso, vago, no estríctamente algorítmico y el tratamiento de este aspecto se ha abordado desde el marco de la Lógica y el Algebra Difusas o desde aproximaciones probabilísticas.

La Escuela de Verano de Informática (EVI98) de la UCLM siguiendo ya una tradición sólidamente establecida, aborda este año el tema monográfico : "SISTEMAS EXPERTOS PROBABILISTICOS". Un conjunto de expertos presentan en Julio el estado del arte del tema y este libro recoge el conjunto de sus trabajos.

Isidro Ramos
Director EVI98

Índice General

Sistemas Expertos Probabilísticos: Modelos Gráficos

Juan F. Huete

Dpto. Ciencias de la Computación e Inteligencia Artificial
Universidad de Granada
Avda. Andalucía s/n
18071 Granada
correo-e: jhg@decsai.ugr.es

Resumen

Un sistema experto es una herramienta informática que es capaz de simular el comportamiento de un experto humano en una materia especializada. Un problema clave en el desarrollo de sistemas expertos es encontrar la forma de representar y usar el conocimiento que los expertos humanos en esa materia poseen y utilizan. Este problema se hace más díficil por el hecho de que, en muchos campos, el conocimiento de los expertos es a menudo impreciso o incierto y sin embargo, los expertos son capaces de llegar a conclusiones útiles.

Por tanto, todo sistema experto que pretenda razonar "como si" lo hiciese un ser humano debe de ser capaz de trabajar con este tipo de información. Uno de los formalismos más potentes y mejor desarrollados para el tratamiento del conocimiento incierto es la Teoría de la Probabilidad, que nos permite medir la creencia que tenemos en la ocurrencia de un determinado suceso.

En este trabajo presentamos un tipo particular de Sistema Experto Probabilístico: Las *Redes Bayesianas* que utilizan el lenguaje de los grafos dirigidos acíclicos para representar las relaciones de relevancia entre las variables. La fuerza de estas relaciones viene expresada mediante un conjunto de distribuciones de probabilidad condicionadas.

1 Sistemas Expertos

Para tratar de entender lo que es un Sistema Experto, imaginemos la siguiente situación:

Hemos comprado e instalado un paquete software que permite la edición de documentos. Sin embargo, cuando queremos enviar un determinado documento a la impresora aparecen los problemas y el documento no se imprime. ¿ Qué podemos hacer ?. Una posible solución consiste en llamar

a un amigo, del que conocemos que domina el producto que hemos comprado, y consultarle nuestro caso. Para identificar el fallo, nuestro amigo nos hará preguntas sobre los mensajes de error que aparecen, nos pedirá que realicemos algún tipo de pruebas o modificaciones en la instalación, etc. Finalmente, y como resultado de toda nuestra charla, tendremos la secuencia de pasos a realizar para resolver nuestro problema, es decir, que la impresora funcione.

Este tipo de situaciones son muy comunes el la vida diaria, donde para la resolución de un gran número de problemas necesitamos consultar a un experto. Sin embargo, hoy dia es posible pensar en el propio ordenador como una herramienta que, dotada de un software adecuado, puede ser de gran utilidad en la resolución de problemas. En este sentido, el ordenador podría actuar como 'aquel amigo' que nos permitía poner en funcionamiento la impresora. Para ello, será necesario el poder establecer un 'dialogo' con el ordenador, que éste se encarge de guiar la conversación con la finalidad de obtener el conjunto de fallos o detectar los síntomas que aparecen, que pueda realizar un diagnóstico del problema y finalmente que proporcione el tratamiento o conjunto de pasos que nos permitan solucionarlo.

Son muchas las formas posibles de realizar el proceso anteriormente descrito. Por ejemplo, el conjunto de síntomas puede ser obtenido mediante consultas a un usuario, utilizando sensores que tomen la información directamente del mundo real, o una combinación de ambos. De igual forma, el diagsnóstico del problema puede ser presentado como una ayuda al usuario o por el contrario que sea el propio sistema el que se encarge del control de aquellos elementos que lo resuelvan. En cualquier caso, al conjunto Hardware (ordenador, sensores, ...) y Software que nos permite resolver el problema lo podríamos considerar como un *experto* en la materia.

Podemos encontrar distintas definiciones de lo que es un **Sistema Experto**:

- [6] "Sistema informático que utiliza el conocimiento sobre un determinado dominio para alcanzar la solución ante un problema de ese dominio. Esta solución es esencialmente la misma que la que se obtendría por una persona con conocimiento de la materia cuando se enfrentase al mismo problema".
- [2] "Sistema informático que simula a los expertos humanos en un área de especialización dada".

De forma genérica, y considerando las definiciones anteriores, podemos pensar que la mayoría de los programas de ordenador son sistemas expertos ya que resuelven un problema concreto dentro de un dominio determinado. Sin embargo, para poder ser considerado como Sistema Experto, el sistema tiene que tener la capacidad de justificar y explicar la solución propuesta.

Este tipo de sistemas son de gran utilidad en aquellos dominios donde el número de expertos no es muy grande, o bien son muy costosos. También se pueden aplicar para resolver problemas cuando los datos, o la forma en la que los expertos humanos razonan, no están completamente determinadas o para obtener soluciones en aquellos problemas donde las bases teóricas aún estan incompletas.

1.1 Componentes de un Sistema Experto

En esta sección comentaremos las distintas componentes de un sistema experto. Antes de analizar cada una de estas componentes queremos hacer especial énfasis en el hecho de que para el usuario final el Sistema Experto no es mas que un programa con tres partes bien diferenciadas (representadas en la siguiente figura) que pasamos a analizar

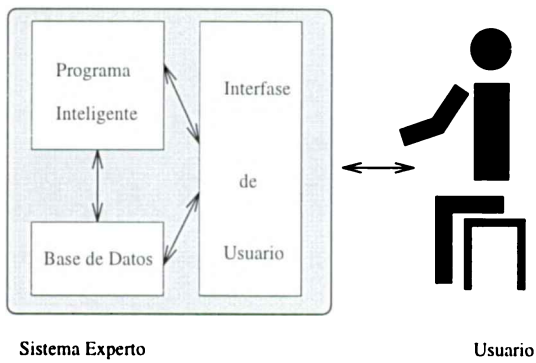


Figura 1. Visión del usuario de un Sistema Experto

- Interfase de usuario que proporciona una comunicación amigable con el sistema, siendo la encargada de gestionar las entradas y salidas del mismo, entre las que podemos encontrar las conclusiones obtenidas, las justificaciones que explican tales conclusiones, etc. Esta interfase puede ser gráfica, utilizando lenguaje natural o mediante el uso de menús.
- El programa inteligente, que de cara al usuario sólo es una caja negra que realiza las tareas de razonamiento y se encarga de obtener los resultados que necesita. El usuario final no tiene idea de cómo se realiza el razonamiento, y generalmente tampoco le interesa el conocerlo.

- La base de datos específica del problema que se está resolviendo, que incluye toda la información proporcionada por el usuario al sistema, la información obtenida de los sensores y todas las conclusiones que el programa inteligente ha sido capaz de obtener.

Desde el punto de vista de la persona que se encarga del desarrollo de sistemas expertos el esquema inicial se amplia, en particular el 'programa inteligente' que observaba el usuario final. En cualquier caso, una característica esencial de todo sistema experto es que se tiene una clara separación entre el conocimiento y la forma de utilizarlo. En la siguiente figura presentamos cada uno de estos módulos, que pasamos a detallar.

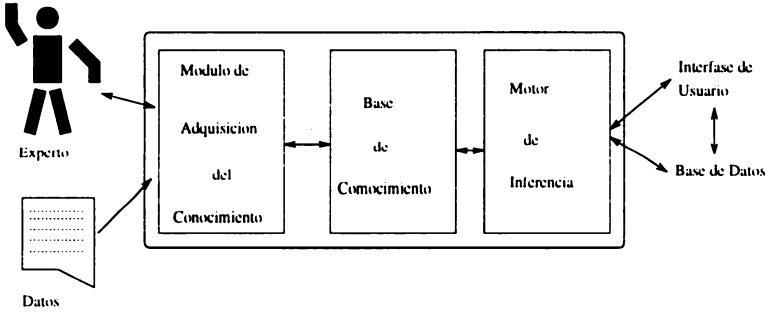


Figura 2. Componentes del Programa Inteligente

- La Base de Conocimiento es la parte más importante de un sistema experto. Incluye todo el conocimiento relevante que se tiene sobre el dominio del problema que estamos considerando. Podemos encontrar distintos formalismos para representar este conocimiento, como por ejemplo la lógica de predicados, reglas, distribuciones de probabilidad, etc.
- El Motor de Inferencia se encarga de obtener las conclusiones a partir de la información de la que dispone (almacenada en la base de datos y en la base de conocimiento). Este módulo se puede considerar como el cerebro del sistema experto.
- Módulo de adquisición de conocimiento se encarga de obtener la base de conocimiento. Cuando disponemos de un experto humano, éste módulo permite obtener la información necesaria y en el formato seleccionado. Sin embargo, son muchos los problemas para los que no disponemos de un experto. En este caso, podemos utilizar la información almacenada en una base de datos (u

obtenida mediante la repetición sucesiva de un experimento) con la finalidad de obtener la base de conocimiento.

Es muy importante tener clara la diferencia entre lo que son DATOS y CONOCIMIENTO. Los datos hacen referencia a una ejecución particular del sistema experto, tienen una validez temporal (la de la propia ejecución), destruyéndose al finalizar la aplicación. Por otro lado, el conocimiento expresa afirmaciones de validez general, teniendo una validez permanente. Por ejemplo, en un sistema experto médico la base de conocimiento almacena información del tipo *Si un paciente tiene fiebre es conveniente aplicarle un antitérmico*, mientras que en la base de datos se tienen hechos como que el paciente A.P.L. tiene una temperatura de 39.5^oC.

2 Sistemas Expertos Basados en Reglas

En general, los primeros sistemas expertos, por ejemplo Dendral ([12]: Obtiene estructuras moleculares a partir de análisis espectrales) y MYCIN ([18]: Diagnóstico y tratamiento de enfermedades de la sangre) pertenecían a este tipo de sistemas.

Las reglas nos permiten representar conocimiento del siguiente tipo:

Si *la temperatura es superior a* 37^oC **entonces** *el paciente tiene fiebre*
Si *un libro es anterior al siglo XVII* **y** *es raro* **entonces** *es un libro caro*

En general, las reglas son del tipo

SI *Condición* **ENTONCES** *Acción*

En este tipo de sistemas, la parte SI de la regla (también llamada premisa o antecedente) es testeada y en caso de ser cierta, la parte ENTONCES (también llamada acción o consecuente) se activa, dando como resultado un nuevo conjunto de hechos.

Por ejemplo, supongamos el siguiente hecho "El paciente A.P.L. tiene una temperatura de 39.5^oC". Si lo emparejamos con las reglas anteriores, tenemos que $(39.5^oC > 37^oC)$ es cierto y por tanto, podemos concluir que el paciente A.P.L. tiene fiebre.

Tanto en la *Condición* como en la *Acción* de una regla se pueden representar expresiones lógicas compuestas, conectadas por los operadores lógicos **y, o, no**, como por ejemplo:

SI A **y** (**no**) B **ENTONCES** C **y** D

Aún existiendo grandes inconvenientes para este tipo de sistemas, su popularidad, simplicidad y la similaridad con la forma de razonamiento humano hacen que sean una herramienta de gran utilidad para un conjunto amplio de problemas

En un sistema experto basado en reglas, el motor de inferencia se encarga de seleccionar de la base de conocimiento aquellas reglas que son aplicables. Para ello, empareja la *Condición* de las reglas con el conjunto de hechos (almacenados en la base de datos) y en caso de ser ciertos aplica las reglas obteniendo ('infiere') nuevos hechos que se incorporan a la base de datos. Repitiendo este proceso se produce un encadenamiento de conclusiones.

El principal problema que se plantea es el de crear un conjunto de inferencias que nos permita llegar desde la definición inicial de problema a la solución. En este sentido podemos encontrar dos estrategias principales:

1 Avanzar desde el conjunto de datos o hechos hacia las conclusiones o razonamiento hacia delante. La regla de inferencia necesaria para realizar este tipo de razonamiento es el MODUS PONENS que expresa la siguiente idea:

MODUS PONENS		
Regla:	**SI** A **Entonces** B	
Hechos:	A es cierto	
Conclusión:		B es cierto

2 Seleccionar una posible conclusión e intentar demostrar su validez encontrando algunas evidencias que la soporten o razonamiento hacia atrás. La regla de inferencia que se utiliza en este sentido es el MODUS TOLLENS:

MODUS TOLLENS		
Regla:	**SI** A **Entonces** B	
Hechos:		B es falso
Conclusión:	A es falso	

Ejemplo 1. Consideremos el siguiente problema, donde para obtener la calificación global de una asignatura se realizan dos evaluaciones y donde dicha calificación se establece con el siguiente conjunto de reglas:

Base de Conocimiento	
R1: Si (Nota-Practicas ≥ 5)	Entonces Prácticas-Aptas
R2: Si (Nota-Practica ≥ 4) y (Nota-Práctica < 5)	Entonces Prácticas-Cond.
R3: Si (Nota-Teoria ≥ 5)	Entonces Teoría-Aprobada.
R4: Si (Nota-Teoría ≥ 3) y (Nota-Teoría < 5)	Entonces Teoría-Cond.
R5: Si (Prácticas-Aptas) y (Teoria-Aprobada)	Entonces Aprobado.
R6: Si (Prácticas-Cond.) y (Teoría-Aprobada)	Entonces Aprobado.
R7: Si (Prácticas-Cond.) y (Teoría-Cond.)	Entonces Suspenso.

Supongamos que tenemos los siguientes hechos:

Base de Datos

Nota-Prácticas = 4.5 Nota-Teoría = 7

Aplicando el modus ponens (se consideran la base de datos actual y la base de conocimiento) tenemos que utilizando la regla R2 podemos inferir que Práctica-Cond es cierto y cuado lo aplicamos sobre R3 concluimos que la teoría de la asignatura está aprobada. Por tanto, la base de datos se transforma, incluyendo dos nuevos hechos:

Base de Datos

Nota-Prácticas = 4.5 Nota-Teoría = 7
Prácticas-Cond Teoría-Aprobada

Con estos dos nuevos hechos tenemos (usando R6) que la nota final de la asignatura sería Aprobado. □

Un esquema similar se puede considerar para el razonamiento hacia atrás.

Hay que destacar que el razonamiento hacia delante es especialmente interesante cuando partimos de un conjunto de datos no muy elevado (en comparación con el número de conclusiones posibles) y, de forma inversa, el razonamiento hacia atrás es especialmente útil cuando el número de conclusiones que se pueden obtener no es muy elevado.

Este tipo de sistemas se han utilizado para resolver problemas en una gran cantidad de dominios. Entre las razones de peso que justifican su uso, podemos encontrar

1. Su modularidad: Cada regla es una unidad independiente de conocimiento que puede ser añadida, modificada o eliminada independientemente del resto de las reglas existentes. Este hecho proporciona a este tipo de sistemas de una gran flexibilidad.
2. Su uniformidad: Todo el conocimiento del sistema es expresado con el mismo formato. Esto permite que la adquisición del conocimiento sea una tarea más fácil

3. La naturalidad: El expresar el conocimiento en forma de reglas se aproxima a la forma de razonar de los expertos humanos.

Aún siendo muchas sus ventajas, este tipo de sistemas presentan inconvenientes, como por ejemplo:

1. Mantenimiento de la coherencia entre las reglas de la base de conocimiento: En este sentido son dos los principales problemas que pueden aparecer
   - Un encadenamiento infinito, que aparece cuando en la base de conocimiento encontramos reglas del tipo:

     Si A **Entonces** B

     ...

     Si B **Entonces** A

     En este caso, el motor de inferencia puede ciclar infinitamente. Este hecho es especialmente difícil de detectar cuando tenemos un elevado número de reglas y la regla que cierra ciclo aparecen después de varias etapas de razonamiento, como por ejemplo:

     Si A **Entonces** B ; **Si** B **Entonces** C ; ... ; **Si** K **Entonces** A

   - Problemas de ampliación de la base de conocimiento: En algunas situaciones es necesario realizar una actualización del conocimiento, por ejemplo añadiendo excepciones para un determinado conjunto de reglas o bien incorporando nuevo conocimiento. En estos casos, y con la finalidad de mantener la coherencia entre las reglas, puede ser necesario incluir un elevado número de reglas, provocando que la base de conocimiento se haga innecesariamente grande. En estas situaciones, puede ser preferible reconstruir la base de conocimiento, con el coste que ello implica.
2. Tienen dificultades para retractarse de anteriores conclusiones: Este problema viene provocado por el carácter modular y monótono de este tipo de sistemas. Así, cuando se cumple la premisa de una regla, nos da licencia para actuar sin tener en cuenta el resto del conocimiento. Por ejemplo, consideremos el siguiente conjunto de reglas. Partiendo de que tenemos como hecho A, la primera regla nos permite deducir B. Si posteriormente aparece como hecho C, entonces podremos inferir **no** B (aplicando la segunda y tercera regla), obteniendose una contradicción en el proceso de razonamiento (se deducen como hechos B y **no** B).

Si A **Entonces** B
Si C **Entonces** D
Si D **Entonces no** B

3. Opacidad: Las división de la base de conocimiento en pequeña reglas tiene como ventaja que cada una de ellas es fácil de utilizar individualmente, ganando el sistema en modularidad. Sin embargo, se tiene que pagar un precio por ello (que en muchos casos resulta elevado) consistente en una pérdida de una perspectiva global sobre el problema que estamos considerando.
4. Ineficiencia: Durante el proceso de inferencia, en cada iteración es necesario chequear cada regla para ver si es aplicable. Este proceso, aún cuando se han hecho avances para tratar de solucionarlo, es altamente costoso.

3 Sistemas Expertos que trabajan en entornos con incertidumbre

Hasta este momento, hemos venido considerando que para aplicar una regla es condición imprescindible que su premisa sea cierta. Además, como consecuencia de su aplicación tenemos que se añade un nuevo hecho (o conjunto de hechos) a la base de datos.

Sin embargo, cuando consideramos problemas reales, la situación no es tan idílica como la presentada. En la mayoría de los casos, el experto obtiene su conocimiento en base a su experiencia sobre el problema en cuestión, es decir, el conocimiento es de tipo heurístico. Por ejemplo, un experto puede tener como regla que *El fumar provoca cáncer de pulmón.* Sin embargo, del hecho "Juan fuma"no se puede concluir con certeza que "Juan tenga cáncer de pulmón". De igual forma, la incertidumbre puede venir asociada al conjunto de datos, por ejemplo, pueden faltar datos o bien estos no han podido determinarse de forma precisa por un error del aparato de medida.

Se puede decir, sin temor de faltar a la verdad, que la mayor parte del conocimiento humano consiste en sentencias y reglas, de las que no podemos garantizar su certeza. Usualmente, las evidencias que tenemos (los hechos de un sistema basado en reglas) no nos permiten deducir con seguridad las conclusiones ni su negación, sin embargo permiten dar mayor credibilidad a una determinada sentencia, aunque no se disponga de garantía absoluta sobre la corrección de la misma. Por tanto, si queremos diseñar un sistema experto capaz de obtener las mismas conclusiones que un experto humano, tenemos que dotarlo de la capacidad de razonar con este tipo de conocimiento incierto.

Cuando tratamos de incorporar el tratamiento de la incertidumbre dentro de un sistema experto hemos de tener en cuenta los siguientes factores:

1. ¿ Cómo se representa la incertidumbre sobre los datos?
2. ¿ Cómo combinar dos o mas elementos de información incierta?

3. ¿ Cómo realizar inferencias utilizando datos inciertos.?

El primer sistema experto que consideró conocimiento incierto fué MYCIN. Una regla **Si** A **Entonces** B se representaba de la forma $A \xrightarrow{m} B$ expresando la idea de que si conoces A, entonces se puede actualizar la certeza de B en cierta cantidad, función de la fuerza de la regla, m. El valor m es denominado factor de certeza de la regla y toma valores en el intervalo $[-1, 1]$ (1 para completamente cierto y -1 para completamente incierto).

La modularidad de los sitemas basados en reglas permite que el valor de verdad de un conjunto de reglas se defina como una función del valor de verdad de las subformulas que las componen. En este sentido podemos decir que tratan la incertidumbre como valores de verdad generalizados.

A modo de un ejemplo sencillo, consideremos el siguiente conjunto de reglas:

$$A \xrightarrow{m_1=0.8} C \ ; \ C \xrightarrow{m_2=0.7} D \ ; \ D \xrightarrow{m_3=0.9} E$$

y supongamos que observamos A (la certeza de A es 1). Entonces nuestra creencia sobre C se actualiza a un valor 0.8, Ctz(C)=0.8, de igual forma, encadenando el razonamiento, la certeza que tenemos sobre D se obtiene mediante el producto Ctz(C)$*m_2$, esto es Ctz(D)= $0.8 * 0.7 = 0.56$ y la certeza sobre E, Ctz(E)=$0.56 * 0.9 = 0.504$.

Podemos tener reglas con un antecedente formado por una sentencia compuesta, como por ejemplo

$$\text{A y B} \xrightarrow{m=0.8} \text{C}$$

donde la certeza de A es 0.9 y la certeza de B es 0.5. En estos casos, MYCIN calcula el valor de certeza para la sentencia (A **y** B) como una función de las certezas de A y B, por ejemplo el operador mínimo, esto es Ctz(A **y** B) = $\min\{0.9, 0.5\} = 0.5$ y por tanto asigna a C un valor de certeza 0.4 (0.8*0.5). En [18] podemos encontrar mecanismos de propagación que permiten trabajar con sistemas más complejos.

Un sistema basado en reglas que incorpore el tratamiento de la incertidumbre en su proceso de inferencia hereda las mismas ventajas y desventajas que los sistemas basados en reglas tradiciones. Sin embargo, el uso de este tipo de información incierta provoca la aparición de nuevos problemas:

1. Manejo incorrecto de inferencias bidireccionales. En este tipo de sistemas aparecen problemas cuando tratamos de utilizar un razonamiento en los dos sentidos. Así, consideremos la siguiente regla

Si Hay-Fuego **Entonces** Hay-Humo

Supongamos que tenemos el hecho "Hay-Humo". En este caso, ni el modus ponens (razonamiento hacia delante), ni el modus tollens (razonamiento hacia atras) se pueden aplicar. Sin embargo, parece sensato pensar en una segunda regla que expresase la idea de que

Si Hay-Humo **Entonces** Es-Mas-Creible Hay-Fuego

La incorporación de esta regla a la base de conocimiento puede hacer que nuestro sistema cicle indefinidamente. Para evitar este tipo de problemas, los sistemas basados en reglas no permiten el uso de los dos tipos de razonamiento simultanemente.

2. No tratan de manera adecuada las fuentes de información dependientes. Cuando se dispara una regla, el peso que se asigna a la conclusión dependene únicamente del peso de las premisas, pero no se tiene en cuenta de donde vienen esas premisas. Los resultados obtenidos son los mismos, independientemente de que si la información proviene de una única fuente que ha seguido diferentes caminos, o por el contrario proviene de fuentes independientes.

Desde su aparición, son muchos los sistemas que han utilizado los factores de certeza con buenos resultados en su área de aplicación. Sin embargo, en gran parte, su éxito se debe a una correcta descripción del conocimiento en forma de reglas y no a la asignación de valores concretos a los factores de certeza. Además, el uso de factores de certeza ha recibido multiples críticas por su incapacidad de representar ciertas dependencias entre las observaciones y la forma en la que combina el conocimiento. Este hecho provoca la necesidad de encontrar otros formalismos más adecuados para trabajar con incertidumbre.

Entre los más antiguos podemos encontrar la Teoría de la Probabilidad [16]. En 1654 Pascal y Fermat, partiendo de una noción intuitiva de la idea de 'azar' o 'aleatoriedad', presentan una primera aproximación al concepto de probabilidad. El transcurso de los años ha dotado a este formalismo de unas sólidas bases matemáticas, convirtiendolo en uno de los mecanismos más utilizados para el tratamiento de la incertidumbre.

La Teoría de la Probabilidad permite codificar la información sobre el problema desde otra perspectiva. En lugar de asignar valores de verdad de forma independiente a cada una de las fórmulas, el conocimiento inicial es considerado desde un punto de global, ofreciendo una semántica clara. Este hecho, unido a su solidez teórica, se puede considerar como la causa de que en los primeros sistemas expertos se intentase utilizar la teoría de la probabilidad como herramienta para tratar la incertidumbre. El principal problema que se planteaba era el alto coste computacional necesario, llegando incluso a considerarlo como una tarea intratable (Gorry 1973 [7]).

Por tanto, este tipo de sistemas necesitan de mecanismos especiales que permitan realizar el razonamiento de forma eficiente. Con este fin se recurre al uso de relaciones de dependencia/independencia entra variables. La idea es tener una codificación del conocimiento de tal manera que lo que realmente es relevante pueda ser reconocido fácilmente, y en este sentido, aquello que no es conocido localmente es ignorado.

Un tipo de sistema experto que tiene en cuenta estas consideraciones lo constituyen las Redes Probabilísticas, permitiendo obtener (utilizando cálculos locales) los mismos resultados que si se hubiese trabajado con la información global. Son muchos los sistemas expertos que utilizan estas estructuras como base de su razonamiento, como por ejemplo MUNIN [5], PATHFINDER [9] en medicina, BOBLO [14] em agricultura, VISTA [10] en aeronaútica, etc.

Para finalizar la sección, destacar que existen otros muchos formalismos alternativos para el tratamiento de la incertidumbre, como por ejemplo la Teoría de la Posibilidad [4], medidas de evidencia [3,17], o los conjuntos difusos [20]. Dichos formalismos también han sido utilizados, con mayor o menor éxito, en el desarrollo de sistemas expertos [8].

4 Teoría de la Probabilidad

Nuestro interés se centra en el estudio de sistema expertos probabilísticos. Por tanto, dedicaremos esta sección a realizar un breve repaso sobre la Teoría de la Probabilidad.

Podemos encontrar distintas aproximaciones al concepto de PROBABILIDAD. Entre ellas podemos destacar la aproximación objetiva, que considera la probalidad como la frecuencia relativa de un experimento (razón entre el número de veces que se obtiene una determinada salida y el número total de veces que se realiza el experimento). Por ejemplo, si lanzamos 100 veces un dado y 19 de ellas obtenemos el valor 5, entonces la probabilidad de dicha salida se obtiene mediante la razón $\frac{19}{100}$, esto es, $P(Dado = 5) = 0.19$.

Supongamos que queremos realizar un experimento, E. El conjunto de posibles salidas para este experimento se denomina espacio muestral, U. Un subconjunto del espacio muestral es lo que se denomina suceso A. Los sucesos que incluyen un único elemento se denominan sucesos simples o átomos. La probabilidad de que ocurra un suceso A se denotará por $P(A)$.

Ejemplo 2. Nuestro experimento, E, consiste en seleccionar de forma aleatoria a una población de 100 personas. De estas personas estamos interesados en estudiar la variable color de pelo CP, que tomará valores en el conjunto {Rubio (R), Moreno (M), Castaño (C), Pelirrojo (P)}. En este caso, U viene representado por

$U = \{R \cup M \cup C \cup P\}$. Un suceso puede ser un único átomo, $CP = \{R\}$ o bien estar formado por un conjunto de estos $CP = \{R \cup C\}$.

Imaginemos que los individuos seleccionados se distribuyen como indica la siguiente tabla:

Castaño = 45; Rubio = 15
Pelirojo = 5; Moreno = 35

□

La aproximación frecuentista me permite determinar la probabilidad de un suceso A, $P(A)$ como

$$P(\mathrm{A}) = \frac{\text{Número de individuos que hacen cierto el suceso } A}{\text{Número total de individuos}}$$

También es posible utilizar una aproximación subjetivista al concepto de probabilidad, considerandose como la *creencia* que un individuo determinado tiene sobre la salida de un experimento.

Axiomas de Kolmogorov A.N. Kolmogorov establece el siguiente conjunto de axiomas:

1. La probabilidad de un suceso es no negativa, $P(A) \geq 0$ ($P(A) = 0$ expresa que el suceso no ocurre y $P(A) = 1$ indica que el suceso es seguro).
2. La probabilidad del espacio muestral X es 1, $P(X) = 1$, indicando que con seguridad la salida del experimento se encuentra en X.
3. Cuando tenemos un conjunto de sucesos mutuamente excluyentes (con intersección vacia) $A_1, A_2, \ldots, A_n$ entonces la probabilidad de que al menos uno de estos sucesos ocurran es la suma de las probabilidades individuales, esto es

$$P(A_1 \cup A_2 \cup \ldots A_n) = P(A_1) + P(A_2) + \ldots P(A_n)$$

Una probabilidad nos permite asignar nuestras creencias sobre el "conjunto de de mundos posibles" que forman el *espacio muestral.* Por ejemplo, como A y su complementario, $\overline{A}$ son sucesos disjuntos y considerando que $X = A \cup \overline{A}$, podemos deducir que

$$P(A) + P(\overline{A}) = 1$$

De forma análoga obtenemos que

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

Ejemplo 3. (Continuación) Si consideramos los átomos del experimento tenemos que

$$P(CP = C) = 0.45 \; P(CP = R) = 0.15$$
$$P(CP = P) = 0.05 \; P(CP = N) = 0.35$$

y la probabilidad de que una persona escogida al azar (dentro de la población) tenga el color de pelo rubio o castaño se obtiene como

$$P(CP = \{R \cup C\}) = P(CP = R) + P(CP = C) = 0.15 + 0.45 = 0.60$$

□

Probabilidad Marginal Sean $X_1, X_2, \ldots, X_n$ un conjunto de variables aleatorias que toman valores discretos y sea $\{x_1, x_2, \ldots, x_n\}$ el conjunto de sus posibles realizaciones. Sea $P(x_1, x_2, \ldots, x_n)$ una probabilidad sobre $X_1, X_2, \ldots, X_n$ (distribución de probabilidad conjunta), esto es

$$P(x_1, x_2, \ldots, x_n) = P(X_1 = x_1, X_2 = x_2, \ldots, X_n = X_n)$$

Entonces la distribución de probabilidad marginal sobre la i-ésima variable se obtiene mediante

$$P(x_i) = P(X_i = xi) = \sum_{x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n} P(x_1, \ldots, x_n)$$

Ejemplo 4. Si X_1 representa si una persona es fumador o no $X_1 = \{\text{si,no}\}$, y X_2 representa si una persona tiene cáncer de pulmón o no, $X_2 = \{\text{si,no}\}$, entonces las posibles realizaciones serán pares de la forma {si,si}, {si,no}, etc. Supongamos que la probabilidad se distribuye entre los átomos como indica la figura

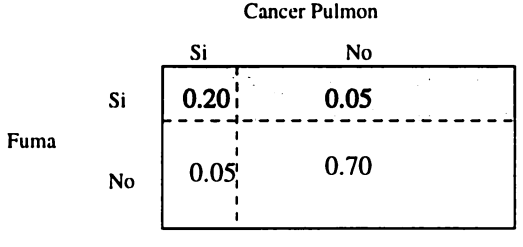


Figura 3. Experimento

La probabilidad de que una persona sea fumadora, $P(X_1 = si)$, se obtendría como la marginal sobre la variable X_1, es decir, $P(X_1 = si) = P(X_1 = si, X_2 = si) + P(X_1 = si, X_2 = no) = 0.20 + 0.05 = 0.25$ □

Probabilidad Condicionada Sean X e Y dos conjuntos disjuntos de variables que toman valores en $\{x_1, \ldots, x_n\}$ y $\{y_1, \ldots, y_m\}$. La distribución de probabilidad condicionada de X dado que $Y = y_j$ (con $j \in \{1, \ldots, m\}$ y $P(Y = y_j) > 0$) viene dada por

$$\forall x_i \in X; P(X = x_i | Y = y_j) = P(x_i | y_j) = \frac{P(x_i, y_j)}{P(y_j)}$$

Por tanto, la distribución de probabilidad conjunta puede obtenerse como

$$P(x_i, y_j) = P(y_j) P(x_i | y_j)$$

La distribución de probabilidad conjunta nos va a permitir actualizar nuestro conocimiento a la luz de nueva información.

Ejemplo 5. Continuando con el ejemplo anterior, supongamos que tenemos información adicional y sabemos que una determinada persona de la población es fumadora. Entonces la probabilidad de que esa persona padezca cancer de pulmón se obtiene como

$$P(X_2 = si | X_1 = si) = \frac{P(X_1 = si, X_2 = si)}{P(X_1 = si)} = \frac{0.20}{0.25} = 0.80$$

□

Independencia Probabilística Las siguientes definiciones nos permiten establecer la independencia entre sucesos o variables.

Definición 1. Sean X e Y dos subconjuntos disjuntos del conjunto de variables aleatorias $\{X_1, \ldots, X_n\}$. Entonces se dice que X es MARGINALMENTE INDEPENDIENTE de Y, y lo notamos por $I(X, \emptyset, Y)$, si y solamente si para todos los posibles valores x de X e y de Y se satisface que

$$P(x|y) = P(x)$$

En caso contrario, X se dice que es (marginalmente) dependiente de Y, y se denota por $\neg I(X, \emptyset, Y)$. □

Ejemplo 6. Si consideramos el ejemplo anterior podemos ver que ser fumador y tener cáncer de pulmón son dos variables dependientes:

$$P(X_1 = si) = 0.25 \neq P(X_1 = si | X_2 = no) = (0.05/0.75) = 0.066.$$

□

Definición 2. Sean X, Y y Z tres conjuntos disjuntos de variables, entonces se dice que X es CONDICIONALEMENTE INDEPENDIENTE de Y dado que conocemos Z, y lo notamos por $I(X, Z, Y)$, si y solo si para todos los valores x, y, z de X, Y, Z (respectivamente) se satisface que

$$P(x|z, y) = P(x|z)$$

En caso contrario, se dice que son condicionalmente dependientes, y lo notamos por $\neg I(X, Z, Y)$. □

Ejemplo 7. Supongamos que X, Y y Z son tres variables que toman valores en el conjunto $\{0, 1\}$ y supongamos que la distribución de probabilidad conjunta viene representada en la siguiente tabla.

X	Y	Z	P	X	Y	Z	P
0	0	0	0.015	1	0	0	0.21
0	0	1	0.135	1	0	1	0.14
0	1	0	0.03	1	1	0	0.245
0	1	1	0.12	1	1	1	0.105

En este ejemplo, podemos ver como las variables X e Y son marginalmente independientes $I(X, \emptyset, Y)$, esto es

$$P(X = 0) = 0.3 = P(X = 0|Y = 0) = P(X = 0|Y = 1)$$

$$P(X = 1) = 0.7 = P(X = 1|Y = 0) = P(X = 1|Y = 1)$$

Sin embargo, conocido el valor de Z, X e Y son condicionalmente dependientes, $\neg I(X, Z, Y)$ ya que, por ejemplo, $P(X = 0|Y = 1, Z = 1) \neq P(X = 0|Z = 1)$.

$$P(X = 0|Y = 1, Z = 1) = \frac{P(X = 0, Y = 1, Z = 1)}{P(Y = 1, Z = 1)} = \frac{0.12}{0.225} = 0.533$$

$$P(X = 0|Z = 1) = \frac{P(X = 0, Z = 1)}{P(Z = 1)} == \frac{0.255}{0.5} = 0.51$$

□

Teorema de Bayes Este teorema nos permite representar la probabilidad condicionada $P(y|x)$ mediante la siguiente expresión:

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

Si tenemos en cuenta que $P(x) = \sum_{y \in Y} P(x, y)$ y que $P(x, y) = P(x|y)P(y)$ el teorema de Bayes lo podemos representar mediante la siguiente expresión

$$P(y|x) = \frac{P(x|y)P(y)}{\sum_{y \in Y} P(x|y)P(y)}$$

5 Sistemas Basados en Reglas Probabilísticos

Veamos como podemos utilizar la teoría de la probabilidad como herramienta para el tratamiento de la incertidumbre en un sistema basado en reglas. En este caso, las reglas serán de la forma:

> **SI** X es cierto
> **Entonces** puede deducirse Y con probabiliada p

donde p se puede interpretar como la probabilidad condicionada de Y, dado que conocemos X, $P(Y|X \text{ } es \text{ } cierto)$.

Son muchos los dominios en los que el experto tiene codificado el conocimiento en base a relaciones del tipo *causa - efecto*. Así, en problemas médicos las reglas suelen ser del tipo:

> **Si** el paciente tiene la enfermedad
> **entonces** presentará un síntoma con una probabilidad p

Por ejemplo,

> **Si** un paciente está resfriado
> **entonces** estornudará con una probabilidad de 0.75,

o desde el punto de vista probabilístico, $P(Y = est|X = res) = 0.75$

En estos casos, cuando consideramos problemas de diagnóstico, los datos o hechos que se conocen están formados por un conjunto de síntomas, como por ejemplo que el paciente estornuda. La pregunta que nos planteamos es ¿ Es posible modificar nuestra creencia sobre el hecho 'el paciente está resfriado'?. A este tipo de razonamiento se le conoce como razonamiento abductivo, pretendiendo buscar el conjunto de causas (hipótesis) que mejor explican los síntomas (evidencias).

El uso del formalismo probabilístico nos permite realizar este tipo de razonamiento. En concreto, es suficiente con el uso de la regla de Bayes

Ejemplo 8. Supongamos que conocemos los siguientes datos:

- La probabilidad de que Pedro esté resfriado es de 0.2, $P(X = res) = 0.2$
- La probabilidad de que Pedro esturnude cuando está resfriado es 0.75, $P(Y = est|X = res) = 0.75$
- La probabilidad de que Pedro esturnude cuando no está resfriado es 0.1, $P(Y = est|X = \overline{res}) = 0.1$

Entonces, si tenemos en cuenta que $1 = P(X = res) + P(X = \overline{res})$, podemos calcular la probabilidad de que Pedro estornude como

$$P(Y = est) = [P(Y = est|X = res)P(X = res)] + [P(Y = est|X = \overline{res})P(X = \overline{res})]$$

$$(0.75)(0.2) + (0.1)(0.8) = 0.15 + 0.08 = 0.23$$

y, utilizando la regla de Bayes obtenemos que, si sabemos que Pedro ha estornudado, la probabilidad de que esté resfriado es

$$P(X = res|P(Y = est) = \frac{P(Y = res|X = res)P(X = res)}{P(Y = est)} = \frac{(0.75)(0.2)}{0.23} = 0.65$$

Por tanto, podemos decir que el que Pedro estornude incrementa 3 veces la probabilidad de estar resfriado. □

Cuando utilizamos un sistemas basado en reglas estamos asumiendo de forma implícita que cada regla es independiente de las demas reglas. Esta suposición es poco realista y como resultado de ella podemos obtener resultados extraños a la hora de realizar las tareas de razonamiento. Por ejemplo, dos síntomas, considerados de forma independiente, pueden indicar que cierta enfermedad es probable en un grado 0.8. Sin emgargo, puede ocurrir que cuando consideramos los síntomas de forma conjunta pueden eliminar la creencia de que el paciente sufra dicha enfermedad (los síntomas se anulan entre sí). Este mal funcionamiento proviene de un uso incorrecto de las hipótesis de independencia entre las variables que componen la regla. En la siguiente sección veremos como los sistemas expertos probabilísticos permiten resolver el problema.

6 Sistemas Expertos Probabilísticos

Son dos los elementos esenciales que caracterizan a un sistema experto: La base de conocimiento y el motor de inferencia. Desde un punto de vista general, en un sistema experto probabilístico la base de conocimiento está formada por un conjunto de variables $X_1, \ldots, X_n$ y una distribución de probabilidad conjunta sobre ellas $P(x_1, \ldots, x_n)$. Por otro lado, un motor de inferencia básico será aquel que nos permita actualizar nuestra información sobre una determinada variable (o conjunto de ellas), X, ante la presencia de un conjunto de hechos, evidencias o síntomas determinados, E . En teoría de la probabilidad este motor de inferencia nos es mas que el cálculo de la probabilidad condicional $P(X|E)$.

Si tratamos de realizar una aproximación directa, en la que representamos la distribución de probabilidad conjunta con una tabla, pronto nos damos cuenta de que, incluso en problemas con un conjunto pequeño de variables, el problema es intratable. Supongamos que para representar un valor numérico, esto es, un

valor de probabilidad concreto, necesitamos 4 bytes y supongamos que tenemos 10 variables bivaluadas. En este caso, necesitaremos de una tabla con 2^{10} entradas y, por tanto, de 4 KiloBytes para almacenar la distribución de probabilidad conjunta. Este tamaño puede parecer razonable, pero si multiplicamos por dos el número de variables, el tamaño necesrio para almacenar la tabla (2^{20} entradas) pasa a ser de 4 MegaBytes, y si volvemos a duplicar (40 variables) necesitamos de 4095 GigaBytes. Este comportamiento es debido a que el tamaño de la tabla crece exponencialmente con el número de variables (para n variables bivaluadas necesitamos 2^n entradas en la tabla). Como consecuencia de esto, el proceso de inferencia con este tipo de estructuras es altamente costoso.

Sin embargo, son muchas las aplicaciones prácticas en las que conocemos "a priori" que, por ejemplo, dos variables son (marginal o condicionalmente) independientes. En estos casos, podemos utilizar dicha información con el objetivo de reducir el espacio necesario para almacenar la distribución de probabilidad conjunta. La idea es dividir o factorizar dicha distribución en un conjunto de distribuciones más pequeñas (que involucran a menos variables), pero con la misma representatividad. En cualquier caso, es necesario proporcionar un método que permita recuperar los valores originales de la distribución de probabilidad conjunta.

Por ejemplo, supongamos que tenemos 2 variables, X e Y, donde cada una de ellas puede tomar 10 valores, $\{x_1, \ldots, x_{10}\}$ e $\{y_1, \ldots, y_{10}\}$. Para almacenar la distribución conjunta, el número de entradas necesarias en una tabla (cada entrada es de la forma (x_i, y_j) con $i, j \in \{1, \ldots, 10\}$) es de $10 \times 10 = 100$. Si conocemos que X e Y son variables independientes, esto es $P(x|y) = P(x)\ \forall x \in X, y \in Y$, tenemos que $P(x, y)$ se puede expresar mediante $P(x, y) = P(x)P(y)$. En este caso, es posible almacenar únicamente las distribuciones de probabilidad (marginal) para X e Y, $P(x)$ y $P(y)$, de forma independiente (necesitaremos 2 tablas con 10 entradas cada una) y recuperar la distribución conjunta realizando una operación de multiplicación.

Semánticamente, el que dos variables, X e Y, sean independientes expresa la idea de que " *el conocer que la variable Y toma un determinado valor $(Y = y_j)$ no aporta ninguna información sobre nuestra creencia en el valor que puede tomar la variable X (y viceversa)* ". Si tenemos en cuenta dicha información a la hora de realizar tareas de razonamiento, podemos evitar el realizar cálculos que desde el principio sabemos que son innecesarios.

La idea básica es codificar el conocimiento de tal manera que no sea necesario el utilizar información que sea irrelevante y, por otro lado, la información relevante sea fácilmente accesible. Podemos encontrar distintos modelos para implementar esta idea. Entre ellos, queremos destacar las redes de Markov [2,13] y las redes Bayesianas [2,11,15,13]. Ambos sistemas se apoyan en modelos gráficos para re-

presentar de forma explícita las relaciones de dependencia e independencia entre las variables.

Por ejemplo, las redes de Markov se representan gráficamente mediante grafos no dirigidos, donde los nodos representan las variables y una relación de dependencia entre dos variables se representa mediante la existencia de un camino o conexión entre ellas. Por otra parte, en estos modelos también se representan las relaciones de independencia. En concreto, si X, Y y Z son conjuntos disjuntos de variables, entonces

1. Una independencia marginal $I(X, \emptyset, Y)$ viene representada por la inexistencia de conexión entre las variables de X e Y.
2. Una relación de independencia condicionada del tipo $I(X, Z, Y)$ se representa por el hecho de que todo camino que conecta las variables de X con variables de Y contiene algún nodo de Z. Por tanto, si los nodos en Z son borrados del grafo las variables X e Y quedan desconectadas.

En conclusión, hemos dotado a la estructura gráfica (el grafo no dirigido) de una semántica clara de dependencia / independencia. Esto es, dado un grafo y analizando caminos en el mismo, somos capaces de determinar cuando dos variables son dependientes o no. Notaremos por $I(.,.,.)_G$ al conjunto de relaciones de independencia que se pueden obtener del grafo.

Ejemplo 9. Supongamos que tenemos cuatro variables bivaluadas:

- Llueve: (1 - Llueve en este momento; 0 - No llueve)
- Suelo Mojado: (1 - Suelo está mojado; 0 - Suelo seco)
- Accidente: (1 - Se produce un accidente; 0 - No hay accidente)
- Novela: (1- Hemos leido una novela; 0 - No la hemos leido)

La figura representa las relaciones de relevancia entre ellas.

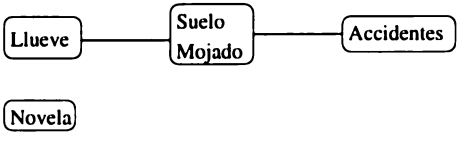


Figura 4. Red de Markov

Considerando el anterior criterio de independencia gráfico podemos decir que:

- $I(Novela, \emptyset, Accidente)_G$: El hecho de haber leido una novela no modifica mi creencia sobre el que se produzca un determinado accidente o no

$\neg I(Llueve, \emptyset, Mojado)_G$: Existe una relación directa entre el hecho de llover y que encontremos mojado el suelo.
$\neg I(Accidente, \emptyset, Llueve)_G$: Podemos encontrar una relación entre el número de accidentes y el hecho de que ha llovido o no.
$I(Accidente, Mojado, Llueve)_G$: Si sabemos que el suelo está seco, entonces el conocer que ha habido muchos accidentes no cambia mi creencia sobre el hecho de que no ha llovido.
.....

□

Son muchas las cuestiones (un estudio detallado de las mismas lo podemos encontrar en [2],jPearl88) que nos podemos plantear sobre este tipo de estructuras, como por ejemplo:

- ¿ Cómo se almacena la distribución de probabilidad conjunta?
- ¿ Qué mecanismo de inferencia podemos encontrar?
- ¿ Puede el modelo gráfico representar todas las relaciones de dependencia / independencia que se derivan de una distribución de probabilidad conjunta?
- etc.

Nos centraremos en el análisis de la última de ellas. En este caso podemos encontrar distribuciones de probabilidad, como la expresada en la la siguiente tabla, para las que no existe un grafo no dirigido que sea capaz de representar las relaciones de independencia que se derivan de la distribución.

X	Y	Z	P	X	Y	Z	P
0	0	0	0.015	1	0	0	0.21
0	0	1	0.135	1	0	1	0.14
0	1	0	0.03	1	1	0	0.245
0	1	1	0.12	1	1	1	0.105

En este ejemplo, podemos encontrar las siguiente relaciones de independencia:

- $I(Z, \emptyset, Y) \Longrightarrow$ no existe un camino que conecte X con Y.
- $\neg I(X, \emptyset, Z) \Longrightarrow$ existe un camino que conecta X con Z.
- $\neg I(Z, \emptyset, Y) \Longrightarrow$ existe un camino que conecta Z con Y.

De donde podemos deducir que existe un camino que conecta X con Y, el que pasa por Z. Desde un punto de vista más formal podemos decir que las redes de Markov no son capaces de representar relaciones de independencia no transitivas.

En la siguiente sección analizaremos con detalle las redes Bayesianas: Una herramienta para diseñar sistemas expertos probabilísticos utilizando el formalismo más potente de los grafos dirigidos para representar las relaciones entre las variables.

7 Redes Bayesianas

Las redes Bayesianas constituyen una de las herramientas más poderosas en el diseño de sistemas expertos probabilísticos. Desde un punto de vista gráfico una red Bayesiana es un Grafo Dirigido Acíclico, donde los nodos representan las variables del problema que queremos resolver. Estas estructuras nos permiten representar el conocimiento desde dos puntos de vista:

- Cualitativo: Expresa las relaciones de dependencia e independencia entre las variables. De forma gráfica se representa mediante la presencia de conexiones o caminos entre variables. Así, si tenemos dos variables X e Y conectadas por un arco $X \longrightarrow Y$ podemos deducir que X es Y son variables que están relacionadas (por ejemplo, X puede ser una causa de Y). Cuando dicho arco no existe, entonces podemos decir que existe una relación de independencia (bien marginal o bien condicional) entre X e Y.
- Cuantitativo: Expresa la fuerza con la que nos creemos las relaciones de relevancia o dependencia. Nos permite representar la incertidumbre que tenemos sobre la ocurrencia de los sucesos (supuesto que conocemos un conjunto determinado de hechos). Este tipo de conocimiento se proporcionará mediante un conjunto de distribuciones de probabilidad condicionadas.

Pasamos a ver de una forma más detallada la red Bayesiana como un formalismo que permite representar la base de conocimiento de un sistema experto probabilístico.

7.1 Grafos Dirigidos como modelo para representar Independencias

Al igual que ocurre en las redes de Markov, la topología de de la red nos permite representar la componente cualitativa del conocimiento en base a un conjunto de relaciones de dependencia e independencia entre variables.

El siguiente ejemplo muestra una posible interpretación semántica de las relaciones de dependencia e independencia representadas en una red Bayesiana.

Ejemplo 10. Supongamos que vamos a alquilar un vehículo para realizar un viaje por carretera. Una posible representación del problema la tenemos en la siguiente figura, donde el conjunto de variables consideradas relevantes son:

TV: Tipo de Vehículo con el cual vamos a realizar un viaje, que puede tomar los valores {*Utilitario,Deportivo,Berlina*}.

TC: Tipo de Carretera por la cual transcurre el viaje, tomando valores {*Autopista, Nacional, Comarcal, Urbana* }.

VM: Velocidad Media en el viaje. Supongamos que discretizamos los posibles valores en los intervalos (en Km/h.) $\{[0,50],(50,80],[80,120),[120,\ldots]\}$.
D: Duración (en horas) del viaje, tomando valores en $\{[0,1),[1,2),[2,3),[3,\ldots]\}$.
P: Precio de alquiler, tomando valores en $\{[0,10000),[10000,30000),[30000,\ldots]\}$.
K: Kms. por recorrer, tomando valores en $\{[0,10),[10,50),[50,100),[100,\ldots]\}$.

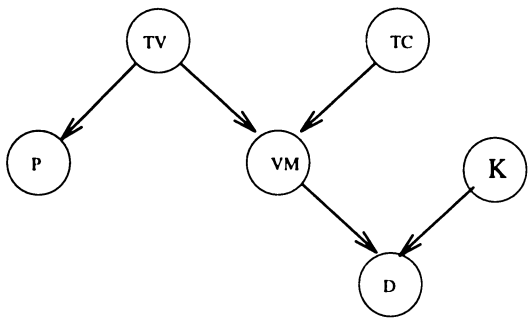


Figura 5. Viaje por Carretera.

La presencia de un arco se interpreta como la existencia de una relación de relevancia o dependencia directa, por ejemplo $TV \rightarrow P$ nos expresa la idea de que el precio de alquiler de un determinado modelo está relacionado con el tipo de vehículo.

Sin embargo, en la estructura también se encuentran representadas otro tipo de relaciones de una forma no tan directa.

Analicemos el subgrafo $TC \rightarrow VM \rightarrow D$: En este caso, las relaciones de dependencia que tenemos son: El tipo de vía influye sobre la velocidad media del viaje y ésta influye directamente sobre la duración del mismo. Además, cuando no se sabe nada sobre la velocidad media en el trayecto, la duración del viaje influye en nuestra creencia sobre el tipo de carretera y viceversa. Sin embargo, si sabemos que la velocidad media del viaje pertenece al intervalo $[120,\ldots]$, entonces el saber que la duración del viaje es de 4 horas, no altera mi creencia en que la vía debe ser una autopista o autovía. En términos de relaciones de independencia, podemos decir que TC y D son variables dependientes, sin embargo conocida la velocidad media del viaje, TC y D son independientes.

En el subgrafo $P \leftarrow TV \rightarrow VM$, podemos hacer un razonamiento análogo: Si el precio de alquiler es bajo, entonces podemos imaginar que el vehículo es un utilitario y por tanto la velocidad media no debe ser muy elevada. Sin embargo, si conocemos que el vehículo es un deportivo, el conocer el precio de alquiler no

aporta información sobre la velocidad media en el viaje. En este caso, tenemos que P y VM son variables dependientes, pero conocido el valor de TV, se hacen independientes.

Para finalizar, analicemos el subgrafo $TV \rightarrow VM \leftarrow TC$. Aquí observamos como el tipo de vehículo es independiente del tipo de carretera por la que se va a realizar el viaje, es decir, saber que el viaje se realiza en un utilitario, no dice nada sobre el tipo de vía por la que se va a circular. En cambio, si se sabe que se realizó el viaje en un utilitario y que la velocidad media fue de 140Km/h, mi creencia en que el viaje se hizo por autopista aumenta. Por tanto, las variables TV y TC son independientes, pero conocido VM se hacen condicionalmente dependientes. □

El concepto de independencia, además de facilitar una representación cualitativa del problema, nos permite identificar qué información es relevante y qué información es superflua. Por tanto, a la hora de encontrar posibles explicaciones para una determinada consulta, podemos modularizar el conocimiento de forma que sólo sea necesario consultar la información relevante. Consideremos el anterior ejemplo: Supongamos que nuestro interés se centra en conocer la duración D de un desplazamiento, y supongamos que nos proporcionan como dato de entrada la velocidad media del mismo, $VM = 50$, y los kilómetros del desplazamiento $K = 70$. En este caso, conocer cualquier otra información sobre el resto de variables representadas en la red no aportará ninguna información adicional sobre D.

Una vez presentados estos ejemplos, podemos entender que para dotar de una interpretación semántica completa a un grafo dirigido se necesita de un criterio que determine, de forma precisa, qué propiedades de independencia son reflejadas por la topología de la red. Sin embargo, para grafos dirigidos el criterio de independencia gráfica, que denominaremos *d-separación* o *separación dirigida*, es algo más complejo que el simple criterio de separación gráfica en grafos no dirigidos. Antes de considerar el criterio, detallaremos algunas definiciones previas.

Definición 3. El esqueleto de un GDA G es el grafo no dirigido que se forma al eliminar de G las direcciones en los arcos. Un camino es una secuencia de nodos conectados por arcos en el grafo. Un camino no dirigido, es un camino en el que no se consideran las direcciones de los arcos. Un enlace cabeza a cabeza en un nodo es un camino que tiene la forma $X \rightarrow Y \leftarrow W$, el nodo Y es un nodo cabeza a cabeza en el camino. Un camino c se dice activo por un conjunto de nodos Z si se satisface que

1. Todo nodo de c con arcos cabeza a cabeza está en Z o tiene un descendiente dentro de Z.
2. Cualquier otro nodo en el camino no pertenece a Z.

Si no se satisface esta relación se dice que el camino está bloqueado por Z. □

Vistas estas definiciones el criterio gráfico de independencia en un grafo dirigido, [15,13,19], puede expresarse como

Definición 4. d-separación. Si X, Y y Z son tres subconjuntos de nodos disjuntos en un GDA G, entonces Z se dice que **d-separa** X de Y, o lo que es lo mismo X e Y son gráficamente independientes dado Z y lo notamos como $< X \mid Z \mid Y >_G$, si todos los caminos entre cualquier nodo de X y cualquier nodo de Y estan bloqueados por Z. □

El siguiente ejemplo nos permite clarificar los conceptos presentados.

Ejemplo 11. Consideremos el siguiente grafo dirigido acíclico, en el que se representan las relaciones de relevancia entre las variables $A, B, \ldots, J$.

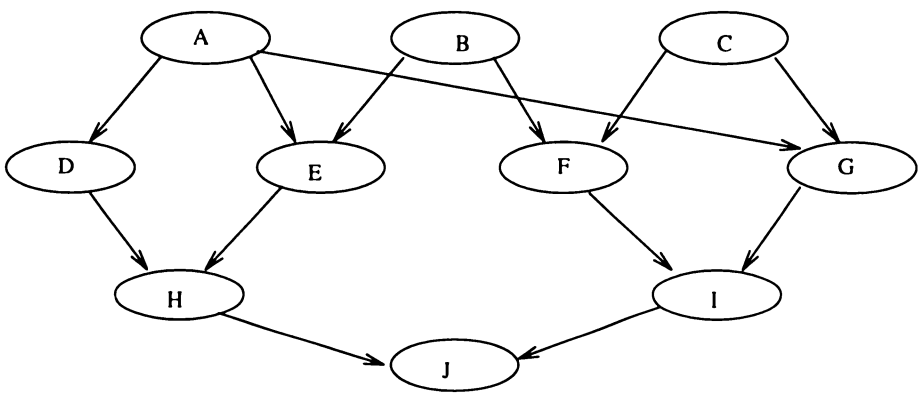


Figura 6. Criterio de d-separación

Utilizando el criterio de d-separación podemos ver como en la figura se satisfacen, entre otras muchas, las siguientes relaciones:

Relación	Comentarios
$< A\|\emptyset\|B >_G$	En todos los caminos entre A y B podemos encontrar un nodo cabeza a cabeza.
$\neg < J\|\emptyset\|C >_G$	Encontramos el camino $C \rightarrow F \rightarrow I \rightarrow J$ que no está bloqueado
$\neg < A\|E\|B >_G$	Si conocemos el valor que toma el nodo E, el camino $A \rightarrow E \leftarrow B$ se activa.
$\neg < A\|J\|B >_G$	El conocer J hace que conozcamos algo sobre H y al conocer H conocemos algo sobre E. Por tanto, la modificación de nuestra creencia en E hace que el camino $A \rightarrow E \leftarrow B$ este abierto.
$< A\|D,E\|H >_G$	En todo camino entre A y H o bien encontramos que está bloqueado por $\{D,E\}$ (como por ejemplo $A \rightarrow D \rightarrow H$) o bien podemos encontrar un nodo cabeza a cabeza que no pertenece a $\{D,E\}$ (por ejemplo, si consideramos el camino $A \rightarrow E \leftarrow B \rightarrow F \rightarrow I \rightarrow \mathbf{J} \leftarrow H$, podemos ver que aunque el nodo E lo active el nodo J lo bloquea finalmente)

□

Dado un grafo, es posible establecer las relaciones:

- X es padre de Y si el arco $X \rightarrow Y$ pertenece al grafo, de forma análoga se dice que Y es hijo de X.
- X es antecesor de Y si podemos encontrar un camino dirigido que partiendo de X alcance el nodo Y, es decir $X \rightarrow \ldots \rightarrow Y$. En este caso también diremos que Y es un descendiente de X.

Dos propiedades importantes que se pueden obtener utilizando el criterio de d-separación son las siguientes:

Proposición 1. *Toda variable X_i es condicionalmente independiente de todos sus no-descendientes, dado que conocemos el conjunto de padres* □

Proposición 2. *Si conocemos los padres, los hijos y los padres de los hijos, entonces una variable X_i queda separada (es independiente) del resto de las variables del grafo* □

7.2 Expresando la incertidumbre sobre el problema

Hasta este momento hemos visto que en un grafo dirigido podemos representar las relaciones de relevancia/irrelevancia entre las variables de un problema. En esta sección abordaremos cómo podemos tratar de forma numérica la incertidumbre que tenemos sobre la fuerza de estas relaciones.

Supongamos que tenemos dos variables X e Y y una relación de entre ellas del modo $X \longrightarrow Y$. Es este caso, estamos expresando que existe una dependencia directa entre las dos variables, por ejemplo, que "X es causa de Y". La incertidumbre asociada a este tipo de relaciones la podemos representar mediante el uso de una distribución de probilidad condicionada sobre Y, dado que conocemos el valor de X, $P(Y|X)$. Así, podemos decir que la creencia que tenemos de que Y tome el valor y, $(Y = y)$, dado que conocemos que X toma un valor $X = x$ es de 0.75, esto es, $P(Y = y|X = x) = 0.75$.

Es importante notar que con una distribución de probabilidad también es posible asignar valores de certeza total a una relación entre variables. Por ejemplo, supongamos la regla: **Si** $X = x$ **entonces** $Y = \overline{y}$. Esta información la podemos representar considerando el arco $X \rightarrow Y$, y asignándole la distribución de probabilidad condicional $P(Y = y|X = x) = 0$ y $P(Y = \overline{y}|X = x) = 1$.

De forma genérica, para cada variable X_i representada en el grafo, necesitamos almacenar un conjunto de distribuciones de probabilidad condicionadas a los valores que tomen el conjunto de sus padres en la red.

Ejemplo 12. Sean X, Y, Z, W, T y R variables bivaluadas donde X toma los valores $\{x, \overline{x}\}$, Y toma los valores $\{y, \overline{y}\}$, etc. Sea G el grafo dirigido acíclico que representa las relaciones de relevancia entre las variables. En este caso, como X e Y son

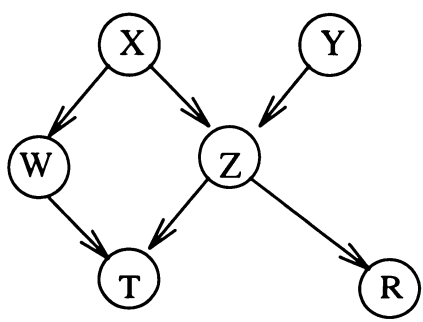


Figura 7. Criterio de d-separación

variables que no tienen padres es suficiente con almacenar para cada nodo su distribución de probabilidad marginal, esto es,

Para X $P(X = x) = 0.7$ y $P(X = \overline{x}) = 0.3$
Para Y $P(Y = y) = 0.5$ y $P(Y = \overline{y}) = 0.5$

En el nodo W se almacenan un conjunto de distribuciones condicionadas, una para cada uno de los posibles valores que toma X, el padre de W. Un razonamiento análogo se puede realizar para el nodo R.

$$\begin{array}{rrl} \text{Para } W & \text{Supongamos } X = x & P(W = w|X = x) = 0.4 \\ & & P(W = \overline{w}|X = x) = 0.6 \\ & \text{Supongamos } X = \overline{x} & P(W = w|X = \overline{x}) = 0.3 \\ & & P(W = \overline{w}|X = \overline{x}) = 0.7 \end{array}$$

$$\begin{array}{rrl} \text{Para } R & \text{Supongamos } Z = z & P(R = r|Z = z) = 0.8 \\ & & P(R = \overline{r}|Z = z) = 0.2 \\ & \text{Supongamos } Z = \overline{z} & P(R = r|Z = \overline{z}) = ... \\ & & P(R = \overline{r}|Z = \overline{z}) = ... \end{array}$$

Finalmente, para los nodos Z y T las distribuciones de probabilidad a almacenar serán respectivamente

$$\begin{array}{rrl} \text{Para } Z & \text{Supongamos } X = x, Y = y & P(Z = z|X = x, Y = y) = 0.5 \\ & & P(Z = \overline{z}|X = x, Y = y) = 0.5 \\ & \text{Supongamos } X = x, Y = \overline{y} & P(Z = z|X = x, Y = \overline{y}) = 0.3 \\ & & P(Z = \overline{z}|X = x, Y = \overline{y}) = 0.7 \\ & \text{Supongamos } X = \overline{x}, Y = y & P(Z = z|X = \overline{x}, Y = y) = ... \\ & & P(Z = \overline{z}|X = \overline{x}, Y = y) = ... \\ & \text{Supongamos } X = \overline{x}, Y = \overline{y} & P(Z = z|X = \overline{x}, Y = \overline{y}) = ... \\ & & P(Z = \overline{z}|X = \overline{x}, Y = \overline{y}) = ... \end{array}$$

y análogamente,

$$\begin{array}{rrl} \text{Para } T & \text{Supongamos } W = w, Z = z & P(T = t|W = w, Z = z) = 0.1 \\ & & P(T = \overline{t}|W = w, Z = z) = 0.9 \\ ..... & & ..... \end{array}$$

□

Una vez que tenemos los valores para las distribuciones de probabilidadad condicionadas, es posible construir la distribución de probabilidad conjunta sobre las variables representadas en el grafo $X_1, \ldots, X_n$. Para ello, se hace uso de las relaciones de independencia representadas en la red. La distribución de probabilidad conjunta se puede obtener utilizando la siguiente expresión:

$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i|\Pi(X_i)$$

donde $\Pi(X_i)$ representa el conjunto de padres de un nodo X_i en la red.

Para ver cómo se construye, consideremos la red del ejemplo anterior. En este caso tenemos que:

$$P(X,Y,W,Z,T,R) = P(X) * P(Y) * P(Z|X,Y) * P(W|X) * P(T|W,Z) * P(R|Z)$$

Así, por ejemplo, la probabilidad $P(= x, Y = y, W = w, Z = z, T = t, R = r) = P(x,y,w,z,t,r)$ se obtiene como

$$\begin{aligned} P(x,y,w,z,t,r) &= P(x) * P(y) * P(z|x,y) * P(w|x) * P(t|wz) * P(r|z) = \\ &= 0.7 * 0.5 * 0.5 * 0.4 * 0.1 * 0.8 = 0.0063 \end{aligned}$$

y de forma análoga

$$\begin{aligned} P(x,\overline{y},w,z,\overline{t},\overline{r}) &= P(x) * P(\overline{y}) * P(z|x,\overline{y}) * P(w|x) * P(\overline{t}|wz) * P(\overline{r}|z) = \\ &= 0.7 * 0.5 * 0.3 * 0.4 * 0.9 * 0.2 = 0.00756 \end{aligned}$$

Por tanto, podemos considerar que la red es una representación gráfica de una distribución de probabilidad conjunta. Es suficiente con asegurarnos que ciertas relaciones de independencia que se encuentran reflejadas en la red son ciertas en la distribución (recordemos que la Proposición 1 establece que un nodo es condicionalmente independiente del resto de sus no-descendientes dado que conocemos el valor que toman sus padres). Así, si expresamos las siguientes relaciones de independencia de la red (obtenidas mediante d-separación) utilizando el formalismo probabilístico tenemos que:

Ind. Gráfica	Ind. Probabilística
(1) $< X\|\emptyset\|Y >_G$	$P(X,Y) = P(X) * P(Y)$
(2) $< W\|X\|Y,Z >_G$	$P(W\|X) = P(W\|X,Y,Z)$
(3) $< T\|W,Z\|X,Y)_G$	$P(T\|W,Z) = P(T\|W,Z,X,Y)$
(4) $< R\|Z\|X,Y,W,T >_G$	$P(R\|Z) = P(R\|X,Y,W,Z,T)$

Si utilizamos la primera relación (1) tenemos que

$$P(x) * P(Y) * P(Z|X,Y) = P(X) * P(Y) * \frac{P(X,Y,Z)}{P(X,Y)} = P(X,Y,Z)$$

y sustituyendo en la expresión que nos permite obtener la distribución conjunta tenemos que

$$P(X,Y,W,Z,T,R) = P(X,Y,Z) * P(W|X) * P(T|W,Z) * P(R|Z)$$

Aplicando el mismo razonamiento, en orden, para las relaciones de independencia (2), (3) y (4) concluimos que la red representa una factorización de una distribución de probabilidad conjunta en base a una serie de distribuciones de probabilidad condicionadas, es decir,

$$P(X,Y,W,Z,T,R) = P(X) * P(Y) * P(Z|X,Y) * P(W|X) * P(T|W,Z) * P(R|Z)$$

7.3 Redes Bayesianas y Modelos de Dependencia

Para poder considerar un grafo dirigido acíclico (GDA), al que le hemos asociado un conjunto de distribuciones de probabilidad condicionadas para cada nodo, como una representación de una distribución de probabilidad conjunta es necesario que ciertas relaciones de independencia expresadas por el grafo sean válidas en la distribución de probabilidad. Sin embargo, dado una distribución de probabilidad P, no siempre es posible construir una red que satisfaga todas las relaciones de independencia de la distribución.

En esta sección nos proponemos analizar, considerando las relaciones de independencia desde un punto de vista abstracto, las posibles correspondencias entre una representación gráfica y una distribución de probabilidad. Podemos encontrarnos con alguno de los siguientes casos.

Definición 5. I-mapa: Un GDA G se dice que es un **I-mapa o mapa de independencias**[13] de una distribución P si toda relación de d-separación en G corresponde a una relación de independencia válida en el modelo P, es decir, si dados X, Y, Z conjuntos disjuntos de vértices se tiene que

$$< X \mid Z \mid Y >_G \Longrightarrow I(X, Z, Y)_P$$

□

Dado un GDA G, que es un I-mapa de una distribución P, decimos que es un I-mapa minimal de P si al borrar alguno de su arcos, G deja de ser un I-mapa del modelo.

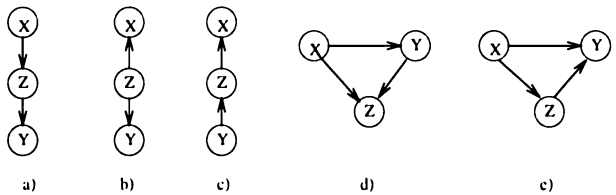
Definición 6. D-mapa: Un GDA G se dice que es un **D-mapa o mapa de dependencias** [13] de una distribución P si toda relación independencia en el modelo P se corresponde con una relación de d-separación en G, es decir, si dados X, Y, Z conjuntos disjuntos de vértices se tiene que

$$< X \mid Z \mid Y >_G \Longleftarrow I(X, Z, Y)_P$$

□

Un I-mapa garantiza que los vértices que están d-separados corresponden a variables independientes, pero no garantiza que para aquellos vértices que están d-conectados (o sea, no d-separados), sus correspondientes variables sean dependientes. Recíprocamente, en un D-mapa se puede asegurar que los vértices d-conectados son dependientes en el modelo, aunque un D-mapa puede representar un par de variables dependientes como un par de vértices d-separados. Ejemplos triviales de D-mapa e I-mapa son, respectivamente, los grafos donde el conjunto de arcos es vacío y los grafos completos (existe un arco entre cada par de vértices).

Ejemplo 13. Supongamos que P es una distribución de probabilidad donde se satistace que $I(X,Z,Y)_P$ (y su simétrica, $I(Y,Z,X)_P$. Entonces, la siguiente figura representa cinco grafos que son I-mapas de la distribución:



Los grafos a), b) y c) son I-mapas minimales, (toda independencia en el grafo es cierta en la distribución P). El grafo d) es un I-mapa trivial, por no representar ninguna relación de independencia, y además es minimal ya que si eliminamos cualquier arco aparece alguna relación de independencia que no es cierta en el modelo. El grafo e) es un I-mapa, pero no es minimal, ya que podemos eliminar el arco $X \rightarrow Y$ y la estructura resultante sigue siendo un I-mapa. □

Definición 7. Mapa-Perfecto: Un GDA, G se dice que es un **Mapa-Perfecto** [13] de una distribución P, si es I-mapa y D-mapa simultáneamente, es decir

$$< X \mid Z \mid Y >_G \Longleftrightarrow I(X,Z,Y)_P$$

□

Si un grafo G es un Mapa-Perfecto de una distribución de probabilidad, diremos que los modelos son *Isomorfos*, pudiendo hablar indistintamente de relaciones de independencia tanto en el GDA como en la distribución. Hemos de notar que no toda distribución de probabilidad tiene un grafo dirigido que le sea isomorfo.

Tanto a un GDA como una distribución de probabilidad pueden ser consideradas como un Modelo de Dependencias: "Conjunto de variables y un conjunto de reglas que permiten dar valores de verdad al predicado X *es independiente de* Y. *dado* Z".

Dado un Modelo de Dependencias cualquiera, pueden existir distintas representaciones gráficas reflejando las mismas relaciones de independencia que el modelo. En este caso decimos que las representaciones son Isomorfas, y lo notamos por $\approx$. Por ejemplo, los grafos a) b) y c) de la figura anterior son isomorfos entre si, ya que reglejan el hecho de que X e Y son marginalmente dependientes, pero conocida Z se hacen condicionalmente independientes.

7.4 Red Bayesiana: Definición

Como resultado final, podemos dar una definición formal de una red Bayesiana

Definición 8. *Una red Bayesiana es un par $(G(X,A),P)$, donde G es un grafo dirigido acíclico, X es el conjunto de vértices (o variables) en G, A el conjunto de arcos y $P = \{P(X_1 \mid \Pi_1),\ldots,P(X_n \mid \Pi_n)\}$ es un conjunto de n funciones de probabilidad condicionada, una para cada variable, y Π_i es el conjunto de padres del nodo X_i en G ($\forall Y \in \Pi_i$, $Y \rightarrow X_i \in G$). El conjunto P define una función de probabilidad asociada mediante la factorización*

$$P(X) = \prod_{i=1}^{n} P(X_i \mid \Pi_i).$$

El grafo acíclico G es un I-mapa minimal de $P(X)$ □

Por tanto, toda relación de independencia representada en la red es una relación de independencia válida en la distribución de probabilidad $P(X)$. Este hecho es de gran importancia ya que nos permite detectar fácilmente (mediante el criterio de d-separación) cuando la información que proporciona una determinada variable es relevante ante una determinada consulta.

La definición anterior nos dice que dada una red Bayesiana G es posible encontrar una distribución de probabilidad, P, siendo G un I-mapa de P. Ahora nos planteamos la relación inversa, esto es, dada una distribución de probabilidad P ¿Es posible construir una red G que sea un I-mapa de P?.

Antes de ver la respuesta a esta cuestión, presentaremos algunas consideraciones previas:

Sea P una distribución de probabilidad sobre las de variables $X_1,\ldots,X_n$ y sea θ un orden entre las variables. Notaremos por $\mathrm{Pred}_\theta(X_i)$ al conjunto de predecesores de X_i en el orden θ, es decir, $\mathrm{Pred}_\theta(X_i) = \{X_1, X_2, \ldots, X_{i-1}\}$.

Definición 9. Manto de Markov: El manto de Markov para un nodo X_i en P, con respecto al conjunto de sus predecesores $\mathrm{Pred}_\theta(X_i)$, y lo notamos por B_i, es aquel conjunto minimal que satisface que

$$B_i \subseteq \mathrm{Pred}_\theta(X_i) \text{ y } I(X_i, B_i, \mathrm{Pred}_\theta(X_i) \setminus B_i)_P$$

□

donde $\mathrm{Pred}_\theta(X_i) \setminus B_i$ representa al conjunto de predecesores de X_i que no pertenecen a B_i.

La siguiente proposición [13,19] nos permite dar respuesta a la pregunta que nos planteamos.

Proposición 3. Sea P una distribución de probabilidad sobre $X_1, \ldots, X_n$, sea θ un orden sobre las variables y sea G el grafo que se obtiene al asignar B_i como el conjunto de padres del nodo X_i en el grafo. Entonces podemos decir que G es un I-mapa minimal de P □

Si, como punto de partida, nuestra base de conocimiento viene representada por una distribución de probabilidad P, este teorema nos permite construir una red Bayesiana que sea una buena representación de P.

En conclusión, las redes Bayesianas se pueden considerar como un formalismo que permite representar eficientemente el conocimiento en un sistema experto probabilístico. Los siguientes capítulos están dedicados al estudio de cómo podemos realizar labores de razonamiento de forma eficiente con este tipo de estructuras, esto es, nos planteamos cómo se construye el motor de inferencia del sistema experto.

8 Construcción de Sistemas Expertos Probabilísticos

La base de conocimiento de un sistema experto probabilístico esta formada por un conjunto de variables y una distribución de probabilidad conjunta sobre ellas. Tenemos dos alternativas para especificar la base de conocimiento: la primera en la cual hacemos uso de una tabla de la distribución conjunta (esta aproximación resulta inabordable incluso para problemas con un número de variables pequeño) o bien hacer uso de modelos más sotisficados que (utilizando relaciones de independencia entre variables) factorizen la distribución en funciones de tamaño menor.

En este caso, los pasos que tenemos que seguir a la hora de diseñar el sistema experto son:

1. **Planteamiento del Problema**: Tener una buena definición del problema es un paso crucial a la hora de obtener un buen sistema experto, ya que de ella dependerán en gran medida la calidad de los resultados que obtengamos. Consideremos el siguiente ejemplo:

   Ejemplo 14. En una consulta médica estamos interesados en la construcción de un sistema experto que, ante un conjunto de síntomas que presenta un determinado paciente, nos ayude a determinar (a) ¿ Cuál es la enfermedad más probable ? y (b) ¿ Qué tratamiento tenemos que suministrar?.

   En concreto imaginemos la siguiente situación hipotética:
   Tenemos dos posibles enfermedades, amigdalitis y la otra más extrema, como el padecer de un cáncer en el cerebro.
   • *Si consideramos los síntomas que pueden aparecer encontramos:*

- Cuando un paciente tiene amigdalitis los síntomas que aparecen son dolor de cabeza y fiebre.
- Si el paciente tiene cáncer en el cerebro, no aparecen los síntomas hasta que no se ha producido una metástasis de las células cancerosas. En este caso, los síntomas son dolor de cabeza y mareos.
• *Por otro lado, si consideramos los tratamientos de las enfermedades tenemos que:*
- Si un paciente tiene amigdalitis, entonces se propone un tratamiento en base a penicilina (TA) con un costo bajo. Sin embargo, si este tratamiento es proporcionado a un paciente alérgico a la penicilina podemos provocarle fuertes reacciones. En este caso, se prefiere aplicar un segundo tratamiento, (TB), más costoso, pero que no le es perjudicial.
- Por otro laso, si un paciente padece de cáncer, tenemos que el tratamiento TA carece de eficacia, siendo el tratamiento TB el más aconsejable. □

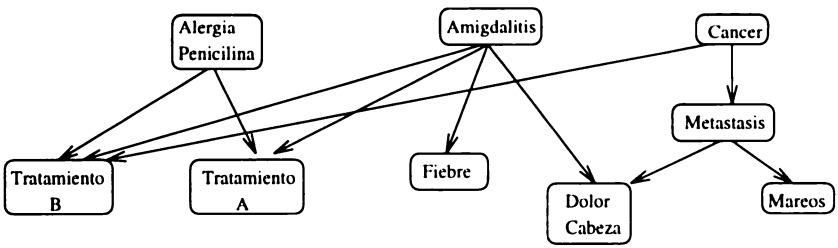
2. **Selección de Variables**: El siguiente paso consiste en seleccionar el conjunto de variables que son relevantes para tener una buena definición del problema (esta tarea debe ser realizada por los expertos en el problema a analizar).

   Ejemplo 15. En el ejemplo anterior, las variables de interés serán:
   Alergia a la Penicilina A con valores $(\{a, \overline{a}\})$; Cancer C con los casos $\{c, \overline{c}\}$; amiGdalitis G $\{g, \overline{g}\}$; Metástasis M $(\{m, \overline{m}\})$; ; Fiebre F $(\{f, \overline{f}\})$; Dolor de Cabeza D $(\{d, \overline{d}\})$; Mareo Mr $(\{mr, \overline{mr}\})$; Tratamiento A TA $(\{ta, \overline{ta}\})$ y Tratamiento B TB $(\{tb, \overline{tb}\})$.
   De forma genérica, para una variable X el caso $X = x$ expresa la idea de que se hace cierta la relación y $X = \overline{x}$ expresa que la relación es falsa, por ejemplo $C = c$ expresa la idea de que el paciente tiene cáncer y $C = \overline{c}$ indica que no tiene cáncer. □

3. **Adquisición de la información cualitativa**: Si disponemos de un experto le pediremos que muestre las relaciones de relevancia entre las variables que definen el problema. En este proceso es importante que el experto también determine las relaciones de independencia entre variables. Es de gran utilidad en esta etapa el apoyarse en modelos gráficos ya que permiten de forma explícita mostrar las relaciones de relevancia entre las variables.
   Cuando no disponemos de un experto para diseñar la estructura de dependencias, podemos utilizar técnicas que, partiendo de una base de ejemplos, permiten construir de forma automática la red.

   Ejemplo 16. Para nuestro problema, el experto nos puede dar la siguiente red:

   □



4. **Adquisición de la información cuantitativa**: Este último paso consiste en asignarle valores a las distribuciones de probabilidad conjunta que tenemos que almacenar en cada nodo en la red. En los últimos dos pasos es muy conveniente que el experto pueda colaborar con especialístas en Estadística ya que el uso de métodos específicos puede ayudar a mejorar la calidad de los datos y validar el modelo construido.

Ejemplo 17. Para nuestro problema, supongamos que tenemos las siguientes distribuciones de probabilidad condicionadas, donde por ejemplo $P(c)$ expresa la probabilidad $P(C = c)$. Notemos que sólo expresamos el valor para un único caso de la variable, el otro puede ser obtenido fácilmente. Así, si $P(c) = 0.05$ entonces $P(\overline{c}) = 1 - P(c) = 0.95$ y de forma análoga, si $P(m|c) = 0.7$) entonces $P(\overline{m}|c) = 1 - P(m|c) = 0.3$:

$$P(c) = 0.05;\ P(g) = 0.35;\ P(a) = 0.25$$

$$\begin{array}{ll} P(m|c) = 0.7 & P(m|\overline{c}) = 0.01 \\ P(mr|m) = 0.9 & P(mr|\overline{m}) = 0.09 \\ P(f|g) = 0.95 & P(f|\overline{g}) = 0.15 \end{array}$$

$$P(d|g,m) = 0.99\ P(d|g,\overline{m}) = 0.7\quad P(d|\overline{g},m) = 0.85\ P(d|\overline{gm}) = 0.15$$
$$P(ta|a,g) = 0.01\ P(ta|a,\overline{g}) = 0.01\ P(ta|\overline{a},g) = 0.99\ P(ta|\overline{ag}) = 0.20$$

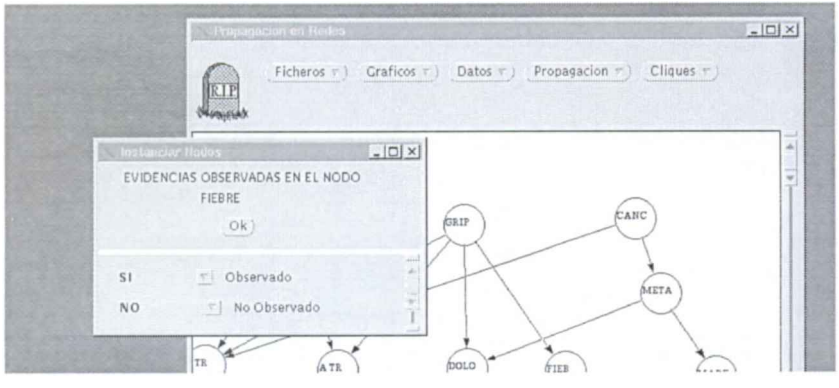
$$P(tb|a,g,c) = 0.99\ P(tb|a,g,\overline{c}) = 0.90\ P(tb|a,\overline{g},c) = 0.98\ P(tb|a,\overline{g},\overline{c}) = 0.01$$
$$P(tb|\overline{a},g,c) = 0.95\ P(tb|\overline{a},g,\overline{c}) = 0.01\ P(tb|\overline{a},\overline{g},c) = 0.95\ P(tb|\overline{a},\overline{g},\overline{c}) = 0.01$$

Para este ejemplo, si quisieramos presentar la tabla completa de la distribución conjunta necesitaremos de 512 valores, mucho mayor que los 25 que realmente hemos tenido que proporcionar. □

8.1 Usando un Sistema Experto probabilístico

Una vez construido el sistema experto, la siguiente etapa es hacer uso del mismo para realizar tareas de inferencia. Por ahora, seguiremos considerando el motor de inferencia como una caja negra encargada de realizar los cálculos.

En general, este tipo de sistemas expertos presentan como interfase de usuario un gráfico que muestra la red de dependencias, proporcionando la opción de modificar la creencia sobre el conjunto de nodos. Por ejemplo, consideremos el sistema Entorno [1] (ver la siguiente figura). Supongamos que recibimos la información de que el paciente tiene fiebre. Para incluir dicha información en la red, pinchamos sobre el botón PROPAGACION y en el menu que aparece volvemos a pinchar la opción INSTANCIAR BEL. Cuando marcamos sobre el nodo Fiebre, aparece una ventana indicando los posibles valores que aparecen. En este momento podemos decir que hemos observado que el paciente tiene fiebre (pinchando sobre la opción Si)



Para propagar la información se pincha de nuevo sobre la ventana propagar. Esto nos permite consultar los resultados particulares marcando sobre un nodo determinado, esto es, la probabilidad de la ocurrencia de ese suceso condicionado a que se conocen el conjunto de evidencias.

Como ejemplo, expresamos lo que podrían ser dos sesiones de trabajo con la red del ejemplo anterior.

Sesión I) Supongamos que un paciente nos comunica que tiene fiebre. En este caso, basta con instanciar el nodo F, es decir, actualizar la probabilidad $P(f) = 1$ y decir al sistema que propage la información sobre el resto de los nodos en el grafo.

Por ejemplo, la probabilidad de que el paciente tenga amigdalitis es ahora de $P(g|f) = 0.773$, más del doble de la probabilidad que teníamos cuando no conociamos ninguna información $P(g) = 0.35$. De igual modo, la probabilidad de que el paciente tenga cáncer no se ve modificada por este dato.

Supongamos que nos planteamos que tratamiento tenemos que aplicar. En este caso, si consultamos la creencia (después de propagar la evidencia de que el paciente tiene fiebre) para los nodos que representan los dos tratamientos vemos que $P(tb|f) = 0.221$ y $P(ta|f) = 0.611$. Es decir, tendremos que aplicar el tratamiento A.

Antes de aplicar el tratamiento, el médico puede preguntar al paciente si es alergico a la penicilina. Imaginemos que éste responde que si. En este caso, tenemos un nuevo dato, una nueva evidencia que debemos de incluir en la red. Para ello, basta con instanciar también el nodo Alergia a la Penicilina, esto es actualizar la $P(a) = 1$ y propagar la información. Como resultado, tenemos que $P(tb|a, f) = 0.713$ y $P(ta|a, f) = 0.01$. Por tanto, hemos rectificado nuestras creencia anterior, decantándonos claramente por el tratamiento B.

Sesión II) Supongamos un nuevo paciente que presenta fuertes dolores de cabeza. En este caso instanciando el nodo y propagando podemos ver que $P(c|d) = 0.1$ y $P(g|d) = 0.679$. Consultando con los valores "a priori" ($P(c) = 0.05$ y $P(g = 0.35)$) obtenemos que, si conocemos que el paciente tiene fuertes dolores de cabeza, la probabilidad de que el paciente tenga cáncer se ha duplicado y la de tener amigdalitis es 1.94 veces mayor. Ante esto, el médico puede preguntar por la presencia de nuevos síntomas. El paciente puede responder que tiene mareos. En este momento, podemos incluir nueva información al sistema y propagar, con lo que obtenemos que obtenemos que $P(c|d, mr) = 0.440$ y $P(g|d, mr) = 0.534$. Es decir, la creencia de que el paciente tenga cáncer es 8 veces la original. En cualquier caso, antes de tomar ninguna decisión consulta la presencia de fiebre en el paciente, descubriendo que este no tiene fiebre. Esta información hace que obtengamos los siguientes valores de probabilidad al propagar, $P(c|d, mr, \overline{f}) = 0.559$ y $P(g|d, mr, \overline{f}) = 0.063$. De nuevo, se incrementa la probabilidad de que el paciente tenga cáncer. Para asegurarse, el médico decide hacer una prueba más específica, y detectar si se ha producido una metástasis, obteniendo unos resultados negativos. Si incluimos también esta información obtenemos que $P(c|d, mr, \overline{f}, \overline{m}) = 0.016$ y $P(g|d, mr, \overline{f}, \overline{m}) = 0.129$. Por tanto, la creencia que tenemos en que el paciente pueda tener cancer desaparece.

9 Conclusiones

Hemos presentado a las redes Bayesianas como una herramienta que nos permite diseñar sistemas expertos probabilístos, y en concreto nos hemos centrado

en el estudio de este tipo de estructuras como mecanismo para representar la base de conocimiento de un sistema experto.

Podemos decir, haciendo uso de las relaciones de independencia presentadas en la estructura, que una red Bayesiana no es más que una representación eficiente de un conjunto de variables y una distribución de probabilidad conjunta sobre ellas. Por tanto, utilizando información local (para cada nodo sólo necesitamos conocer las distribuciones de probabilidad condicionada a los valores que toman sus padres) estas estructuras nos permiten realizar tareas de razonamiento y obtener los mismos resultados que si consideramos globalmente la información.

Las ventajas que aporta utilizar este tipo de sistemas son:

1. Todo el conocimiento del sistema se expresa con el mismo formato, en base a relaciones de relevancia entre variables. Además, este tipo de relaciones se aproximan a la forma que tiene el ser humano de representar el conocimiento.
2. La presencia de ciclos es fácilmente detectable, ya que nos apoyamos en herramientas gráficas para su diseño.
3. Permite retractarse de conclusiones obtenidas con anterioridad y que a la luz de nueva información ya no son válidas.
4. Podemos realizar inferencias bidireccionales.
5. Permiten realizar razonamiento abductivo, esto es, encontrar el mejor conjunto de causas que explican unos determinados hechos.
6. Tenemos una visión global del problema que estamos resolviendo.
7. Permiten representar de forma sencilla el hecho de que distintas fuentes de información son dependientes.
8. Como salida, podemos presentar al usuario un conjunto posible de alternativas, ordenadas desde las más probables a las menos probables. Este tipo de información puede ser de gran ayuda a la hora de tomar una decisión.

Pero también presentan inconvenientes, como por ejemplo.

1. Cuando el número de padres asociados a una variable es muy elevado, podemos tener problemas de especio para almacenar la distribución de probabilidad conjunta (es exponencial en el cardinal del conjunto de padres).
2. Cuando las estructuras son complejas, las labores de inferencia son ineficientes, necesitando del uso de algoritmos que permitan obtener una solución aproximada.
3. Generalmente, los expertos no razonan utilizando criterios probabilísticos y sin embargo, necesitamos que sean capaces de dar valores consistentes y comprensibles para las distribuciones de probabilidad condicionadas. Si estos valores se obtiene consultando una base de casos, necesitaremos de un número elevado de ejemplos. En ambos casos, el recurrir a un experto en Estadística será de gran utilidad.

Agradecimientos.

Este trabajo ha sido financiado por la Comisión Interministerial de Ciencia y Tecnología (CICYT). Proyecto n. TIC96-0781.

Referencias

1. J.E. Cano. *Propagación de probabilidades inferiores y superiores en grafos.* Tesis Doctoral. Universidad de Granada, 1992.
2. E. Castillo, J.M. Gutierrez, and A.S. Hadi. *Sistemas Expertos y modelos de redes probabilísticas.* Academia de Ingeniería, 1997.
3. A.P. Dempster. Upper and lower probabilities induced by a multivalued mapping. *Annals of Mathematics and Statistic*, 38:325–339, 1967.
4. D. Dubois and H. Prade. *Possibility Theory: An approach to computerized processing of uncertainty.* Plenum Press, 1988.
5. S. Andreassen et al. Munin - an espert emg assistant. In *Computer-aided electromyography and expert systems*, pages 255–277. J.E. Desmedt (ed.), 1989.
6. A. J. Gonzalez and D. D. Dankel. *The engineering of knowledge-based systems: Theory and practice.* Prentice-Hall, 1993.
7. G. Gorry. Computer-assisted clinical decision making. *Methods of Information in Medicine*, 12:45–51, 1973.
8. J.W. Grzymala-Busse. *Managing uncertainty in expert systems.* Kluwer Academic Publisher, 1991.
9. D. Heckerman, E. Horvitz, and B.Nathwani. Towards normative expert systems: Part I. the Pathfinder project. *Methods of Information in Medicine*, 31:90–105, 1992.
10. E. Horvitz and B. Barry. Display of information for time critical decision making. *Proc. of the eleventh conference on Uncertainty in Artificial Intelligence*, pages 296–305, 1995.
11. F.V. Jensen. *An introduction to Bayesian Networks.* UCL Press, 1996.
12. R.K. Lindsay, B.G. Buchanan, E.A. Feigenbaum, and J. Lederberg. *Applications of Artificial Intelligence for Organic Chemestry.* McGraw-Hill, 1980.
13. J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference.* Morgan and Kaufmann, San Mateo, 1988.
14. L.K. Rasmussen. BOBLO: an expert system based on bayesian networks to blood group determination of cattle. In *Research Report 16.* Research Center Foulum, PB 23, 8830 Tjele, Dennmark., 1995.
15. R.Neapolitan. *Probabilistic Reasoning in Expert Systems.* John Wiley and Sons, New York, 1990.
16. S. Ross. *A Fist Course in Probability Theory.* New York- Macmillan, 1984.
17. G. Shafer. *A mathematical theory of evidence.* Princeton University Press, Princenton N.J., 1976.
18. E.H. Shortliffe. *Computer-Based medical consultation:MYCIN.* Elsevier, New York, 1976.
19. T. Verma and J. Pearl. Causal networks: Semantics and expressiveness. In R.D. Shachter, T.S. Lewitt, L.N. Kanal, and J.F. Lemmer, editors, *Uncertainty in Artificial Intelligence 4*, pages 69–76. North-Holland, 1990.
20. L.A. Zadeh. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, (1):3–28, 1978.

Algoritmos de Propagación I: Métodos Exactos

Luis Daniel Hernández Molinero

Dpto. de Informática, Inteligencia Artificial y Electrónica
Universidad de Murcia
correo-e: ldaniel@dif.um.es

Resumen

En este trabajo se describe como fusionar y propagar el impacto de nueva evidencia a través de una Red Bayesiana de manera que las nuevas asignaciones de certidumbre sobre las variables del modelo sean consistentes con los axiomas de la probabilidad. El trabajo se centra en los denominados métodos exactos y, en particular, en las técnicas más utilizadas y relevantes: el algoritmo para poliárboles y el basado en árboles de intersecciones.

1 Introducción

Una forma de modelar el conocimiento incierto sobre un conjunto de proposiciones es mediante las medidas de probabilidad. Entre sus ventajas se encuentra que su formulación proporciona una base para establecer un *formalismo de razonamiento* sobre la creencia de las variables proposicionales del modelo [32]. Más concretamente, en este formalismo, las proposiciones tienen asignadas parámetros numéricos (probabilidades) que indican el grado de creencia atendiendo a algún tipo de conocimiento; y el razonamiento consiste en la **manipulación** de dichos parámetros atendiendo a las reglas de la probabilidad.

Básicamente son dos los operadores que permiten la manipulación de información en el formalismo probabilístico [41]:

Combinación. Dadas dos informaciones, cada una de ellas refererentes a un conjunto de proposiciones, la combinación tiene como objetivo obtener nueva información (sobre el conjunto unión de todas las proposiciones) de tal manera que (a) recoja la información compartida de las informaciones iniciales y, (b) que sea coherente con aquellas.

Marginalización. Dada una información sobre un conjunto de proposiciones, la marginalización busca cómo particularizar dicha información a un subconjunto de tales proposiciones.

Con estos dos operadores puede diseñarse el *operador de inferencia* (o razonamiento), cuyo objetivo es el siguiente: Si se tiene cierta información sobre un

conjunto de proposiciones, y si por algún medio se conoce información más concreta o específica sobre algunas de esas proposiciones, el operador de inferencia establece cómo debe de modificarse la información inicial a la luz de esos nuevos resultados. Al problema de como diseñar una técnica que implemente el operador de inferencia para que éste se realice de la forma más eficiente posible se conoce como *El Problema de la Inferencia.*

La técnica más general para realizar inferencia sobre un conjunto de proposiciones consiste en **combinar toda** la información de la que se dispone para, posteriormente, marginalizar dicha información sobre esas proposiciones. En particular, en el contexto de las redes bayesianas, sería necesario combinar todas las distribuciones de probabilidad condicionadas (informaciones locales) que se encuentran en el grafo para calcular la distribución de probabilidad conjunta (información global) y, después, calcular la distribución marginal sobre alguna de esas variables (información 'a posteriori').

Ya que la generación de información global a partir de la información local puede resultar muy ineficiente, la solución se encuentra en trabajar únicamente con pequeñas partes de conocimiento para realizar la inferencia; es decir, inferir de forma local para obtener el mismo resultado que si se hubiese realizado inferencia global. Son dos los grupos de técnicas que resuelven el problema de la inferencia local en el contexto de las redes bayesianas: (a) Métodos Exactos y (b) Métodos Aproxiamados. Sólo nos centraremos en las técnicas más relevantes y usadas del primer grupo de métodos.

Los métodos exactos se basan en la idea de conseguir las distribuciones marginales de cada variable mediante la modificación de las valoraciones de los nodos vecinos a través de *expresiones matemáticas exactas* (fórmulas) ya preestablecidas - de ahí el nombre de exactos.

Cuando en un nodo se modifica la información asociada, ésta se traspasa a los nodos vecinos a través de los arcos que los unen; éstos a su vez pasan la nueva información junto con la que ya tenían a aquellos nodos vecinos aun no modificados y así sucesivamente. Aunque la idea básica de traspasar información de un nodo a otro mediante una serie de mensajes de información parece sencilla, la realidad es que el problema es NP-completo [8] y por tanto no siempre se encuentra soluciones al problema de la inferencia local en tiempo polinomial.

No obstante se pueden encontrar estructuras sencillas en las que el tiempo de resolución es polinomial como es el caso de los grafos encadenados (un nodo tiene a lo más un sólo padre y/o un sólo hijo), árboles (cada nodo sólo tiene un padre) y poliárboles (existe un único camino que une cualquiera dos nodos del grafo). En este caso es fácil intuir que bastará realizar un recorrido de "abajo-arriba" y otro de "arriba-abajo" para recoger toda la información involucrada en el grafo y

traspasar esa información a todos los nodos. Este caso fue resuelto por Pearl [30] y será comentado más detenidamente en los apartados 3 y 8.

El problema surge cuando en el grafo se presentan ciclos ya que en tal situación no se puede realizar un recorrido por los nodos del grafo sin ciclar la información de forma indefinida. Las alternativas al problema son de distinta naturaleza y se pueden distinguir tres metodologías. La primera se basa en aplicar la propia definición del criterio de d-separación (apartado 5), la segunda consiste en en realizar modificaciones estructurales del grafo original para obtener nuevos grafos que sea computacionalmente tratables (apartado 6) y la tercera en buscar soluciones mediante técnicas de Monte Carlo, también conocidos como métodos aproximados (ver siguiente capítulo).

2 Notación y Definiciones Básicas

En este apartado se introducirá la notación y definiciones báscias que serán necesarias para el desarrollo de este trabajo.

Dado un vector de variables proposicionales $X = (X_1, \ldots, X_i, \ldots, X_n)$, donde cada una de ellas toma valores en U_i, se notará por N al conjunto de índices de dichas variables ($N = \{1, 2, \ldots, n\}$) y por $X^{\downarrow I}$ al vector $((X_i))_{i \in I \subseteq N}$; es decir, el resultado de ignorar en X las variables cuyo índice no se encuentra en I. Un valor particular de la variable X se notará por x y un valor particular de $X^{\downarrow I}$ por $x^{\downarrow I}$. Una función $f : U_I = \prod_{i \in I} U_i \rightarrow R$ definida sobre X_I se llamará un potencial sobre X_I y se notará por $s(f)$ al conjunto de índices de las variables para las que está definida f (es decir, $s(f) = I$).

Sobre dicho conjunto de variables se supondrá definida una red bayesiana $G = (N, R)$, donde cada nodo $i \in N$ se identificará con la variable proposicional X_i y tendrá definida una función de probabilidad condicionada f_i con $s(f_i) = \{i\} \cup P_i$, donde P_i denota a lo padres directos de i en G. Es decir:

$$f_i(x) = f_i(x^{\downarrow i}, x^{\downarrow P_i}) \ \forall x \in U_{s(f_i)} \text{ y verifica } \sum_{x^{\downarrow i} \in U_i} f_i(x^{\downarrow i}, x^{\downarrow P_i}) = 1 \ \forall x^{\downarrow P_i} \in U_{P_i}$$

Con esta notación, la probabilidad conjunta asociada a las variables de la red puede expresarse como $p(x) = \prod_{i \in N} f_i(x^{\downarrow s(f_i)}) \ \forall x \in U_N$.

Una observación es el conocimiento certero sobre el valor que toma una variable, en cuyo caso se dice que la variable ha sido observada. Al conjunto de índices de las variables observadas se notará por E y a la instancia que define $X^{\downarrow E}$ se notará por e y se le llamará conjunto evidencia (u observaciones o simplemente evidencia). Toda variable observada tiene asociada una función delta de Dirac

definida como sigue:

$$\delta_{e_i}(x^{\downarrow i}) = \begin{cases} 1 \text{ , si } x^{\downarrow i} = e_i \\ 0 \text{ , si } x^{\downarrow i} \neq e_i \end{cases} \quad \forall x \in U_N$$

Los operadores de combinación y marginalización se definen como siguen:

Definición 1 (Combinacion). Dados k potenciales $\{f_i\}_{i=1}^k$, se define la combinación (o producto) de éstos como el potencial f, definido sobre el conjunto de variables con índices en $K = \bigcup_{i=1}^{k} s(f_i)$, dado por

$$f(x) = \bigotimes_{i=1}^{k} f_i(x^{\downarrow s(f_i)}) = \prod_{i=1}^{k} f_i(x^{\downarrow s(f_i)}) \ \forall x \in U_K$$

Definición 2 (Marginalización). Dado un potencial f definido sobre variables con índices en I, y $J \subset I$, se define la marginalización de f en X_J como el siguiente potencial:

$$f^{\downarrow J}(x) = \sum_{\substack{y \in U_I \\ y^{\downarrow J} = x}} f(y) \quad \forall x \in U_J$$

3 Inferencia en Poliárboles mediante un Ejemplo

Existen distintos métodos exactos que utilizan la estructura de la red original para propagar información [14,21,30,29,32]. Este apartado se centrará en el el algoritmo de Pearl [30] para grafos simplemente conectados.

Considérese la red bayesiana dada por la figura 1. Se desea calcular $P(X_5|e)$ para $E = \emptyset$. Si se aplicara la técnica de inferencia global los pasos a seguir son:

1. Combinar toda la información: $p(x) \equiv \bigotimes_{i=1}^{9} f_i(x^{\downarrow\{j\}\cup P_j})$
2. Marginalizar la información global sobre X_5: $f(x_5) = \sum_{\substack{x^{\downarrow i} \\ (i \neq 5)}} \left[\bigotimes_{j=1}^{9} f_j \right]$

Sin embargo, es fácil comprobar que la última expresión es equivalente a:

$$f(x_5) = \sum_{x^{\downarrow 2}} \sum_{x^{\downarrow 4}} \Big\{ f_5 \bigotimes \tag{1}$$

$$\bigotimes \left[\left(f_4 (\sum_{x^{\downarrow 3}} f_3) \right) \otimes \left(f_2 \, (\sum_{x^{\downarrow 1}} f_1) \right) \right] \bigotimes \tag{2}$$

$$\bigotimes \left[\left(\sum_{x^{\downarrow 9}} (\sum_{x^{\downarrow 8}} f_9 \cdot f_8) \right) \otimes \left(\sum_{x^{\downarrow 7}} (\sum_{x^{\downarrow 6}} f_7 \cdot f_6) \right) \right] \Big\} \tag{3}$$

En esta reformulación llama la atención el que la expresión se descompone en tres partes: La información asociada al nodo en el que se está interesado - el potencial f_5; la que involucra a variables que se encuentran "por encima" de X_5 - expresión (2); y la involucra a variables que se encuentran "por debajo" del nodo - expresión (3). Es más:

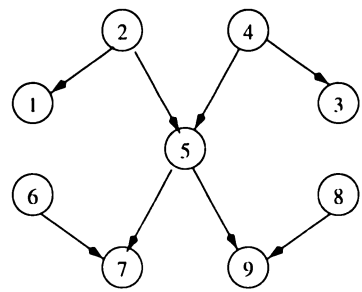


Figura 1. Un poliárbol

- La expresión (2) es un potencial definido sobre las variables (X_2, X_4, X_5) y la notaremos por $M_{P,5}$. Este potencial se llamará el mensaje (de información) que llega al nodo 5 a través de sus **P**adres.
- La expresión (3) es un potencial definido sobre X_5 y la notaremos por $M_{H,5}$. Este potencial se llamará el mensaje (de información) que llega al nodo 5 a través de sus **H**ijos.

De este modo, el cálculo de $f(x^{\downarrow 5})$ puede expresarse como sigue:

$$f(x^{\downarrow 5}) = \sum_{x^{\downarrow 2}} \sum_{x^{\downarrow 4}} [f_5 \otimes M_{P,5} \otimes M_{H,5}] \tag{4}$$

Es decir, una vez que el nodo de interés haya recibido toda la información que le suministra sus padres mediante $M_{P,5}$ y sus hijos mediante $M_{H,5}$, éstas se combinan con la información del nodo para, posteriormente marginalizar sobre la variable del nodo.

Centrémonos ahora en $M_{P,5}$[1]. Si se denota por $M_{2,5} = f_2 \sum_{x^{\downarrow 1}} f_1$ y por $M_{4,5} = f_4 \sum_{x^{\downarrow 3}} f_3$ obtenemos que $M_{P,5} = M_{2,5} \cdot M_{4,5}$. La expresión $M_{2,5}$ puede

[1] Un razonamiento totalmente análogo puede hacerse para $M_{H,5}$.

interpretarse como la información que manda el nodo 2 al nodo 5; y $M_{4,5}$ puede interpretarse como la información que manda el nodo 4 al nodo 5. Es decir, la información que recibe 5 desde sus padres es la combinación de la información que manda cada uno de sus padres.

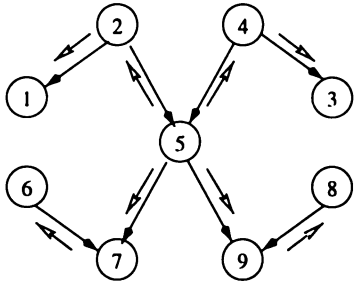


Figura 2. Petición de Información desde el nodo 5

Fijemonos más detenidamente en $M_{2,5}$[2]. Si se denota por $M_{1,2} = \sum_{x^{\downarrow 1}} f_1$ se puede expresar $M_{2,5}$ como $M_{2,5} = f_2 M_{1,2}$. De nuevo, $M_{1,2}$ puede interpretarse como la información que manda el nodo 1 al nodo 2.

Notar que, en general, el pasar información de un nodo a otro (lease $M_{2,5}$) es un proceso recursivo consistente en combinar la información que contiene el nodo (lease f_2) con la información que recibe desde el otro lado de la red (lease $M_{1,2}$). Así, desde esta perspectiva, la técnica de inferencia parece constar de los siguientes pasos:

1) Cuando un nodo requiere información de sus vecinos, éstos vuelven a realizar un requerimiento a sus vecinos excepto para el nodo vecino que hizo el requerimiento.

En el ejemplo, el nodo 5 necesita información de sus hijos (nodos 7 y 9) y de sus padres (nodos 2 y 4) para poder calcular $f(x^{\downarrow 5})$. De esta manera el nodo 7 pide información a sus nodos vecinos excepto para el nodo 5; es decir al nodo 6. De la misma forma el nodo 9 pide información al nodo 8, el nodo 2 al nodo 1 y el nodo 4 al nodo 3 (ver figura 2).

2) Cuando ya no hay más nodos a los que pedir más información el proceso se invierte; pero ahora cada nodo al que se le habia pedido información manda un mensaje con información concreta al que se lo solicitaba. Este proceso se realiza hasta llegar al primer nodo que hizo el requerimiento. En este punto, el nodo que hizo el requerimiento recibirá una serie de mensajes de sus vecinos. Al algoritmo consistente en los pasos **1)** y **2)** se llamará RecogerEvidencia.

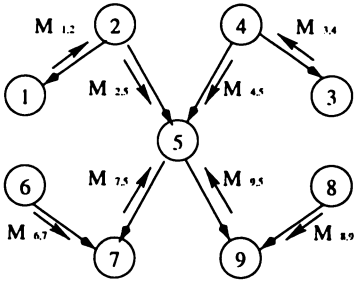


Figura 3. Recogida de Información para el nodo 5

[2] Un razonamiento totalmente análogo puede hacerse para $M_{4,5}$.

En el ejemplo, el segundo paso de RecogerEvidencia consta de los siguientes pasos: el nodo 6 manda la información que le requería el nodo 7 y éste manda información al nodo 5, el nodo 8 manda la información que le requería el nodo 9 y éste manda información al nodo 5, y así sucesivamente (ver figura 3).

3) Por último, el nodo que hizo el requerimiento, combina toda la información que recibe con la que él ya poseía para marginalizar en la variable de interés. En el ejemplo, ver expresión (4).

Así, para calcular todos los $f(x^{\downarrow i})$ bastaría repetir de forma análoga los tres pasos anteriores, y es fácil observar que entonces muchos mensajes son comunes. Por ejemplo, para calcular $f(x^{\downarrow 4})$ se utilizan los mismos mensajes usados para el cálculo de $f(x^{\downarrow 5})$ salvo el mensajes $M_{4,5}$ que se sustituye por un mensajes $M_{5,4}$.

Un modo de optimizar el algoritmo consiste en introducir el algoritmo DistribuirEvidencia consiste en: "un nodo envía mensajes a todos sus vecinos quienes, recursivamente, envían mensajes a todos sus vecinos excepto el que mandó el mensaje". Los algoritmos RecogerEvidencia y DistribuirEvidencia (y en este orden) se utilizan entonces para pasar los mensajes de una forma organizada. En el ejemplo, si se realiza RecogerEvidencia desde 1 y DistribuirEvidencia desde 1 se obtienen los grafos de la figura 4 donde los números asociados a cada mensaje representa el orden en los que estos se envían. El algoritmo de inferencia para poliárboles puede verse con detalle en la sección 8.

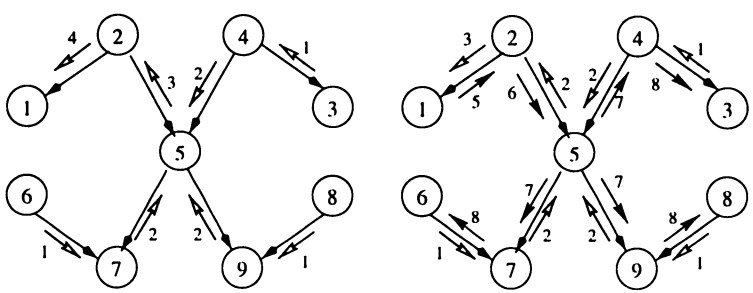


Figura 4. Recogida y Distribución de evidencia para el nodo 1

4 ¿Qué Ocurre cuando hay Ciclos?

El método presentado está limitado al uso de retículos simplemente conectados. Sin embargo, lo normal es que se presenten grafos con conexión múltiple (con ciclos). Esta aparición de ciclos hace que el método para poliárboles no sea apropiado por dos razones:

1. Los padres de un nodo pueden compartir información; esto es, cada padre no puede influir independientemente de los demás sobre la probabilidad de sus hijos comunes. Así, si se usase el algoritmo para poliárboles, se pueden obtener cálculos incorrectos en la probabilidad, a no ser que la información compartida por los dos padres esté interceptada por un nodo que produzca d-separación.
2. Aún suponiendo que las fórmulas fuesen válidas para el caso de grafos con ciclos, el método de propagación haría que la información ciclase indefinidamente. Por ejemplo, el algoritmo **RecogerEvidencia** sólo finaliza cuando se alcanza un nodo que no puede enviar más peticiones, por lo que, si se entrara en un ciclo, **RecogerEvidencia** nunca finalizaría.

Aunque el problema es irresoluble desde el punto de vista del método de Pearl, diversos autores han propuesto métodos alternativos o generalizaciones de aquel que permiten obtener resultados aún con ciclos en el grafo. Todos ellos pueden dividirse en tres grandes grupos:

- Métodos basados en condicionamiento.
- Métodos basados en modificaciones estructurales.
- Métodos aproximados (ver siguiente capítulo).

5 Métodos Basados en Condicionamiento

Estos métodos se basan en la idea de transformar el grafo en un poliárbol dando valores concretos a un conjunto de variables. Más concretamente, si se consiguiera seleccionar un conjunto de nodos $D = \{d_1, \ldots, d_p\}$ con la única condición de que bloqueen (d-separen) todas aquellas dependencias que no permiten aplicar el método para poliárboles, entonces, si se instanciara la variable X_D a una posible configuración, $x^{\downarrow D}$, se conseguiría un grafo simplemente conectado. Parece claro entonces que, en principio, el conjunto $\{d_1, \ldots, d_p\}$ debería de contener al menos un nodo de cada ciclo que exista en el grafo con objeto de que la instanciación de X_{d_i} a un valor $x^{\downarrow d_i}$ permita eliminar el flujo cíclico de la información, en el ciclo del cual d_i es su representante, así como la dependencia entre nodos. Una vez instanciadas las variables se obtendrá un poliárbol y podrá entonces aplicarse el algoritmo de Pearl para la evidencia $e \cup x^{\downarrow D}$.

Pero instanciar los valores del vector X_D a un sólo valor es considerar tan sólo uno de los posibles casos de simplificación el problema. Para conseguir todos los posibles casos, deberá instanciarse X_D a todas sus posibles configuraciones. En cuanto al modo de obtener la solución global como combinación de todas las posibles intancias, ésta viene dada por la expresión:

$$Bel(x^{\downarrow i}) = P(x^{\downarrow i}|x^{\downarrow E} = e) = \sum_{x^{\downarrow D} \in U_D} P(x^{\downarrow i}|x^{\downarrow E}, x^{\downarrow D}) P(x^{\downarrow D}|x^{\downarrow E}) \quad \forall x^{\downarrow i} \in U_i \tag{5}$$

Al conjunto D se le conoce por *conjunto de (nodos de) corte de ciclos* y al conjunto de variables $\{X_{d_1}, \ldots, X_{d_p}\}$ *variables de corte de ciclos.*

Dependiendo de cómo se trabaje con la expresión (5) se obtienen dos grupo de técnicas:

1. Si (5) se interpreta como el proceso de seleccionar una serie de variables "llave", considerar sus valores, derivar las consecuencias de esos valores, integrar las consecuencias y marginalizar en todas las variables X_D, entonces se dice que se aplican un MÉTODO DE CONDICIONAMIENTO GLOBAL [42,43]. Pearl [29,31,33] utiliza el término *razonamiento por suposiciones, razonamiento hipotético* o *razonamiento por casos* para indicar este mecanismo.
2. Sin embargo, la expresión (5) puede refinarse si se aplica la distributividad. En este caso, cuando se considera un nodo i que pertenezca a un conjunto de ciclos con nodos de corte $c(i)$, solo se marginaliza en $X^{\downarrow c(i)}$ las probabilidades condicionadas que contienen a las variables $X_{c(i)}$. De este modo se consigue no tener que marginalizar la probabilidad conjunta sobre $x^{\downarrow D}$ (como ocurre en el condicionamiento global) sino sobre distintos subconjuntos del conjunto de corte de ciclos. Estos métodos se conocen como MÉTODOS BASADOS EN CONDICIONAMIENTO LOCAL (ver [12] para más detalles).

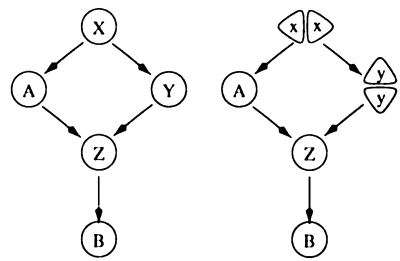


Figura 5. Eligiendo de modo apropiado los nodos (p.e. X o Y) se puede obtener siempre un poliárbol. Notar que Z no es un nodo válido.

Como ejemplo consideremos la figura 5. Si se considera como conjunto de corte el nodo X y como variable de interés el nodo B, entonces (5) se expresa, para

$E = \emptyset$, como

$$Bel(b) = \sum_x \left\{ \sum_{a,y,z} f_X(x) f_A(a,x) f_Y(y,x) f_Z(z,a,y) f_B(b,z) \right\} \quad (6)$$

$$= \sum_z \left\{ f_B(b,z) \sum_{a,y} \left[\sum_x (f_Z(z,a,y) f_X(x) f_A(a,x) f_Y(y,x)) \right] \right\} \quad (7)$$

La expresión (6) respondería a la metodología del condicionamiento global y la expresión (7) respondería al condicionamiento local. Notese que en el primer caso se mariginaliza sobre X después de realizar el producto de las f.d.p. mientras que en el sugundo caso, se marginaliza sobre X cuando se han agrupado todas las variables involucradas en el ciclo.

En principio, no hay restricciones en tomar más de un nodo en cada ciclo siempre y cuando éstos permitan romper los ciclos; sin embargo, como la complejidad de la expresión (5) es exponencial en el número de nodos del D, interesará tomar el menor número posible de nodos de corte (es decir, D deberá ser minimal en el sentido de que el producto del número de valores de las variables asociadas a los nodos de corte sea minimal). Si bien el problema de encontrar el conjunto de corte es NP-completo [42], en muchas ocasiones se puede encontrar un pequeño conjunto de nodos que es minimal o próximo al minimal [42].

6 Métodos Basados en Modificaciones Estructurales

Éstos se basan en realizar cambios en la estructura del grafo de forma que las nuevas estructuras contengan la misma información que la red bayesiana original [1,3,4,19,20,26,40,10,27,9,37,35,36]. En general, existen dos grandes subgrupos: Los basados en eliminación de variables y los basados en árboles de cliques.

6.1 Métodos Basados en Eliminación de Varibles.

Estos métodos se basan en que pueden eliminarse las variables en una secuencia dada para obtener la probabilidad 'a posteriori' de un conjunto de variables de interés. Los distintos métodos se diferencian entre sí en el modo en que definen la secuencia de eliminación [27].

Para entender la técnica general considere de nuevo el ejemplo del apartado 3. Se observa que la única función que contiene a la variable X_1 es f_1 y sobre ésta se realiza una marginalización - ver expresión (2). Después de esta operación lo que se obtiene es la función: $g(x_2) = \sum_{x^{\downarrow 1}} f_1(x^{\downarrow 1}|x^{\downarrow 2})$. Notar que la variable X_1 *ha sido eliminada*: no existe ninguna otra función que contenga a la variable X_1.

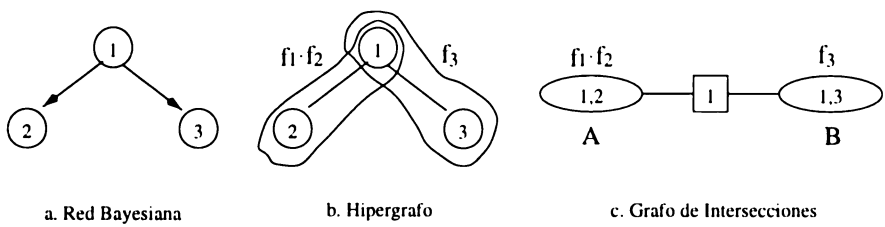
De forma análoga la única función que contiene a la variable X_3 es f_3, y, después de realizar la marginalización, no existe ninguna otra función que contenga a X_3. Fijemonos ahora en el término $(\sum_{x^{\downarrow 9}} (\sum_{x^{\downarrow 8}} f_9 \cdot f_8))$ de (3). Notar que las únicas funciones que contienen a la variable X_8 son f_8 y f_9, que tras combinarlas y sumar en $x^{\downarrow 8}$ se obtiene una nueva función g' definida en (X_5, X_9). De nuevo, no existen ninguna otra función que contenga a la variable X_8 después de este cálculo: se ha conseguido eliminar la variable X_8. En general, la eliminación de una variable i consiste en sustituir todas las funciones que contienen a dicha variable, $H(i)$, por la función que se obtiene después de:

1. Combinar todas las funciones de $H(i)$,
2. Marginalizar en $x^{\downarrow i}$

Se puede demostrar que si se realiza la eliminación de todas las variables con índices en $N - I$, el resultado será la probabilidad marginal de $X^{\downarrow I}$. La eficiencia del algoritmo vendrá dada por el orden que se considere en la eliminación de nodos y este problema coincide con el conocido problema de la triangulación, que será estudiado en el apartado siguiente.

6.2 Métodos Basados en Árboles de Cliques.

Estos métodos se basan en la idea de agrupar de forma adecuada las variables involucradas en la red causal, formar un grafo acíclico dirigido relacionando entre sí estos conjuntos y aplicar un tratamiento semejante al utilizado en poliárboles [32,26,1,41,38,17].



a. Red Bayesiana b. Hipergrafo c. Grafo de Intersecciones

Figura 6. Una red bayesiana sencilla

Cosidérese la red bayesiana de la figura 6.a. Si se combinan f_1 y f_2, las variables de las funciones $f_1 \cdot f_2$ y f_3 pueden representarse graficamente como se muestra en la figura 6.b. y que recibe el nombre de hipergrafo [41]. Dicha representación es equivalente a la figura 6.c, donde:

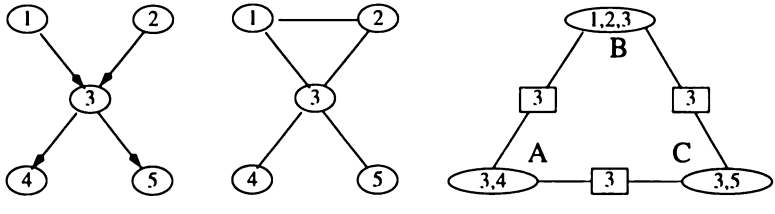
- Los nodos representan a conjuntos de variables. A los nodos de este tipo se les llaman clusters o grupos (de variables). En la figura 6.c los grupos son $A = \{1,2\}$ y $B = \{1,3\}$.
- Cada nodo contiene la combinación de algunas funciones de la red inicial, lo que define un potencial sobre las variables de cada grupo. En la figura 6.c los potenciales son $\psi_A = f_1 \cdot f_2$ y $\psi_B = f_3$.
- La etiqueta del enlace representa a la variable compartida por los grupos y recibe el nombre del separador. En la figura, el separador está formado por la variable X_1.

Un grafo de este tipo recibe el nombre de **grafo de intersecciones** o **de grupos**.

La probabilidad marginal de X_A puede expresarse en términos de los potenciales: $P(X_A) \equiv (f_3^{\downarrow 1}) \otimes (f_1 \cdot f_2) = \psi_b^{\downarrow 1} \otimes \psi_A$. Si se interpreta $\psi_B^{\downarrow 1}$ como el mensaje que manda el grupo B al grupo A, dicha expresión indica que $P(X_A)$ es el resultado de combinar el mensaje que recibe A con el potencial asociado a A. Análogamente, $P(X_B) \equiv \psi_A^{\downarrow 1} \otimes \psi_B$ viene a decir que $P(X_B)$ se obtiene de combinar el mensaje que recibe B con el potencial asociado a B. Notar que los mensajes son las marginalizaciones de los potenciales en el separador y, en general, el mensaje que recibe un grupo B desde un grupo vecino A es el resultado de combinar todos los mensajes que recibe A con el potencial asociado a A y, despues marginalizar sobre el separador de A y B (ver expresión 13).

Obsérvese que lo expuesto no es más que:

1. Aplicar el esquema de propagación de mensajes para RecojerEvidencia(A) y DistribuirEvidencia(A).
2. Para cada grupo, combinar su potencial con los mensajes que recibe. A este paso se le llama AbsorberEvidencia.



a. Red Bayesiana b. Grafo Moral c. Grafo de Grupos

Figura 7. Una red bayesiana donde el grafo de grupos presenta un ciclo

En la figura 7 se tiene una red bayesiana cuyo grafo de intersecciones no es un árbol. Esto significa que el paso de mensajes no puede usarse directamente.

En este caso A mandaría un mensaje a B y C; después B enviaría un mensaje a C. Sin embargo esto es redundante. En efecto, el mensaje que mandaría A viene dado por $\psi_A^{\downarrow 3}$, que es necesario para calcular $P(X_C)$. Sin embargo, C recibiría la información $\psi_A^{\downarrow 3}$, por un lado, directamente desde A y, por otro, indirectamente a través de B. En este caso puede romperse el ciclo borrando cualquiera de los enlaces ya que, si un enlace se elimina, siguen quedando caminos a través de los cuales la información contenida en A puede alcanzar cualquier parte del grafo. Después de eliminar alguno de los enlaces el grafo resultante recibe el nombre de **árbol de intersecciones** o **de grupos** y, sobre éste podrá realizarse el esquema de propagación. Formalmente,

Definición 3. Un árbol de intersecciones es un árbol no dirigido formado por grupos verificando la *propiedad de intersección* [2]: si para cada par de nodos Γ y Γ' con intersección no vacia $S = \Gamma \cap \Gamma' \neq \emptyset$ verifica que todos los grupos del camino que los unen contiene a S.

El grafo de la figura 7.b se llama *grafo moral*, y se obtiene a partir de la red bayesiana uniendo entre si todos los padres que tienen hijos comunes (de ahí el nombre) e ignorando la dirección de los arcos. El grafo moral se utiliza como estructura auxiliar sobre la que se pueden "leer" los grupos de variables que deben considerarse en la construcción del árbol de grupos. Notar que el establecer enlaces entre los padres de un nodo permite mantener las dependencias que se pierden al eliminar la direccionalidad de los enlaces. Además el grafo moral tiene la propiedad de que todas las independencias que refleja éste son también independencias en el grafo original (aunque algunas independencias del grafo original pueden no estar en el grafo moral).

Entre los árboles de grupos asociados a una red causal llaman especial atención los **árboles de cliques**. Un clique de un grafo es un subconjunto maximal de nodos donde todos están relacionados (son subgrafos completos). Puede demostrarse que un árbol de grupos es un árbol de cliques si y solo si no hay clusters que sean subconjuntos propios de un grupo vecino. Así, para obtener el árbol de cliques a partir de un árbol intersección de un grafo triangulado basta incluir los grupos mas pequeños en los "supergrupos"que los contengan hasta que no haya grupos que incluir [38]. Los grafos de grupos de las figuras 6.c y 7.c están formado por cliques.

Sin embargo, no siempre pueden obtenerse grafos de cliques a partir del grafo moral. Los árboles de cliques sólo pueden obtenerse cuando el grafo del cual se obtiene los grupos - el grafo moral - está *triangulado* (cualquier ciclo de longitud mayor que 3 tiene una cuerda) [2,20].

Ya que los árboles de cliques vienen caracterizados para redes trianguladas, el modo de actuar para obtener uno de tales árboles consiste en: (a) Moralizar la red

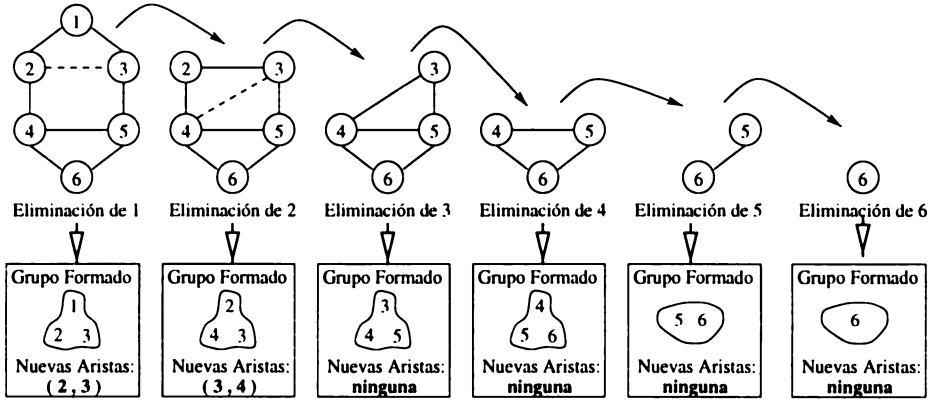


Figura 8. Eliminación de nodos para la figura 9.b

bayesiana, (b) triangular el grafo moral, (c) detectar los cliques, (d) construir un árbol con los cliques y (e) asociar potenciales a los cliques. En estas condiciones se podrá entonces aplicar el esquema de propagación.

El proceso de la triangulación consiste en añadir arcos extras a un grafo no dirigido hasta conseguir que se transforme en un grafo triangulado. Por otro lado, la **eliminación** de un vértice i en un grafo no dirigido es el proceso por el cual

1. se añaden las aristas necesarias para que el vértice y sus nodos adyacentes formen un subgrafo completo C_i y
2. se borra el vértice con los arcos incidentes en él.

La **triangulación** puede verse como un proceso consistente en añadir arcos extras (al grafo original) producidos por la eliminación de todos los vértices del grafo. Desde este punto de vista las técnicas de la triangulación de grafos consiste en establecer ordenaciones de los vértices que especifican la secuencia en la cual deberán de eliminarse; por ello, a estos algoritmos también se les denominan algoritmos de búsqueda u ordenación.

En la figura 8 puede verse el resultado de eliminar los nodos de la figura 9.b para el orden $(1,2,3,4,5,6)$. El resultado de la eliminación es que se se han creado las aristas $2-3$ y $3-4$. Añadiendo éstas al grafo 9.b, se obtiene el grafo triangulado de la figura 9.c. Las figuras 9.d y 9.e responden a la triangulacion de la red para otras ordenaciones. Notar también que distintas ordenaciones pueden dar lugar a una misma triangulación. En las figuras 9.c, 9.d, 9.e se muestran dos ordenaciones que generan el mismo grafo triangulado.

El árbol de cliques se construye entonces seleccionando del grafo triangulado los subgrafos maximales. Hay un modo muy fácil de **identificar los cliques** en

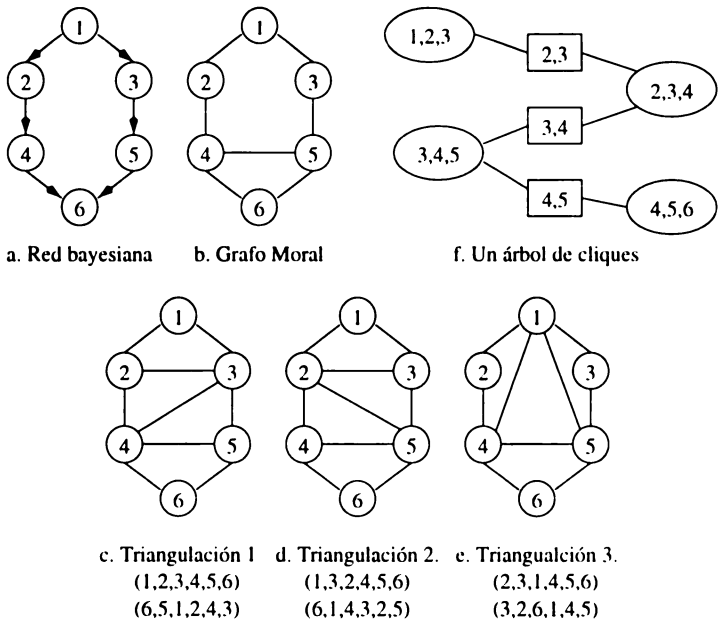


Figura 9. Distintas triangulaciones para una red bayesiana

el grafo triangulado: Si $(1, 2, \ldots, n)$ es la secuencia de eliminación empleada para obtener el grafo triangulado, entonces los subgrafos completos obtenidos en el paso 1 de la eliminacion y que sean maximales son los cliques. Por ejemplo, para la figura 8, los cliques son: $(1, 2, 3)$, $(2, 3, 4)$, $(3, 4, 5)$ y $(4, 5, 6)$. Posteriormente **se unirán** éstos exigiendo la propiedad de intersección. También hay técnica sencilla para su construcción. En primer lugar se ordenan los cliques, por ejemplo en el orden en que se han generado. Posteriormente, cada vez que se añada un nuevo clique al grafo, éste se enlazará con aquel clique que defina un separador mayor [18].

Igualmente fácil es asignar **potenciales** a los cliques. Los pasos son:

- Asignar cada función f_i a un clique C que contenga a las variables sobre la que está definida.
- Caso de existan cliques C que no tengan asociadas funciones f_i se define $\psi_C = 1$, en otro caso se define ψ_C como el producto de todas las f_i asociadas.

Notar que definiendo así los potenciales la probabilidad conjunta de las variables de la red puede expresarse en términos de los potenciales:

$$p(x) \equiv \bigotimes_{i=1}^{n} f_i = \bigotimes_{C} \psi_C$$

Es decir, el árbol de cliques contiene la misma información probabilística que la contenida en la red bayesiana pero expresada de otra manera. Y, como en la nueva representación no se presentan ciclos, puede desarrallarse el paso de mensajes de forma análoga al esquema expuesto en poliárboles (ver apartado 9 para más detalles).

El único paso problemático en el proceso para obtener un árbol de cliques es el de la triangulación. El que se pueda obtener una triangulación a partir de una secuencia de eliminación no es problema, pero sí lo es el que la secuencia puede afectar a la eficiencia del algoritmo de propagación. En el árbol de cliques, cada grupo tiene un potencial (tabla de valores) asociado. El tamaño del potencial (número de valores de la tabla) es el producto del número de estados de las variables. Así, el tamaño incrementa exponencialmente con el tamaño de los cliques. Una buena triangulación será, en consencuencia, una triangulación que produce potenciales de tamaño "pequeño".

El problema de determinar una triangulación óptima es NP-completo [45]. Se pueden encontrar distintas metodologías para encontrar buenas triangulaciones [34,44,13,22-24,15,25,5] pero son las basadas en heurísticas las que han presentado mejores resultados. Por ejemplo: eliminar sucesivamente el primer nodo que no necesite la creación de nuevos arcos, si hay empates seleccionar el que produzca un menor tamaño.

7 Comentarios Finales

El método para poliárboles resulta fácil de implementar y trabaja en tiempo polinomial. Estas ventajas se deben precisamente al tipo de grafos con que trabaja (cadenas, árboles y poliárboles) lo que, a su vez, limita su uso ya que este método no puede utilizarse en grafos con ciclos.

El métodos basados en condicionamiento presentan la ventaja de que pueden trabajar con ciclos. A cambio se ven obligados a instanciar un conjunto de variables para poder aplicar el método para poliárboles. Para dicha instanciación se utiliza como expresión básica la ecuación (5), que produce un aumento en memoria y tiempo computacional ya que (*a*) el cálculo de (5) crece exponencialmente en el número de nodos de corte; (*b*) Necesita realizar varias pasadas por el grafo

para poder determinar el conjunto de nodos de corte de ciclos, (c) es necesario aplicar el método de Pearl para cada poliárbol que produce cada instanciación de los nodos de corte.

El método basado en eliminación de nodos permite trabajar con ciclos y es muy rápido, pues sólo se basa en la información que suministra una variable. Además, pueden adaptarse para trabajar con otras teorías. Su inconveniente es que es necesario repetir cálculos si se desea obtener distintas probabilidades 'a posteriori' y presentan el mismo inconveniente que el problema de la triangulación (determinar una secuencia de eliminación).

Los algoritmos basados en árboles de cliques también permiten trabajar con ciclos, pero se encuentra con una fuerte limitación: cómo determinar los mejores cliques. Estos algoritmos presentan más ventajas que inconvenientes ya que:

- Presenta una metodología que puede extenderse a otras teorías de tratamiento de la incertidumbre sin importar la estructura del grafo utilizado extendiendo, por tanto, el esquema de propagación a casi cualquier tipo de dominios [41,11,7,6,16,16].
- Se ha demostrado que los métodos exactos pueden expresarse en términos de esta técnica [38]. El punto en común entre los distintos métodos exactos es que todos construyen un grafo de grupos, y la diferencia estriba en que cada uno busca el que resulta más adecuado para el esquema de inferencia que define. Es decir, las diferencias entre los métodos pueden entenderse como diferentes aproximaciones para desarrollar las mismas tareas en el *algoritmo general* de construcción de árboles de grupos.

8 Anexo 1: Esquema de Propagación para Poliárboles

Pearl desarrolló un método de modificación de las distribuciones de los nodos de un árbol, generalizando su desarrollo a poliárboles y posteriormente a grafos dirigidos acíclicos en general [30,33]. En este apartado se presenta dicha técnica readaptada al planteamiento de [28].

Fórmulas básicas de propagación para Poliárboles

1. La distribución a posteriori de un nodo X_i viene dada por:

$$Bel(x^{\downarrow i}) = P(x^{\downarrow i} \mid e) = \alpha\lambda(x^{\downarrow i})\pi(x^{\downarrow i}) \quad \forall x \in U_N \tag{8}$$

2. El λ-valor de X_i viene dado por:

$$\lambda(x^{\downarrow i}) = \begin{cases} \prod_{h \in H_i} M_{hi}(x^{\downarrow i}) & Si\ i \notin E \\ 1 & Si\ i \in E\ \ y\ \ x^{\downarrow i} = e_i \\ 0 & Si\ i \in E\ \ y\ \ x^{\downarrow i} \neq e_i \end{cases} \quad \forall x \in U_N \tag{9}$$

Donde H_i denota a los hijos de i.

3. El π-valor de X_i viene dado por:

$$\pi(x^{\downarrow i}) = \sum_{x \in \Omega} \left[f_i(x^{\downarrow i} | x^{\downarrow P_i}) \prod_{f \in P_i} M_{fi}(x^{\downarrow f}) \right] \quad \forall x \in U_N \tag{10}$$

Donde P_i denota a los padres de i.

4. Cada hijo h de i le envía un λ-mensaje $M_{hi}(x^{\downarrow i}) = \lambda_h(x^{\downarrow i})$ y si h tiene como padres a P_h, entonces, para cada $x \in U_N$, éste viene dado por:

$$M_{hi}(x^{\downarrow i}) = \begin{cases} \sum\limits_{x^{\downarrow P_h}} \lambda(x^{\downarrow h}) f_h(x^{\downarrow h} | x^{\downarrow P_h}) \left[\prod\limits_{r \in P_h - \{i\}} M_{rh}(x^{\downarrow r}) \right] & Si\ i \notin E \\ 1 & Si\ i \in E\ \ y\ \ x^{\downarrow i} = e_i \\ 0 & Si\ i \in E\ \ y\ \ x^{\downarrow i} \neq e_i \end{cases} \tag{11}$$

5. Cada padre f de i le envía un π-mensaje $M_{fi}(x^{\downarrow f}) = \pi_i(x^{\downarrow f})$ que viene dado por:

$$M_{fi}(x^{\downarrow f}) = \begin{cases} \frac{P(x^{\downarrow f} | e)}{M_{if}(x^{\downarrow f})} & Si\ f \notin E \\ 1 & Si\ f \in E\ \ y\ \ x^{\downarrow f} = e_f \\ 0 & Si\ f \in E\ \ y\ \ x^{\downarrow f} \neq e_f \end{cases} \quad \forall x \in U_N \tag{12}$$

Algoritmo de Propagación para Poliárboles

Inicialización

1. Tomar todos los λ-valores, λ-mensajes y π-mensajes igual a 1.
2. Para cada nodo raiz del grafo, hacer $\pi(x) = P(x)$.
3. Para cada nodo raiz del grafo, enviar un π-mensaje a cada uno de sus hijos, es decir ir a Modificación.

Modificación

1. Si $i \in E$ y $X_i = x_i = e_i$ entonces:
   (a) $Bel(X^{\downarrow i}) = \begin{cases} 1\ si\ x^{\downarrow i} = e_i \\ 0\ si\ x^{\downarrow i} \neq e_i \end{cases}$
   (b) Calcular $\lambda(X^{\downarrow i})$ según (9)
   (c) Enviar a cada uno de sus padres un λ-mensaje, según (11), ir a Modificación.
   (d) Enviar a cada uno de sus hijos un π-mensaje, según (12), ir a Modificación.
2. Si i recibe un λ-mensaje de uno de sus hijos y si X_i **NO** está instanciado, entonces:
   (a) Calcular $\lambda(X^{\downarrow i})$ según (9).
   (b) Calcular $Bel(X)$ según (8).

(c) Enviar a cada uno de sus padres un λ-mensaje, según (11), ir a Modificación.

(d) Enviar a cada uno de sus hijos un π-mensaje, según (12), ir a Modificación.

3. Si i recibe un π-mensaje de uno de sus padres, entonces

(a) Si $X^{\downarrow i}$ **NO** está instanciado entonces:

i. Calcular su π-valor según (10).

ii. Calcular $Bel(X^{\downarrow i})$ según (8).

iii. Enviar a cada uno de sus hijos un π-mensaje, según (12), ir a Modificación.

(b) Si $\lambda(X^{\downarrow i}) \neq (1, \ldots, 1)$ entonces

i. Enviar a todos los padres, excepto del que recibe el π-mensaje, un λ-mensaje, según (11), ir a Modificación.

9 Anexo 2: Esquema de Propagación para Árboles de Cliques

Shachter et. al. [38] desarrollaron un método general de inferencia para cualquier grafo basado en los resultados de [39,40]. Las operaciones básicas y el algoritmo de inferencia son como sigue:

Fórmulas básicas de propagación para Árboles de Cliques

Supongase que para una red bayesiana ya se ha construido un árbol de cliques. Considérese S como el separador entre un clique C y un clique vecino D. Denotemos por L_{Cl} la lista de las funciones f_i de la red bayesiana que han sido asignadas a cualquier clique Cl.

1. Se define el potencial asociado a cualquier clique Cl como:

$$\psi_{Cl}(x^{\downarrow Cl}) = \begin{cases} \prod_{f_i \in L_{Cl}} f_i(x^{\downarrow s(f_i)}) & \text{si } L_{Cl} \neq \emptyset \\ 1 & \text{si } L_{Cl} = \emptyset \end{cases} \quad \forall\, x \in U_N$$

2. Se define el mensaje que recibe el clique C por parte de D a través de S como

$$M^S_{D,C}(x^{\downarrow S}) = \sum_{x^{\downarrow D-S}} \left[\psi_D(x^{\downarrow D}) \prod_{S' \in Sep(D)-\{S\}} M^{S'}_{\cdot,D}(x^{\downarrow S'}) \right] \quad \forall\, x \in U_N \quad (13)$$

$$= \left[\psi_D \otimes \left(\bigotimes_{S' \in Sep(D)-\{S\}} M^{S'}_{\cdot,D} \right) \right]^{\downarrow S} \quad (14)$$

donde $Sep(D)$ representa a los separadores de D y $M^{S'}_{\cdot,D}$ representa el mensaje que recibe D a través de S'. $M^{S}_{D,C}(x^{\downarrow S})$ también se define como el mensaje que manda el clique D hacia C a través de S.
Analogamente, se define el mensaje que recibe el clique D por parte de C a través de S como

$$M^{S}_{C,D}(x^{\downarrow S}) = \left[\psi_C \otimes \left(\bigotimes_{S' \in Sep(C)-\{S\}} M^{S'}_{\cdot,C}\right)\right]^{\downarrow S} \tag{15}$$

3. Absorción.
Dado un universo Cl, se dice que absorve información si recoje toda la información de todos los mensajes que recibe. Es decir, el potencial de Cl, ψ_{Cl}, queda modificado según la expresión:

$$\psi^{nuevo}_{Cl} = \psi^{viejo}_{Cl} \times \prod_{S \in Sep(Cl)} M^{S}_{\cdot,C} \tag{16}$$

donde $Sep(C)$ denota al conjunto de separadores del universo C.
4. IntegrarEvidencia.
Integrar la evidencia e en un árbol de grupos consiste en restringir los potenciales de los cliques a dicha evidencia. Algoritmicamente puede expresarse como sigue: Para cada $i \in E$ determinar todos los grupos que contienen a la variables i. Si C es uno de tales grupos redefinir ψ_C como $\psi_C \otimes \delta_{e_i}$.
5. RecojerEvidencia(RE).
Si un clique C recibe una petición RE de un clique C_p, entonces C envia una petición RE a todos sus vecinos excepto a C_p; cuando todos los vecinos C_q de C terminan la tarea encomendada, entonces C recoje los mensajes $M^{C_q \cap C}_{C_q,C}$ de todos sus vecinos C_q y manda el mensaje $M^{C_p \cap C}_{C,C_p}$ a C_p.
6. DistribuirEvidencia (DE).
Si un clique C recibe una petición DE de un clique C_p, entonces C recoge el mensaje $M^{C \cap C_p}_{C_p,C}$ de C_p y posteriormente manda una petición DE a todos sus vecinos excepto a C_p.

Algoritmo de Propagación para Árboles de Cliques

1. Moralizar la red bayesiana.
2. Seleccionar un orden de eliminación de los nodos del grafo moral.
3. Determinar los cliques obtenidos en el proceso de triangulación.
4. Construir un árbol de cliques.

5. Llamar a **IntegrarEvidencia**
6. Elegir un universo C como universo pivote.
7. Llamar a **RecojerEvidencia**(C).
8. Llamar a **DistribuirEvidencia**(C).
9. Llamar a **Absorción** para cada uno de los cliques..

Referencias

1. Andersen, S.K., Olesen, K.G. Jensen, F.V., Jensen, F. Hugin: a shell for building belief universes for expert systems. *11th International Joint Conference on Artificial Intelligence, Detroit.*, 1989.
2. Beeri, C., Fagin, R., Maier, D., Yannakakis,M. On the desirability of acyclic database schemas. *Journal of the Assocaition for Computing Machinery*, 30(3):479–513, 1983.
3. Cannings, C., Thompson, E.A., Skolnick, M.H. Recursive derivation of likelihoods on pedigrees. *Adv. Appl. Probabil.*, 8:622–625, 1976.
4. Cannings, C., Thompson, E.A., Skolnick, M.H. Probabilistic functions on complex pedigrees. *Adv. Appl. Probabil.*, 10:26–61, 1978.
5. Cano, A, Moral, S. Heuristic algorithms for the triangulation of graphs. *Advances in Intelligent Computing.*, pág. 166–171, 1995.
6. Cano, J.E. *Propagación de probabilidades inferiores y superiores en grafos.* PhD thesis, Dpto. de C.C. e I.A. Facultad de Ciencias. Universidad de Granada, 1993.
7. Cano J.E., Delgado, M., Moral, S. An axiomatic framework for the propagation of uncertainty in directed acyclic graphs. *International Journal of Approximate reasoning*, 8:253–280, 1993.
8. Cooper, G.F. Probabilistic inference using belief netwoks is np-hard. Technical Report KSL-87-27, Knowledge systems laboratory, Stanford University.California 94305-5479, Julio 1987.
9. Cooper, G.F. Bayesian belief-network inference using recursive decomposition. Technical Report KSL-90-05, Knowledge systems laboratory, Stanford University.California., 1990.
10. D'Ambrosio, B. Symbolic porbabilistic inference in belief nets. Technical report, Oregon State University, 1989.
11. Dawid, A.P., Kjærulff U., Lauritzen, S.L. Hybrid propagation in junction trees. Technical Report R-93-2028, Institute for Electronic Systems, Institute for Electronic Systems, Aalborg University, September 1993.
12. Díez, F.J. Local conditioning in bayesian networks. *Articial Intelligence*, 87:1–20, 1996.
13. Fujisawa, T., Orino, H. An efficient algorithm of finding a minimal triangulation of a graph. *IEEE International Symposium on Circuits and Systems*, pág. 172–175, 1974.
14. Good, I.J. A causal calculus. *Philosophy of Science*, 11:305–318, 1961.
15. Hernández, L.D., Bolaños, M.J. Applicación de algoritmos evolutivos para el problema de la triangulación en redes causales. *Tecnologías y Lógica Fuzzy*, pág. 127–132, 1994.
16. Hernández, L.D., Moral, S. Mixing exact and importance sampling propagation algorithms in dependence graphs. *International Journal of Intelligent Systems*, 12:553–576, 1997.
17. Jensen, F. Implementation aspects of vaious propagation algorithms in hugin. Technical Report R 94-2014, Department of Mathematics and Computer Science, Institute for Electronic Systems, Aalborg University, March 1994.

18. Jensen, F.V. *An introduction to Bayesian networks.* SpringerVerlag NewYork Inc. k, 1996.
19. Jensen, F.V., Lauritzen, S.L., Olensen, K.G. Bayesian updating in causal probabilistic networks by local computations. *Computational Statistics Quarterly*, pág. 269–282, 1990.
20. Jensen, F.V., Olesen, K.G., Andersen, S.K. An algebra of bayesian belief universes for knowledge based sustems. *Networks*, 20:637–659, 1990.
21. Kim, J.H., Pearl, J. A computational model for causal and diagnostic reasoning in inference engines. *8th. International Joint Conference on Artificial Intelligence, Karlsruhe, West Germany*, 1983.
22. Kjærulff, U. Triangulation of graphs-algorithms giving total state space. Technical Report R 90-09, Department of Mathematics and Computer Science, Institute for electronic Systems, Aalborg University, 1990.
23. Kjærulff, U. Optimal descompotition of probabilistic networks by simulated annealing. *Statistics and Computing*, 2:1–21, 1992.
24. Kjærulff, U. *Aspects of Efficiency Improvement in Bayesian Networks.* PhD thesis, Department of Mathematics and Computer Science. Institute of Electronic Systems. Aalborg University, 1993.
25. Larrañaga, P., Kuijpers. C.M.H., Poza, M., Murga, R.H. Optimal decomposition of bayesian networks by genetic algorithms. Report EHU-KZAA-IKT-3-94, Konputazio Zientziak eta Adimen Artifiziala Saila, Informatika Fakultatea. Euskal Herriko Univertsitatea, Noviembre 1994.
26. Lauritzen, S.L. Dawid, A.P., Larsen, B.N., Leimer, H.G. Independence properties of directed markov fields. Reserch R 88-32, Institute for Electronic Systems, Aalborg University, Denmark, 1988. (con discusión).
27. Li and D'Ambrosio. Efficient inference in bayes nets as a combinatorial optimization problem. *Intl Jrnl of Approximate Reasoning*, 11(1):55–81, 1994.
28. Neapolitan, R.E. *Probabilistic Reasoning in Expert Systems: Theory and Algorithms.* Wiley-Interscience, Fohn Wiley & Sons, Inc., 1990.
29. Pearl, J. A constriant-propagation approach to probabilistic reasoning. In L.N. Kanal and J.F. Lemmer, editor, *Uncertainty in Artificial Intelligence (pp 357-370).* Amsterdan: North Holland, 1986.
30. Pearl, J. Fusion, propagation and structuring in belief networks. *Artificial Intelligence*, 29(3):241–288, 1986.
31. Pearl, J. Distributed revision of composite beliefs. *Artificial Intelligence*, 33:137–215, 1987.
32. Pearl, J. Probabilistic reasoning in intelligence systems. *San MAteo, C.A.:Morgan Kaufman*, 1988.
33. Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, Inc., 1988.
34. Rose, D.J., Tarjan, R.E., Lueker, G.S. Algorithmic aspects of vertex elimination on graphs. *SIAM Journal on Computing*, 5:266–283, 1976.
35. Shachter, R.D. Probabilistic inference and influence diagrams. *Operations Research*, 36(4):589–604, July-August 1988.

36. Shachter, R.D. Evidence absorption and propagation through evidence reversals. In *Fifth Workshop on Uncertainty in Artificial Intelligence*, pág. 303–310. University of Windsor, Ontario, 1990.
37. Shachter, R.D., Andersen, S.K., Poh, K.L. Directed reduction algorithms and decomposable graphs. In *Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence*, pág. 237–244, Cambridge, MA, July 27-29 1990.
38. Shachter, R.D., Andersen, S.K., Szlovits, P. The equivalence of exact methods for probabilistic inference on belief network. *Submitted to Artificial Intelligence*, 1991.
39. Shafer, G., Shenoy, P.P. Local computation in hypertrees. *Working paper N.201. School of business, University of Kansas*, 1988.
40. Shafer, G., Shenoy, P.P. Probability propagation. *Annals of Mathematics and Artificial Intelligence*, 2:327–351, 1990.
41. Shenoy, P.P., Shafer, G. Axioms for probability and belief-functions propagation. In R.D.Shachter, T.S. Levitt, L.N.Kanal, J.F.Lemmer, editor, *Uncertainty in artificial intelligence*, number 4, pág. 169–198. Elsevier science publisher B.V. (North-Holland), 1990.
42. Suermondt, H.J., Cooper G.F. Probabilistic inference in multiply connected belief netwoks using loop cutset. *International Journal of approximate reasoning*, 4:283–306, 1990.
43. Suermondt, H.J., Cooper, G.F. Initialization for the method of conditioning in bayesian belief networks. *Artificial Intelligence*, 50:83–94, 1991.
44. Tarjan, R.E., Yannakakis, M. Simple lienar-time algorithms to test chosrdality of graphs,acyclicity of hypergrphs, and selectively reduce acyclic hypergraphs. *SIAM Journal on Computing*, 13(3):566–579, 1984.
45. Wen, W.X. Optimal decomposition of belief networks. *Proceedings of the Sixth Workshop on Uncertainty in Artificial Intelligence*, (Cambridge, MA):245–256, 1990.

Algoritmos de Propagación II. Métodos de Monte Carlo

Antonio Salmerón

Dpto. Estadística y Matemática Aplicada
Universidad de Almería
Almería. 04120
correo-e: asc@stat.ualm.es

Resumen

Es conocido que la propagación exacta de probabilidades en redes bayesianas es un problema NP-duro [6]. Esto quiere decir que si la red es suficientemente complicada, puede que no podamos obtener resultados en un tiempo razonable. Surge entonces la necesidad de emplear métodos aproximados que, a cambio de perder la exactitud de los cálculos, ofrecen resultados en un tiempo menor. En este capítulo estudiamos un grupo de algoritmos aproximados de gran importancia: los basados en métodos de Monte Carlo.

1 Introducción

Los algoritmos aproximados surgieron con el propósito de resolver los casos peores para los métodos exactos en un tiempo más razonable, generalmente mediante técnicas de Monte Carlo, a cambio de la pérdida de la exactitud de los cálculos. La inferencia por métodos de simulación es también un problema NP-duro cuando se requiere una precisión determinada [7]; sin embargo, el conjunto de problemas resolubles es mayor que para los métodos exactos.

En este capítulo describiremos los métodos más importantes de propagación de probabilidades basados en simulación por Monte Carlo.

Comenzaremos planteando el problema en la sección 2. A continuación, en la sección 3, explicaremos el concepto de simulación y veremos cómo se aplica a la estimación de la distribución *a posteriori* de una red bayesiana. En la sección 4 estudiamos el funcionamiento de los métodos de propagación por Monte Carlo más sencillos: los que no utilizan precomputación. Terminaremos el capítulo con un acercamiento a métodos más sofisticados como el muestreo sistemático (sección 5) y muestreo por importancia basado en precomputación aproximada (sección 6).

2 Planteamiento del Problema

Supondremos durante este capítulo una red bayesiana definida sobre un conjunto de variables $X = \{X_1, \ldots, X_n\}$, cada una de ellas tomando valores en un conjunto finito U_i, $i = 1, \ldots, n$ y $N = \{1, \ldots, n\}$. Consideraremos también un conjunto de variables observadas X_E, tomando el valor $X_E = e$ con $e \in U_E$. Al valor e lo llamaremos *evidencia.*

El objetivo que nos proponemos es calcular la distribución *a posteriori* $p(x_k|e)$ para todo $x \in U_k$, correspondiente a cada variable X_k con $k \in N$. Al cálculo de esta probabilidad lo llamamos *propagación de probabilidades.* Esta probabilidad podría obtenerse mediante marginalización a partir de la distribución conjunta de la red,

$$p(x) = \prod_{i \in N} f_i(x^{\downarrow s(f_i)}), \quad \forall x \in U_N, \tag{1}$$

donde $s(f_i)$ representa el conjunto de índices de las variables para las que está definida la función f_i. En este caso, cada función f_i se corresponde con la distribución condicionada de la variable X_i a sus padres Π_{X_i}, es decir, $p(x_i|\pi_{X_i})$, con $x_i \in U_i$, $\pi_{X_i} \in U_{F(i)}$ y $s(f_i) = \{i\} \cup F(i)$, donde $F(i)$ es el conjunto de índices de las variables padre de X_i. Si existen variables observadas, $X_E = e$, entonces la distribución anterior quedará como

$$p(x, e) = \left(\prod_{i \in N} f_i(x^{\downarrow s(f_i)}) \right) \cdot \left(\prod_{j \in E} \delta_{e_j}(x^{\downarrow j}) \right), \quad \forall x \in U_N, \tag{2}$$

donde δ_{e_j} es una función que toma el valor 1 si x es consistente con la evidencia y 0 en otro caso:

$$\delta_{e_j}(y) = \begin{cases} 1 \text{ si } e_j = y, \\ 0 \text{ si } e_j \neq y. \end{cases} \tag{3}$$

Obsérvese que la probabilidad que queremos calcular es

$$p(x_k|e) = \frac{p(x_k, e)}{p(e)}, \tag{4}$$

y, dado que $p(e)$ es constante, ésta es proporcional a $p(x_k, e)$. Por lo tanto, podemos obtener la distribución *a posteriori* si calculamos para cada $x_k \in U_k$ el valor

$p(x_k, e)$ y normalizamos después. Podemos expresar $p(x_k, e)$ como la siguiente suma:

$$S = p(x_k, e) = \sum_{\substack{x \in U_N \\ x^{\downarrow E} = e \\ x^{\downarrow k} = x_k}} p(x) = \sum_{\substack{x \in U_N \\ x^{\downarrow k} = x_k}} p(x, e). \tag{5}$$

Pero suponemos que la distribución $p(x, e)$ es suficientemente complicada como para que los métodos exactos no sean aplicables, y, de igual manera, tampoco será posible calcular la suma anterior en un tiempo razonable. Por lo tanto, nos conformaremos con aplicar un método de simulación para obtener una estimación de la probabilidad que buscamos.

A continuación veremos en qué consiste la simulación y cómo puede ésta aplicarse a nuestro problema.

3 Simulación

Por *simulación* podemos entender la experimentación sobre un modelo de cierto sistema, de cara a predecir el comportamiento del mismo. Si el proceso de simulación conlleva el uso de números aleatorios, se la suele llamar también *simulación por Monte Carlo*. El objetivo de la simulación es extraer conclusiones sobre cierto sistema real sin necesidad de experimentar directamente sobre el sistema en cuestión.

Por ejemplo, supongamos que una empresa está considerando la apertura de un supermercado y nos encarga un informe para decidir el número de cajas registradoras que han de colocar. En este caso, el sistema real es el supermercado. Para decidir el número óptimo de cajas registradoras, podríamos observar el comportamiento del sistema, construyendo el supermercado, poniendo un cierto número de cajas y observando si éstas son suficientes o no. Es evidente que este método es extremadamente costoso. Podríamos recurrir entonces a realizar un modelo de simulación del supermercado y experimentar, en un ordenador, el funcionamiento del mismo. En este caso, sería sencillo hacer pruebas con distintos números de cajas registradoras.

En un modelo de este tipo, necesitamos generar aleatoriamente una población; en este caso, la de los usuarios del supermercado. Se sabe que dicha población puede modelizarse de acuerdo a ciertas distribuciones de probabilidad conocidas: por ejemplo, el número de personas que llegan a una caja registradora para ser atendidos sigue una distribución de Poisson.

Generar individuos de una población no es más que generar valores para una variable aleatoria que sigue una distribución dada. Una forma de hacer esto es mediante el *método de inversión*, fundamentado en el siguiente teorema:

Teorema 1. Sea X una variable aleatoria con función de distribución $F(x)$. Sea $F^{-1}(y)$ la función inversa de F, definida como

$$F^{-1}(y) = \inf\{x \mid F(x) \geq y\}, \quad 0 \leq y \leq 1. \tag{6}$$

Entonces, si U es una v.a. uniformemente distribuida en el intervalo $(0, 1)$, se cumple que la v.a. definida como $Z = F^{-1}(U)$ tiene como función de distribución $F(x)$. □

Este teorema nos dice la forma de generar valores para la variable X. Lo único que hay que hacer es generar un número aleatorio u (entre 0 y 1), y calcular el valor $F^{-1}(u)$. El resultado será un valor para la variable X. Existen numerosas formas de generar números aleatorios [16]. La mayoría de los lenguajes de programación de propósito general ofrecen mecanismos para generarlos. Con esto, el algoritmo para realizar esta tarea es como sigue:

1. Generar un número aleatorio u.
2. $X = F^{-1}(u)$.
3. Devolver X.

El método anterior es válido para variables tanto discretas como continuas. En las redes bayesianas, las variables que manejaremos serán siempre ***discretas y finitas***, es decir, solo podrán tomar un número finito de valores. El siguiente ejemplo ilustra el funcionamiento del método de inversión para una variable discreta y finita.

Ejemplo 1. Sea una variable aleatoria X que puede tomar los valores x_1, x_2 y x_3 con probabilidad $P(X = x_1) = 0.2$, $P(X = x_2) = 0.3$ y $P(X = x_3) = 0.5$. La función de distribución F para la variable X puede verse en la figura 1. Supongamos que hemos generado un número aleatorio $u = 0.7$. Para obtener un valor para X a partir de u hemos de evaluar la función $F^{-1}(0.7)$. Obsérvese que en la gráfica 1 esto se puede hacer situando el punto 0.7 en el eje de ordenadas y viendo con qué punto del eje de abscisas se corresponde de acuerdo con el dibujo de F. Puede comprobarse que el valor 0.7 se corresponde con el valor x_3 de acuerdo con la fórmula (6). □

En general, un algoritmo para generar valores para una variable X con n posibles valores, $\{x_1, \ldots, x_n\}$ y con función de probabilidad $P(X = x_1) = p_1$, $P(X = x_2) = p_2, \ldots, P(X = x_n) = p_n$, es el siguiente:

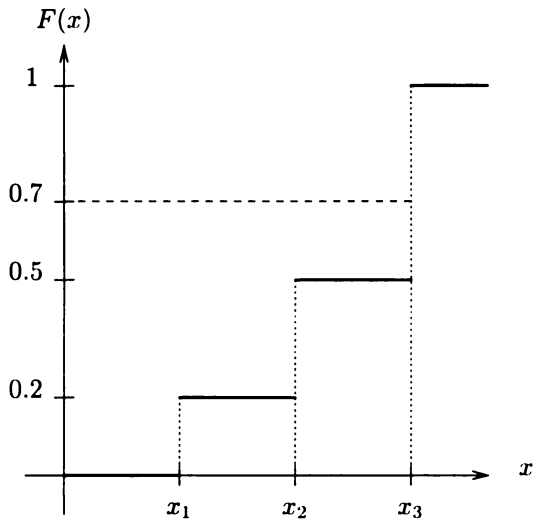


Figura 1. Método de inversión.

1. Generar un número aleatorio u.
2. $P = p_1$.
3. $i = 1$.
4. Mientras $i <= n$ y $P < u$,
   (a) $i = i + 1$.
   (b) $P = P + p_i$.
5. $X = x_i$.
6. Devolver X.

3.1 Obtención de la probabilidad a posteriori mediante simulación

Una forma de obtener una estimación de la probabilidad de interés (fórmula (5)) mediante simulación, sería generando una serie de valores para las variables $X_1, \ldots, X_n$ mediante el método de inversión a partir de la distribución $p(x)$. A partir de la muestra generada, para un cierto x_k podríamos estimar su probabilidad como el cociente entre el número de veces en que X_k toma el valor x_k y el número total de individuos en la muestra generada.

Ejemplo 2. Consideremos la red de la figura 2, para la cual hay definida una distribución de probabilidad $p(x_1, x_2, x_3)$. Supongamos que las tres variables son

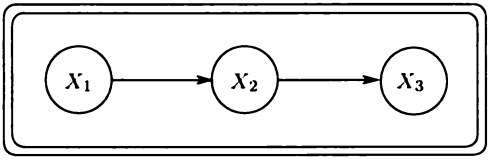


Figura 2. Una red bayesiana con tres variables.

$f_1(0) = P(X_1 = 0) = 0.6$
$f_1(1) = P(X_1 = 1) = 0.4$
$f_2(0,0) = P(X_2 = 0 \mid X_1 = 0) = 0.2$
$f_2(0,1) = P(X_2 = 0 \mid X_1 = 1) = 0.5$
$f_2(1,0) = P(X_2 = 1 \mid X_1 = 0) = 0.8$
$f_2(1,1) = P(X_2 = 1 \mid X_1 = 1) = 0.5$
$f_3(0,0) = P(X_3 = 0 \mid X_2 = 0) = 0.2$
$f_3(0,1) = P(X_3 = 0 \mid X_2 = 1) = 0.3$
$f_3(1,0) = P(X_3 = 1 \mid X_2 = 0) = 0.8$
$f_3(1,1) = P(X_3 = 1 \mid X_2 = 1) = 0.7$

Tabla 1. Probabilidades condicionadas para la red anterior.

binarias, es decir, pueden tomar los valores 0 ó 1, y que hemos generado la siguiente muestra a partir de la distribución p mediante el método de inversión:

$$(0,1,0),(0,1,1),(0,1,0),(1,1,1),(0,0,1),(1,1,1),$$

donde cada coordenada de cada tripleta representa los valores para X_1, X_2 y X_3 respectivamente. Si, por ejemplo, quisiéramos estimar la probabilidad *a posteriori* de la variable X_1, contaríamos los elementos de la muestra en los que X_1 toma el valor 0 y dividiríamos ese número entre el tamaño de la muestra, y análogamente para $X_1 = 1$. Es decir, estimaríamos dicha probabilidad como:

$$\hat{P}(X_1 = 0) = \frac{4}{6} = \frac{2}{3},$$

$$\hat{P}(X_1 = 1) = \frac{2}{6} = \frac{1}{3}.$$

□

x	$p(x)$
$(0,0,0)$	0.024
$(0,0,1)$	0.096
$(0,1,0)$	0.144
$(0,1,1)$	0.336
$(1,0,0)$	0.040
$(1,0,1)$	0.160
$(1,1,0)$	0.060
$(1,1,1)$	0.140

Tabla 2. Probabilidad conjunta para la red anterior.

En la práctica, no será posible utilizar la distribución p para generar la muestra, pues ésta será difícil de manejar y por lo tanto su inversa también lo será. Lo que se hace es utilizar una distribución modificada más sencilla para simular, y luego se asigna un *peso* o importancia a cada individuo de la muestra. El fundamento de este procedimiento consiste en que podemos expresar la suma (5) como sigue:

$$p(x_k, e) = \sum_{\substack{x \in U_N \\ x^{\downarrow k} = x_k}} p(x, e) = \sum_{\substack{x \in U_N \\ x^{\downarrow k} = x_k}} \frac{p(x, e)}{f^*(x)} f^*(x), \tag{7}$$

donde f^* es una función masa de probabilidad positiva en todos los puntos donde p es positiva. A f^* se le llama ***función de muestreo.***

Si f^* se elige de forma que sea sencilla de manejar, podemos utilizarla para generar la muestra de las variables de la red, $\{x^{(j)}\}, j = 1, \ldots, m$, con m el tamaño de la muestra. A cada configuración $x^{(j)}$ le asignamos un *peso* o ***importancia*** w_j definido como

$$w_j = \frac{p(x^{(j)}, e)}{f^*(x^{(j)})}. \tag{8}$$

Entonces, puede estimarse cada probabilidad $p(x_k, e)$ como

$$\hat{p}(x_k, e) = \frac{1}{m} \sum_{j \in J} \frac{p(x^{(j)}, e)}{f^*(x^{(j)})} = \frac{1}{m} \sum_{j \in J} w_j, \tag{9}$$

donde $J \subseteq \{1, \ldots, m\}$ es un conjunto de índices tal que las configuraciones $x^{(j)}$, $j \in J$, verifican que ${x^{(j)}}^{\downarrow k} = x_k$ y ${x^{(j)}}^{\downarrow E} = e$. Es decir, se estima la probabilidad de cada valor x_k como la media de los pesos de las configuraciones que componen la muestra, considerando que tienen peso cero aquellas configuraciones que no son consistentes con x_k. Puede comprobarse que $\hat{p}(x_k, e)$ es un estimador insesgado de $p(x_k, e)$ (ver [17]).

Para obtener la probabilidad a posteriori, $p(x_k|e)$, basta con normalizar los valores estimados de $p(x_k, e)$, lo que es equivalente a dividir entre la suma de todos los pesos.

Configuración $(x^{(j)})$	Peso (w_j)
$(0, 0, 0)$	0.192
$(0, 1, 1)$	2.688
$(0, 1, 0)$	1.152
$(1, 1, 1)$	1.120
$(0, 0, 1)$	0.768
$(1, 1, 1)$	1.120

Tabla 3. Pesos para la muestra del ejemplo.

Ejemplo 3. Supongamos que queremos estimar la probabilidad *a posteriori* de la variable X_1 de la red de la figura 2. Vamos a utilizar el método de los pesos. Imaginemos que hemos utilizado para obtener la muestra una distribución uniforme, es decir, $f^*(x) = 1/8$ para toda configuración x de las variables X_1, X_2 y X_3, y que hemos obtenido la misma muestra que en el ejemplo 2. La tabla 3 muestra los pesos de cada una de las configuraciones que forman la muestra. Procedemos como en el ejemplo 2, pero ahora sumando los pesos de las configuraciones favorables a cada uno de los valores de X_1. Obtenemos la siguiente estimación:

$$\hat{P}(X_1 = 0) = \frac{0.192 + 2.688 + 1.152 + 0.768}{6} = \frac{4.8}{6} = 0.8,$$
$$\hat{P}(X_1 = 1) = \frac{1.120 + 1.120}{6} = \frac{2.240}{6} \approx 0.37.$$

Normalizando, obtenemos la estimación $\hat{P}(X_1 = 0) = 0.68$ y $\hat{P}(X_1 = 1) = 0.32$. □

El proceso anterior queda reflejado en el siguiente algoritmo:

Algoritmo de simulación

1. Desde $i = 1$ hasta m,
   (a) Generar una configuración $x^{(i)}$ a partir de f^*.
   (b) Calcular

$$w_i = \frac{p(x^{(i)}, e)}{f^*(x^{(i)})}. \tag{10}$$

2. Para cada $x_k \in U_k$, $k = \{1, \ldots, n\}$,
   (a) Estimar $p(x_k, e)$ usando la fórmula (9).
3. Normalizar los valores $p(x_k, e)$ para obtener $p(x_k|e)$.

En este esquema, si todas las configuraciones que forman la muestra se elijen de forma independiente, diremos que realizamos un *muestreo por importancia* [16].

Siguiendo este esquema general de simulación, se han desarrollado diversos esquemas de propagación aproximada. La diferencia entre ellos radica en la forma en que se generan las configuraciones que componen la muestra y también en la distribución de muestreo que se emplea. Estudiaremos los siguientes métodos:

- Muestreo lógico probabilístico.
- Ponderación por verosimilitud.
- Simulación estocástica.
- Muestreo estratificado o sistemático.
- Muestreo por importancia basado en precomputación aproximada.

Los tres primeros no requieren ningún proceso de precomputación para calcular las distribuciones de muestreo antes de la simulación; por ello, los llamaremos *algoritmos de Monte Carlo sin precomputación*. El muestreo sistemático tampoco requiere de dicha precomputación, pero difiere de los anteriores en la forma de obtener las muestras. Por último, el método de muestreo por importancia basado en precomputación aproximada conlleva un cálculo inicial enfocado a mejorar la calidad de las funciones de muestreo.

4 Algoritmos de Propagación por Monte Carlo sin Precomputación

4.1 Muestreo lógico probabilístico

Este método, propuesto por Henrion [10], se engloba dentro de los llamados de *propagación hacia delante*. La idea de las técnicas de propagación hacia delante

consiste en elegir un orden *ancestral*[1] de las variables de la red y obtener una configuración para cada variable en secuencia, muestreando según la distribución condicionada de dicha variable dados sus padres en la red. A cada configuración de las variables obtenida se le asigna un peso que, al final del proceso de simulación, y normalizando, resulta en una estimación de la probabilidad a posteriori de cada variable.

En el método de muestreo lógico probabilístico destaca el hecho de que todos los pesos valen 0 ó 1, dependiendo de que la configuración obtenida sea coherente con las observaciones o no. Esto se debe a que la distribución de muestreo elegida coincide con la original, es decir, que para cada configuración $x^{(j)}$, el peso es:

$$\begin{aligned} w_j &= \frac{p(x^{(j)}, e)}{f^*(x^{(j)})} \\ &= \frac{\left(\prod_{i=1}^{n} f_i(x^{(j)\downarrow s(f_i)})\right) \cdot \left(\prod_{l \in E} \delta_{e_l}(x^{(j)\downarrow l})\right)}{\prod_{i=1}^{n} f_i(x^{(j)\downarrow s(f_i)})} \\ &= \prod_{l \in E} \delta_{e_l}(x^{(j)\downarrow l}). \end{aligned}$$

El algoritmo detallado es el siguiente, donde supondremos, sin pérdida de generalidad, que las variables siguen un orden ancestral:

Muestreo Lógico

1. Desde $j = 1$ hasta m (tamaño de la muestra),
   (a) Desde $i = 1$ hasta n,
      i. Obtener un valor $x_i \in U_i$ simulando de acuerdo a la distribución $p(x_i|\pi_{X_i})$, donde π_{X_i} es la configuración ya obtenida para los padres de X_i.
      ii. Si X_i es una variable observada y $x_i \neq e_i$, hacer $w_j = 0$ y volver al paso 1.
   (b) Hacer $w_j = 1$.
2. Para cada $x_k \in U_k$, $k = \{1, \ldots, n\}$,
   (a) Estimar $p(x_k, e)$ usando la fórmula (9).
3. Normalizar los valores $p(x_k, e)$ para obtener $p(x_k|e)$.

[1] Un orden de los nodos de un grafo se dice *ancestral* si cada nodo tiene una posición en dicho orden anterior a cualquier descendiente suyo.

Obsérvese que el problema de este algoritmo es que si la configuración obtenida no concuerda con las observaciones, la iteración no será válida (paso 1.(a).ii. del algoritmo). Este problema no se presenta si todas las observaciones se dan en nodos raíz, dado que en ese caso se puede instanciar cada variable al valor observado y no se simulan. Entonces, la primera variable a simular sería la primera que no estuviera observada, y su distribución de probabilidad estaría restringida a los valores de las variables observadas, luego no se obtendrían configuraciones contradictorias con las observaciones. De cualquier forma, lo normal es que las observaciones se presenten en cualquier parte de la red y no sólo en las raíces, por lo que este método no será aplicable en numerosas ocasiones.

El siguiente ejemplo ilustra el funcionamiento del algoritmo.

Ejemplo 4. Consideremos la red de la figura 2, en la que se ha observado que la variable X_3 toma el valor 1. El orden en que vamos a simular las variables es X_1, X_2, X_3. Veamos:

- **Simulación de** X_1. Para simular un valor para esta variable, generamos un número aleatorio. Supongamos que dicho número es $u = 0.3$. Aplicando el método de inversión a la distribución f_1 (ver tabla 1), obtenemos el valor $X_1 = 0$.
- **Simulación de** X_2. Generamos un nuevo número aleatorio, por ejemplo, $u = 0.7$. Ahora utilizamos la distribución f_2 instanciada al valor $X_1 = 0$ y por el método de inversión obtenemos el valor $X_2 = 1$.
- **Simulación de** X_3. Realizamos el mismo proceso utilizando f_3. Si el número aleatorio generado es $u = 0.4$, obtenemos $X_3 = 1$.

En definitiva, la configuración obtenida es $(0, 1, 1)$, que es consistente con la observación $X_3 = 1$. Si en la simulación de X_3 el número aleatorio hubiera sido, por ejemplo, $u = 0.1$, entonces el valor obtenido para X_3 hubiera sido el 0, lo que produciría la configuración $(0, 1, 0)$ que no es consistente con la evidencia, y, por lo tanto, la simulación no habría sido válida.

□

4.2 Método de ponderación por verosimilitud

El esquema de ponderación por verosimilitud fue desarrollado independientemente por Fung y Chang [9] y Shachter y Peot [18]. El objetivo que persigue es evitar la aparición de configuraciones inconsistentes con la evidencia. Para ello, las variables observadas no se simulan, sino que toman directamente el valor observado. Esto se consigue haciendo que la distribución de muestreo valga 1 para el valor de las variables observadas, de forma que siempre se obtenga ese valor al simular.

Es decir, la función de muestreo será igual al producto de las condicionadas de la red salvo para las variables observadas:

$$\forall x_i \in U_i, \quad f_i^*(x_i) = \begin{cases} p(x_i|\pi_{X_i}) \text{ si } i \notin E, \\ \delta_{e_i}(x_i) \quad \text{si } i \in E. \end{cases} \tag{11}$$

donde f_i^* es la distribución de muestreo para la variable X_i, y π_{X_i} es el valor simulado para las variables Π_{X_i}, con lo que

$$f^*(x) = \prod_{i=1}^{n} f_i^*(x_i) \quad \forall x = (x_i, \ldots, x_n) \in U_N. \tag{12}$$

Obsérvese que al usar las distribuciones condicionadas para simular, es necesario que el orden de simulación de las variables sea ancestral, al igual que en el muestreo lógico probabilístico.

Dado que todas las configuraciones son consistentes con la evidencia, el peso de una configuración $x = (x_1, \ldots, x_n)$ se puede calcular como

$$\begin{aligned} w &= \frac{p(x,e)}{f^*(x)} \\ &= \frac{\left(\prod_{i=1}^{n} f_i(x^{\downarrow s(f_i)})\right) \cdot \left(\prod_{i\in E} \delta_{e_i}(x_i)\right)}{\left(\prod_{i\notin E} f_i(x^{\downarrow s(f_i)})\right) \cdot \left(\prod_{i\in E} \delta_{e_i}(x_i)\right)} \\ &= \prod_{i\in E} f_i(x^{\downarrow s(f_i)}) \\ &= \prod_{i\in E} p(x_i|\pi_{X_i}). \end{aligned}$$

Es decir, el peso de cada configuración viene determinado por la probabilidad de la evidencia dado el resto de las variables, o, lo que es lo mismo, la verosimilitud de la evidencia.

Con esto, el algoritmo de ponderación por verosimilitud es muy similar al de Henrion, y puede enunciarse como sigue:

Ponderación por verosimilitud

1. Desde $j = 1$ hasta m (tamaño de la muestra),
   (a) Desde $i = 1$ hasta n,

   i. Si $i \notin E$, obtener un valor $x_i \in U_i$ simulando de acuerdo a la distribución $p(x_i|\pi_{X_i})$.
   (b) $w_j = \prod_{i \in E} p(x_i|\pi_{X_i})$.
2. Para cada $x_k \in U_k$, $k = \{1, \ldots, n\}$,
   (a) Estimar $p(x_k, e)$ usando la fórmula (9).
3. Normalizar los valores $p(x_k, e)$ para obtener $p(x_k|e)$.

Ejemplo 5. Para ilustrar este método, consideraremos de nuevo la red de la figura 2 y el orden de simulación X_1, X_2, X_3. Supondremos que se ha observado que la variable X_3 toma el valor 1. En estas condiciones, el proceso de simulación sería prácticamente igual que en el ejemplo 4, salvo que la variable X_3 no se simularía, sino que directamente tomaría el valor 1. Luego, si los números aleatorios son los mismos que en el ejemplo 4, la configuración obtenida es $(0, 1, 1)$, y el peso será

$$w = P(X_3 = 1|X_1 = 0, X_2 = 1) = P(X_3 = 1|X_2 = 1) = 0.7.$$

□

El funcionamiento de este método es bueno salvo cuando se presentan probabilidades muy próximas a cero. En este caso es posible que gran parte de las configuraciones simuladas tengan peso nulo [12].

4.3 Método de simulación estocástica

Este método, también llamado de *simulación directa*, fue propuesto por Pearl [15]. Las diferencias más destacadas respecto al algoritmo de ponderación por verosimilitud son:

1. En este caso, las variables no han de simularse en ningún orden en especial.
2. En lugar de simular usando la distribución condicionada de cada variable, se usa la distribución de cada variable condicionada a su envolvente de Markov en la red[2].

El algoritmo detallado queda como sigue.

Simulación estocástica

1. Hacer que todos los nodos de la red a uno de sus posibles valores con probabilidad no nula.

[2] La *envolvente de Markov* de una variable en una red bayesiana es el conjunto de los padres, hijos y padres de los hijos de dicha variable.

2. Para cada variable no observada X_i, $i \in \{1, \ldots, n\}$, hacer $h_i(x_i) = 0$ para todo $x_i \in U_i$.
3. Desde $j = 1$ hasta m (tamaño de la muestra),
   (a) Para cada variable X_i, $i \in \{1, \ldots, n\}$,
      i. Calcular $P(X_i|W_{X_i})$, donde W_{X_i} denota la envolvente de Markov de la variable X_i, de la siguiente manera:

$$p(x_i|w_{X_i}) = \alpha \cdot p(x_i|\pi_{X_i}) \prod_{j \in H(i)} p(x_j|\pi_{X_j}) \quad \forall x_i \in U_i. \tag{13}$$

      donde α es una constante de normalización, $H(i)$ es el conjunto de índices de las variables hijo de X_i y w_{X_i} es la configuración actual de la envolvente de Markov de la variable X_i.
      ii. Simular un valor $x_i^{(j)} \in U_i$ para X_i según la distribución $p(x_i|w_{X_i})$.
      iii. Actualizar h_i según una de la dos siguientes expresiones:

$$h_i(x_i^{(j)}) = h(x_i^{(j)}) + 1,$$
$$h_i(x_i^{(j)}) = h(x_i^{(j)}) + p(x_i^{(j)}|w_{X_i}).$$

4. Normalizar los h_i, $i = 1, \ldots, n$. Cada h_i resultante es la distribución a posteriori de la variable X_i.

Este método presenta dos problemas principales. Por un lado, puede ser difícil encontrar una configuración inicial para las variables de la red que tenga probabilidad positiva. Jensen, Kong y Kjærulff [13] proponen usar inicialmente una técnica de muestreo hacia delante para encontrar la configuración inicial.

Por otro lado, cada configuración depende de la generada inmediatamente antes (ver fórmula (13)). Por eso, puede darse el caso de que, una vez alcanzada una configuración, ésta se repita un gran número de veces, debido a que las dependencias entre las variables sean "casi" funcionales, es decir, las distribuciones generadas en la fórmula 13 tengan valores muy próximos a 0 o a 1. La convergencia de este método hacia la distribución exacta está asegurada, cuando todas las probabilidades son estrictamente positivas, por resultados de la teoría de los procesos de Markov [3,8], pero ésta puede alcanzarse muy lentamente por la razón dicha anteriormente. En el caso de tener probabilidades nulas, puede que no se de la convergencia. El siguiente ejemplo puede aclarar la situación:

Ejemplo 6. Sea una red bayesiana con dos variables binarias conectadas de la forma $X_1 \rightarrow X_2$, con $U_1 = \{x_1, \bar{x}_1\}$, $U_2 = \{x_2, \bar{x}_2\}$ y tales que $p(x_2|x_1) = p(\bar{x}_2|\bar{x}_1) = \delta \simeq 1$. Supongamos que $p(x_1) = 0.5$ y que $X_1 = x_1$, entonces

$p(x_2|w_{X_2}) = p(x_2|x_1) = \delta$. Si en una simulación obtenemos $X_2 = x_2$, en la próxima simulación la distribución usada para simular X_1 será

$$\begin{aligned} p(x_1|w_{X_1}) &= p(x_1|x_2) \\ &= \alpha \cdot p(x_2|x_1) \cdot p(x_1) \\ &= \alpha \cdot 0.5 \cdot \delta = \delta, \end{aligned}$$

dado que, por la regla de Bayes, $\delta = 1/P(X_2 = x_2)$, y

$$\begin{aligned} p(x_2) &= p(x_2|x_1) \cdot p(x_1) + p(x_2|\bar{x}_1) \cdot p(\bar{x}_1) \\ &= \delta \cdot 0.5 + (1 - \delta) \cdot 0.5 = 0.5. \end{aligned}$$

Si continuamos así, obtendremos la configuración (x_1, x_2) con probabilidad muy próxima a 1, y, en el momento en que una de las dos variables cambiara de valor, la otra también lo haría, repitiéndose entonces muchas veces la configuración $(\bar{x}_1, \bar{x}_2)$. Obsérvese, por lo tanto, que la configuración que se obtenga en una simulación puede depender fuertemente de la obtenida en la simulación anterior. □

Tratando de resolver este problema, surgió el denominado *muestreo de Gibbs por bloques*, desarrollado por Jensen, Kong y Kjærulff [13]. Estos autores se dan cuenta de que los problemas de la simulación estocástica se deben a la dependencia entre las configuraciones de una muestra, en el sentido de que, en cada momento, sólo se cambia el valor de una variable. Esto no ocurre en el muestreo hacia delante, en el que todas las variables pueden cambiar de valor de una configuración a la siguiente en una muestra.

El muestreo de Gibbs por bloques es un sofisticado método que se basa en buscar un compromiso entre dependencia entre las configuraciones y coste computacional, partiendo de los dos casos extremos:

1. Simular una sola variable cada vez dada su envolvente de Markov es computacionalmente simple, pero las muestras pueden ser muy dependientes.
2. Simular todas las variables a la vez hace que las muestras sean independientes, pero el coste computacional puede ser intratable.

El método consiste en dividir las variables de la red en una serie de grupos de forma que todas las variables en un mismo grupo se simulan a la vez. Cuanto más grande sea cada grupo, menor será la dependencia entre las muestras, pero mayor será la complejidad de calcular la distribución conjunta que ha de usarse para simular las variables del grupo a la vez.

5 Muestreo Estratificado o Sistemático

La simulación estratificada es una técnica muy conocida en estadística [16] que conduce el proceso de simulación de forma que se eviten las muestras raras o desequilibradas. La idea básica consiste en dividir el espacio muestral en diversas regiones o estratos y elegir en cada uno de ellos un número óptimo de muestras. Esto produce una mejor representación del espacio muestral que la que resulta de las muestras aleatorias, y se pueden obtener mejores estimaciones para un tamaño determinado de la muestra o bien reducir el tamaño de la muestra para obtener la precisión requerida.

Los primeros algoritmos de propagación basados en muestreo estratificado fueron desarrollados por Bouckaert [1] y Bouckaert, Castillo y Gutiérrez [2]. La idea es considerar el espacio de todas las posibles configuraciones de las variables de la red, y asignar a cada una de ellas un subintervalo de $[0,1]$, de tal forma que las configuraciones más probables tengan asignado un subintervalo más amplio. Entonces, se selecciona un grupo de configuraciones muestreando sobre el intervalo $[0,1]$. El procedimiento es el siguiente:

Sea un conjunto de variables $X = \{X_1, \ldots, X_n\}$, donde cada variable X_i toma valores en $U_i = \{0, 1, \ldots, r_i - 1\}$. Sean f_i, $i = 1, \ldots, n$ las distribuciones condicionadas para cada variable dados sus padres en la red. En estas condiciones, podemos calcular todas las posibles configuraciones de las variables y su probabilidad de ocurrencia. El método de muestreo estratificado requiere que las configuraciones estén ordenadas, por ejemplo, según el siguiente criterio [2]:

Definición 1. Sean $x = (x_1, x_2, \ldots, x_n)$ e $y = (y_1, y_2, \ldots, y_n)$ dos configuraciones de la variable n-dimensional X. Se dice que x *precede* a y, y se denota $x < y$ si:

$$x < y \iff \exists k \ t.q. \ \forall j < k \quad x_j = y_j \text{ y } x_k < y_k. \tag{14}$$

□

En base al orden definido en (14), se construye una tabla que representa el espacio muestral. Esta tabla se usa para obtener las configuraciones en el proceso de muestreo. Por ejemplo, sea $X = \{X_1, X_2, X_3\}$ el conjunto de variables de la red de la figura 2, cuyas probabilidades *a priori* se encuentran en la tabla 1. En la tabla 4 pueden verse las configuraciones ordenadas y su probabilidad de ocurrencia, probabilidad acumulada e intervalo asociado. Cada configuración x^i

Configuración	Probabilidad	Prob. acumulada	Intervalo asociado
(0,0,0)	0.024	0.024	(0.000,0.024)
(0,0,1)	0.096	0.120	(0.024,0.120)
(0,1,0)	0.144	0.264	(0.120,0.264)
(0,1,1)	0.336	0.600	(0.264,0.600)
(1,0,0)	0.040	0.640	(0.600,0.640)
(1,0,1)	0.160	0.800	(0.640,0.800)
(1,1,0)	0.060	0.860	(0.800,0.860)
(1,1,1)	0.140	1.000	(0.860,1.000)

Tabla 4. Probabilidades e intervalos para las configuraciones ordenadas.

tiene asociado un intervalo $I_i = [l(i), h(i)) \subseteq [0,1]$ cuyos límites se calculan a partir de las probabilidades acumuladas de acuerdo a las siguientes expresiones:

$$\begin{aligned} l(i) &= \sum_{j<i} \prod_{r=1}^{n} f_r^*(x^{j\downarrow r}), \\ h(i) &= l(i) + \prod_{r=1}^{n} f_r^*(x^{i\downarrow r}). \end{aligned} \tag{15}$$

donde x^j es la j-ésima configuración de la variable n-dimensional X y f_r^*, $r = 1, \ldots, n$, son las distribuciones de muestreo. La figura 3 muestra la división del intervalo $[0,1]$ para la red de la figura 2.

Para obtener una muestra de tamaño m, se generan m números en el intervalo $[0,1]$, y se comprueba qué configuración se corresponde con cada número generado, de acuerdo a la partición de la región (figura 3). A continuación, se pondera cada configuración de acuerdo con la distribución usada para calcular los intervalos (f_r^*) y la distribución original. Los m números no son aleatorios, sino que se calculan de forma determinista [2] de la siguiente manera,

$$k_i = \frac{i - 0.5}{m}, \quad i = 1, 2, \ldots, m.$$

El hecho de que los números "aleatorios" sean generados aquí de forma determinista, motiva el nombre de *muestreo sistemático* para este método.

El siguiente ejemplo explica cómo obtener una muestra a partir de una secuencia de números dada.

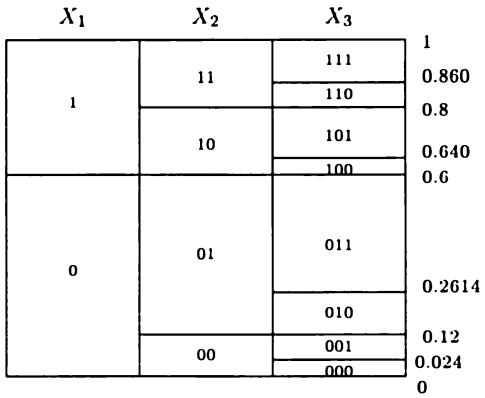


Figura 3. Configuraciones y sus probabilidades acumuladas.

Ejemplo 7. Considérese la red mostrada en la figura 2. Generando cuatro números $k_i = (i - 0.5)/4$, $i = 1, \ldots, 4$, obtenemos la secuencia,

$$(0.125, 0.375, 0.625, 0.875).$$

Ahora, para cada número, buscamos en el diagrama representado en la figura 3 las configuraciones correspondientes. Éstas son:

Número	*Configuración* (x_1, x_2, x_3)
0.125	(0, 1, 0)
0.375	(0, 1, 1)
0.625	(1, 0, 0)
0.875	(1, 1, 1)

□

Se puede apreciar que cuando m aumenta, la frecuencia relativa de cada configuración converge a su valor de probabilidad. El hecho de que no se utilicen números aleatorios hace que este algoritmo tenga un carácter más numérico que de simulación. Nótese que las funciones de muestreo pueden ser cualesquiera, luego dependiendo de las que se usen, se obtendrán distintos resultados. Bouckaert, Castillo y Gutiérrez [2] usan las mismas funciones que en el algoritmo de ponderación por verosimilitud. Una descripción detallada del algoritmo correspondiente a este método puede encontrarse en [5].

6 Muestreo por Importancia basado en Precomputación Aproximada

La decisión más importante a la hora de diseñar un algoritmo de muestreo por importancia es la elección de la distribución de muestreo: ésta debería ser tan similar a la distribución original como sea posible. En el caso particular de una red causal, la distribución original viene dada como el producto de una serie de distribuciones condicionadas y un conjunto de observaciones. Los algoritmos conocidos de muestreo por importancia [4,9,18] usan las funciones originales (distribuciones condicionadas u observaciones) para aproximar la distribución producto. Es decir, estos métodos usan exclusivamente *información local* sobre cada variable a la hora de simularla.

En esta sección veremos un nuevo enfoque para obtener las distribuciones de muestreo. La idea es usar no sólo las condicionadas y las observaciones originales, sino toda la información concerniente a cada variable. Esto es, a la hora de simular valores para una variable, usar todas las funciones de las que disponemos. Éste es el caso ideal, pero si la red es suficientemente complicada, este proceso puede ser inviable; en concreto, la complejidad de este procedimiento sería la misma que la de la propagación exacta, y eso es precisamente lo que queremos evitar. En resumen, el problema es que el coste de la combinación de todas las funciones definidas para una variable puede ser demasiado alto.

El esquema que describimos en esta sección tiene dos fases principales: *precomputación aproximada* y *simulación*. La primera de ellas se basa en realizar una eliminación de variables para encontrar una aproximación de las funciones de muestreo. En la fase de simulación, se utilizan estas funciones obtenidas para generar configuraciones de las variables que serán ponderadas como en los métodos anteriores.

Por *eliminación de una variable* entendemos el proceso de combinación de todas las funciones definidas para dicha variable y la posterior marginalización de la función obtenida sobre el resto de variables. A saber, hay dos formas de realizar la eliminación de una variable X_i: exacta y aproximada.

Exacta

1. Combinar todas las funciones que están definidas para la variable X_i, obteniendo como resultado una función h_i.
2. Eliminar X_i de la combinación, h_i, marginalizando el resultado a $s(h_i) - \{i\}$.
3. Añadir el resultado de la marginalización a H.
4. Eliminar de H todas las funciones que se combinaron para obtener h_i.

Si es posible repetir este proceso para todas las variables, en cada paso se obtiene una distribución de muestreo proporcional a $p(x, e)$. En realidad, el proceso es como un algoritmo de propagación exacta [19], y se verifica el siguiente teorema:

Teorema 2. Supongamos que hemos realizado una eliminación exacta; entonces,

- Si h_n es la función obtenida al eliminar X_n entonces, para todo $x \in U_{\sigma(n)}$, $h_n(x)$ es proporcional a $p(x|e)$.
- Si h_i es la función obtenida al eliminar X_i $(i < n)$, $\Sigma(i) = \{i+1, \ldots, n\}$, y $x_0 \in U_{\Sigma(i) \cap s(h_i)}$, entonces, la restricción de h_i a x_0, h_i' es proporcional a la probabilidad $p(.|e, x_0)$.

□

Las dos propiedades del teorema anterior nos permiten simular un valor $x \in U_N$ con probabilidad igual a $p(x|e)$. Lo que tenemos que hacer es simular valores para las variables en el orden $X_n, \ldots, X_1$. Para obtener un valor para una variable X_i, muestreamos a partir de la función h_i, realizando primero la restricción de esta función a los valores x_0 obtenidos para las variables simuladas previamente $(X_{\Sigma(i)})$ y normalizando después.

En algunos casos, el tamaño[3] de h_i puede ser tan grande que su cálculo sea inviable. En este caso, la eliminación de las variables habrá de hacerse de forma aproximada. Pueden definirse numerosos criterios de aproximación, pero siempre dentro del siguiente esquema:

Aproximado

1. Sea $H(i) = \{h \in H \mid i \in s(h)\}$, el conjunto de funciones definidas para la variable X_i. Eliminar $H(i)$ de H.
2. Transformar $H(i)$ mediante combinación. Para ello, repetir el siguiente proceso:
   (a) Tomar $R \subseteq H(i)$.
   (b) Combinar todas las funciones contenidas en R, es decir, calcular $f = \prod_{h \in R} h$.
   (c) Añadir el resultado de la combinación, f, a $H(i)$.
   (d) Eliminar R de $H(i)$.
3. Calcular $H^+(i)$ a partir de $H(i)$ eliminando X_i en todas las funciones pertenecientes a $H(i)$.

[3] Se define el *tamaño* de una función h como el producto del número de casos de todas las variables para las cuales h está definida.

4. Añadir $H^+(i)$ a H.

Este procedimiento coincide con el exacto si en el segundo paso se combinan todas las funciones contenidas en $H(i)$. La idea del procedimiento aproximado es combinar funciones mientras no se sobrepase cierto umbral de tamaño. Es decir, la forma de elegir los $R \subseteq H(i)$ dependerá del tamaño del resultado de combinar las funciones que lo formen. Una propiedad importante de esta aproximación de cara a su validez para obtener funciones del muestreo por importancia es que no se añaden nuevos ceros. Esto queda garantizado por el siguiente lema:

Lema 1. Sean $H(i)$ y $H^+(i)$ como en el algoritmo aproximado. Sea $x \in U_N$. Se verifica que

$$h(x^{\downarrow s(h)}) > 0 \ \ \forall h \in H(i) \Rightarrow h(x^{\downarrow s(h)}) > 0 \ \ \forall h \in H^+(i).$$

□

Una vez realizado el proceso de eliminación, el siguiente paso es obtener configuraciones de las variables X_N. El proceso para simular un valor para una variable X_i según el algoritmo aproximado es el siguiente: si x_0 es la configuración obtenida para las variables $X_{\Sigma(i)}$, entonces

Simula(X_i,$H(i)$)

1. Sea $H(i)$ el conjunto calculado en el paso 2 del procedimiento de eliminación aproximada.
2. Restringir cada función en $H(i)$ a x_0. Combinar todas las funciones en $H(i)$, obteniendo una nueva función h'_i definida sobre $U_{\sigma(i)}$.
3. Si $N(h'_i)$ es la normalización de h'_i, obtener un valor x_i para X_i siguiendo la distribución de probabilidad $N(h'_i)$.
4. Devolver el valor x_i.

Habiendo definido una forma de calcular las distribuciones de muestreo y de simular valores para las variables, se puede diseñar un algoritmo de propagación sin más que seguir el esquema general de la sección 3.1.

Referencias

1. Bouckaert, R.R., A stratified simulation scheme for inference in Bayesian belief networks. En: *Uncertainty in Artificial Intelligence, Proceedings of the Tenth Conference*, pp. 110-117, 1994.
2. Bouckaert, R.R., E. Castillo, J.M. Gutiérrez, A modified simulation scheme for inference in Bayesian networks. *International Journal of Approximate Reasoning*, 14, pp. 55-80, 1996.
3. Breiman, L., *Probability*. Addison Wesley. 1968.
4. Cano, J.E., L.D. Hernández, S. Moral, Importance sampling algorithms for the propagation of probabilities in belief networks. *International Journal of Approximate Reasoning*, 15, pp. 77-92, 1996.
5. Castillo, E., J.M. Gutiérrez, A.S. Hadi, *Sistemas expertos y modelos de redes probabilísticas*. Monografías de la Academia de Ingeniería. 1996.
6. Cooper, G.F., The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 42, pp. 393-405, 1990.
7. Dagum, P., M. Luby, Approximating probabilistic inference in Bayesian networks is NP-hard. *Artificial Intelligence*, 60, pp. 141-153, 1993.
8. Feller, W., *Introducción a la teoría de probabilidades y sus aplicaciones*. Limusa. 1973.
9. Fung, R., K.C. Chang, Weighting and integrating evidence for stochastic simulation in Bayesian networks. En: *Uncertainty in Artificial Intelligence* 5. (M. Henrion, R.D. Shachter, L.N. Kanal, J.F. Lemmer, eds.) North-Holland (Amsterdam), pp. 209-220. 1990.
10. Henrion, M., Propagating uncertainty by logic sampling in Bayes networks. En: *Uncertainty in Artificial Intelligence*, 2 (J.F. Lemmer, L.N. Kanal, eds.) North-Holland (Amsterdam), pp. 317-324, 1988.
11. Hernández, L.D., S. Moral, A. Salmerón, Importance sampling algorithms for belief networks based on approximate computation. *Proceedings of the Sixth International Conference IPMU'96*. Vol. II, pp. 859-864, 1996.
12. Hernández, L.D., S. Moral, A. Salmerón, A Monte Carlo algorithm for probabilistic propagation based on importance sampling and stratified simulation techniques. *International Journal of Approximate Reasoning*. 1998. En prensa.
13. Jensen, C.S., A. Kong, U. Kjærulff, Blocking Gibbs sampling in very large probabilistic expert systems. *International Journal of Human-Computer Studies*, 42, pp. 647-666, 1995.
14. Jensen, F.V., *An introduction to Bayesian networks*. UCL Press. 1996.
15. Pearl, J., Evidential reasoning using stochastic simulation of causal models. *Artificial Intelligence*, 32, pp. 247-257, 1987.
16. Rubinstein, R.Y., *Simulation and the Monte Carlo Method*. Wiley (New York), 1981.
17. Salmerón, A., *Precomputación en grafos de dependencias mediante algoritmos aproximados*. Tesis Doctoral. Universidad de Granada. 1998.
18. Shachter, R.D., M.A. Peot, Simulation approaches to general probabilistic inference on belief networks. En: *Uncertainty in Artificial Intelligence* 5, (M. Henrion, R.D.

Shachter, L.N. Kanal, J.F. Lemmer, eds.) North Holland (Amsterdam), pp. 221-231. 1990.

19. Shafer, G., P.P. Shenoy, Probability propagation. *Annals of Mathematical and Artificial Intelligence*, 2, pp. 327-351. 1990.

Abducción en Modelos Gráficos

José A. Gámez

Dpto. de Informática
Universidad de Castilla-La Mancha
Albacete. 02071
correo-e: jgamez@info-ab.uclm.es

Resumen

En este trabajo pretendemos introducir el concepto de inferencia abductiva en sistemas probabilísticos y su resolución haciendo uso de modelos gráficos, concretamente redes causales Bayesianas. Comenzaremos por plantear versiones limitadas del problema, para abordar después la inferencia abductiva usando el formalismo de las redes causales. Distinguiremos dos problemas de abducción en redes causales, la abducción total y la abducción parcial. Veremos como ésta última (que puede verse como una generalización de la primera) puede resultar más interesante de cara a problemas prácticos y presenta más problemas para ser resuelta de manera eficiente.

1 Introducción

En los últimos años el razonamiento abductivo ha incrementado su interés en varios campos de investigación relacionados con la inteligencia artificial, como por ejemplo en tareas de análisis y diagnóstico [16,22,23], comprensión del lenguaje natural [30], visión artificial y procesamiento de imágenes [9], generación de planes [1], etc ...

El problema de la *abducción* puede plantearse como la búsqueda de explicaciones a unos hechos observados. Es, por tanto, una regla de inferencia (*inferencia abductiva*) [15]que sigue el siguiente esquema:

regla general	: todas las bolas de la caja A son negras
hecho	: la bola es negra
hipótesis	: la bola es de la caja A

Al igual que en la deducción, en la abducción a partir de un caso particular y de una regla general se obtiene un caso particular; sin embargo, en la deducción el resultado es una consecuencia lógica de la regla general y por tanto "cierto",

mientras que en la abducción el resultado es simplemente una "hipótesis"(una posible explicación al hecho observado) y no una conclusión definitivamente cierta. Otra diferencia entre la deducción y la abducción es que en la primera se requiere una implicación absoluta en la formulación de las reglas (si $A \Rightarrow B$, siempre que se de A es seguro que se da B), mientras que en la abducción la implicación puede relajarse y ser interpretada como una relación causal (si $A \Rightarrow B$, A es una posible explicación de B). Levesque [10] sugiere extender la noción de *explicación* para englobar aquellos casos en los que aunque no exista una relación causal directa entre A y B, conocer A sea suficiente para creer B como cierta. En la mayoría de las aproximaciones actuales las reglas usadas en la inferencia abductiva utilizan la implicación material (lógica) interpretada como una especie de relación causal.

En general el proceso de inferencia abductiva siempre produce más de una explicación posible, lo que hace que sea necesario discriminar entre las distintas alternativas. Los criterios que se utilizan para seleccionar las mejores explicaciones del conjunto de hipótesis generadas se basan en utilizar alguna medida que nos diga cuando una hipótesis es mejor que otra y en preferir siempre las hipótesis más simples (de acuerdo a algún criterio de simplicidad).La figura 1 muestra el proceso de la inferencia abductiva, diferenciándose claramente entre las fases de *generación* y *selección* de hipótesis.

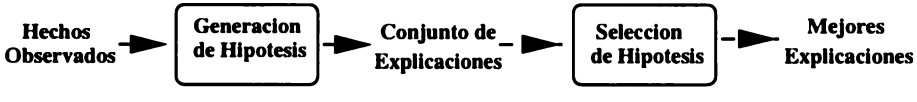


Figura 1. Proceso de inferencia abductiva.

El resto del capítulo se estructura como sigue: en la sección 2 se introduce el concepto de abducción en relación con la tarea de diagnóstico y los sistemas expertos. La sección 3 plantea el problema clásico de diagnóstico (como una red causal restringida) y su resolución por medio de la teoría del recubrimiento parsimonioso. En las secciones 4 y 5 se presentan, respectivamente, los problemas de abducción total y parcial en redes causales, así como una introducción a los métodos de resolución del problema. Por último, en la sección 6 presentamos las conclusiones.

2 Abducción, diagnóstico y sistemas expertos

Está ampliamente aceptado que el proceso del diagnóstico humano pertenece a la categoría de la inferencia abductiva [3,18,20,22] y que probablemente sea el ejemplo más típico y mejor comprendido de la clase de problemas que pueden ser resueltos mediante este tipo de inferencia. Consideremos el diagnóstico médico como un ejemplo. El conocimiento viene presentado como relaciones entre enfermedades y síntomas en la forma *la enfermedad* e_i *puede provocar los síntomas* $s_1, s_2, \ldots, s_k$. Además, asociado a cada una de las relaciones causales hay un grado de incertidumbre, ya que tanto la *gripe* como un *tumor cerebral* pueden provocar un *dolor de cabeza*, si bien el grado de certeza asociado a la relación en ambos casos será distinto. Ante unos síntomas observados la tarea del médico es identificar el conjunto de enfermedades que expliquen los síntomas observados. Si ha identificado más de un diagnóstico posible, se decidirá por áquel que tenga asociado un grado de certeza mayor.

Tradicionalmente muchos de los sistemas expertos desarrollados se han centrado en el campo del diagnóstico médico y, por tanto, podemos decir que realizaban tareas abductivas [19,11,21]. En líneas generales, cuando la incertidumbre era representada con probabilidades muchos de estos sistemas trabajaban usando variaciones de los siguientes esquemas. Distinguiremos dos casos: una única enfermedad y múltiples enfermedades[1].

2.1 Una única enfermedad

En algunos sistemas (IDT [29]) se realiza la suposición de que dos o más enfermedades no pueden estar presentes de manera simultánea y por tanto el diagnóstico resultante sólo contiene una enfermedad. Supongamos que se han observado los síntomas $\{s_1, s_2, \ldots, s_k\}$, entonces el objetivo es obtener la enfermedad e que maximiza la cantidad $p(e|s_1, s_2, \ldots, s_k)$. Aplicando la regla de Bayes tenemos:

$$p(e|s_1, s_2, \ldots, s_k) = \frac{p(e)p(s_1, s_2, \ldots, s_k|e)}{p(s_1, s_2, \ldots, s_k)} \quad (1)$$

La realización de estos cálculos para todas las enfermedades implicaba un esfuerzo computacional inviable y por eso se hacían algunas suposiciones como las siguientes:

[1] Aunque aquí siempre nos estamos refiriendo a enfermedades y síntomas es claro que el modelo puede extenderse a cualquier sistema de diagnóstico, sin más que considerar desórdenes y manifestaciones en general

- Independencia entre síntomas, es decir, $p(s_i, s_j) = p(s_i)p(s_j)$.
- Independencia entre síntomas dada una enfermedad, es decir, $P(s_i|e, s_j) = p(s_i|e)$.

Considerando las suposiciones anteriores la expresión 1 queda de la siguiente forma:

$$p(e|s_1, s_2, \ldots, s_k) = p(e) \cdot \frac{p(s_1|e)}{p(s_1)} \cdot \frac{p(s_2|e)}{p(s_2)} \cdot \ldots \cdot \frac{p(s_k|e)}{p(s_k)} \quad (2)$$

Esta regla permite considerar los síntomas uno a uno con el siguiente algoritmo:

1. Para cada enfermedad e_i hacer
   $A(e_i) = p(e_i)$
2. Para cada síntoma observado s_j hacer
   Para cada enfermedad e_i hacer
   $A(e_i) = A(e_i) \cdot \frac{p(s_j|e)}{p(s_j)}$
3. Listar las N enfermedades con mayor valor A

El algoritmo anterior procede inicializando las enfermedades con sus valores a priori y luego va actualizando el valor en función de los síntomas observados. Es claro que no todos los síntomas afectan a todas las enfermedades y aunque el algoritmo trata bien estos casos (ya que multiplica por 1) sería conveniente obtener antes una lista de las enfermedades relacionadas con cada síntoma, de forma que no se pierda tiempo en realizar esos cálculos.

2.2 Múltiples enfermedades

La hipótesis de que sólo una enfermedad puede estar presente no tiene por qué cumplirse y de ahí que haya que modificar el anterior esquema de funcionamiento. En este apartado vamos a ver como el sistema experto *Internist-1/Caduceus*[2] [11,18,19] trataba el problema de la presencia de múltiples enfermedades utilizando técnicas heurísticas. La idea se basa en dividir la lista de enfermedades en grupos distintos, formados por aquellas enfermedades que explican aproximadamente el mismo conjunto de síntomas. El esquema seguido por *Internist-1/Caduceus* era más o menos el siguiente.

1. Sea S el conjunto de síntomas observados y E_S el conjunto de enfermedades relacionadas con S. Hacer $D = \emptyset$.

[2] *Internist-1/Caduceus* no es un sistema estrictamente Bayesiano, sin embargo, las diferencias no son muy grandes y podemos obviarlas por razones de simplicidad

2. Aplicar el algoritmo del apartado anterior y seleccionar la primera enfermedad e_1 de la lista resultante [3].
3. Sea S_{e_1} el conjunto de síntomas observados que son explicados por e_1 y E_{e_1} el conjunto de enfermedades de la lista obtenida en el paso anterior que explican como mucho el conjunto de síntomas S_{e_1}.
4. Hacer $S = S \setminus S_{e_1}$, $E = E_S \setminus E_{e_1}$ y $D = D \cup \{e_1\}$.
5. Si todos los síntomas han sido explicados ($S = \emptyset$) finalizar con D como diagnóstico, en otro caso volver al paso 2.

Como puede verse este método es una generalización del anterior para poder trabajar con el caso de que varias enfermedades puedan estar presentes de manera simultánea. Un primer problema que podemos observar es que si el paciente sufre dos enfermedades, una de las cuales oculta a la otra (los síntomas de una son un subconjunto de los de la otra), únicamente una de ellas será diagnosticada.

3 El problema clásico de diagnóstico y la teoría del recubrimiento parsimonioso

En el apartado anterior hemos visto cómo operaban algunos sistemas expertos que utilizaban inferencia abductiva para resolver problemas de diagnóstico, compuestos por síntomas y enfermedades (manifestaciones y desórdenes en gereral). También hemos visto algunos de los problemas que tenían estos métodos y sus restricciones a la hora de aplicarlos debido a las suposiciones formuladas para poder aplicarlos. En este apartado vamos a ver una formalización de los problemas que constan de manifestaciones y desórdenes, representándolos como una red causal (restringida); y un método de resolución que evita algunos de los problemas anteriores.

Los problemas que relacionan desórdenes y manifestaciones pueden ser representados mediante una red causal de dos niveles, en los cuales cada una de las variables puede tomar dos valores (presencia x_i o ausencia $\neg x_i$). Nuestro problema estará caracterizado por la siguiente definición [17,12].

Definición 1. (Problema clásico de diagnóstico)

Un *problema clásico de diagnóstico* es una 4-tupla $\mathcal{P} = (D, M, C, M^+)$ donde:

- $D = \{d_1, d_2, \ldots, d_n\}$ es un conjunto finito y no vacío de objetos, llamados *desórdenes.*

[3] Si la diferencia entre la primera y la segunda enfermedad no era suficientemente significativa se solicitaban más datos, hasta obtener una enfermedad ***destacada*** con respecto a las demás

- $M = \{m_1, m_2, \ldots, m_k\}$ es un conjunto finito y no vacío de objetos, llamados *manifestaciones*.
- $C \subseteq D \times M$ es una relación entre desórdenes y manifestaciones y representa el grafo de la red causal.
- $M^+ \subseteq M$ es el conjunto de manifestaciones que se ha observado que están presentes.

□

En la figura 2 podemos ver un ejemplo formado por cinco desórdenes y seis manifestaciones. Notaremos por *efectos*(d_i) al conjunto de manifestaciones directamente causadas por d_i (hijos de d_i en el grafo) y por *causas*(m_j) al conjunto de desórdenes que pueden causar de forma directa la manifestación m_j (padres de m_j en el grafo). Es importante destacar que una arista (d_i, m_j) no se interpreta como "d_i provoca necesariamente m_j", sino que debe interpretarse como que "d_i podría provocar la manifestación m_j".

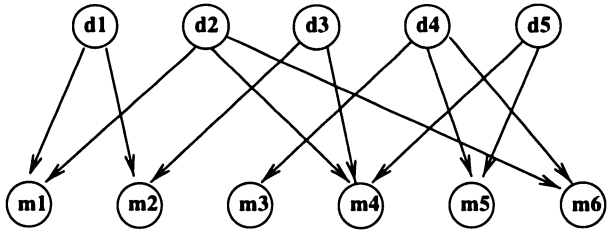


Figura 2. Problema clásico de diagnóstico con cinco desórdenes y seis manifestaciones.

Peng y Reggia [16,17] han estudiado de forma extensa cómo aplicar la inferencia abductiva al problema clásico de diagnóstico: primero, desde un punto de vista simbólico; y posteriormente, añadiendo la componente probabilística. A continuación vamos a comentar brevemente la base de ambos enfoques.

3.1 Teoría del recubrimiento parsimonioso

Una vez hemos caracterizado los problemas de diagnóstico con los que vamos a trabajar, debemos caracterizar ahora cómo resolverlos. Utilizaremos básicamente dos técnicas: el *recubrimiento* para obtener explicaciones y la *parsimonia*, que nos permitirá seleccionar de entre el conjunto de explicaciones. Veamos algunas definiciones básicas.

Definición 2. Para cualquier $D_I \subseteq D$ y $M_J \subseteq M$ en un problema clásico de diagnóstico $\mathcal{P}$ se tiene

- $efectos(D_I) = \bigcup_{d_i \in D_I} efectos(d_i)$, y
- $causas(M_J) = \bigcup_{m_j \in M_J} causas(m_j)$.

□

Definición 3. (Covertura)
El conjunto $D_I \subseteq D$ es una *covertura* de $M_J \subseteq M$ si $M_J \subseteq efectos(D_I)$. □

Podemos decir que la noción de *covertura* representa que D_I explica desde un punto de vista causal la presencia de M_J.

Ejemplo 1. Supongamos que para el problema clásico de diagnóstico de la figura 2 se tiene el conjunto de manifestaciones $\{m_1, m_2, m3\}$, entonces obtenemos las siguientes coverturas. $\{\{d_1, d_2, d_3, d_4, d_5\}, \{d_1, d_2, d_3, d_4\}, \{d_1, d_3, d_4\}, \{d_2, d_3, d_4\}, \{d_1, d_2, d_4\}, \{d_1, d_4\}\}$ □

Definición 4. (Explicación)
Un conjunto $E \subseteq D$ es una *explicación* de M^+ para un $\mathcal{P} = (D, M, C, M^+)$ sii E es una covertura de M^+ y E satisface un criterio dado de parsimonia. □

Por tanto, la noción de *explicación* consta de tres condiciones: debe ser una covertura de M^+, únicamente contiene desórdenes y debe cumplir un criterio de parsimonia. Si no se exigiera algún criterio de parsimonia, D siempre constituiría una explicación de cualquier conjunto M^+, puesto que siempre es una covertura. A continuación se introducen algunos criterios de parsimonia.

Definición 5. (Criterios de parsimonia)

1. Una covertura D_I de M_J se dice que es *mínima* si tiene menor cardinalidad que cualquier otra covertura de M_J.
2. Una covertura D_I de M_J se dice que es *irredundante* si ninguno de sus subconjuntos propios es también una covertura de M_J. En otro caso diremos que es *redundante*.
3. Una covertura D_I de M_J se dice que es *relevante* si es un subconjunto de $causas(M^+)$. En otro caso diremos que es *irrelevante*.

□

De acuerdo con estos criterios podemos clasificar las coverturas del ejemplo 1 de la siguiente forma:

$\{d_1, d_2, d_3, d_4, d_5\}$	irrelevante
$\{d_1, d_2, d_3, d_4\}$	redundante
$\{d_1, d_3, d_4\}$	redundante
$\{d_2, d_3, d_4\}$	irredundante
$\{d_1, d_2, d_4\}$	redundante
$\{d_1, d_4\}$	irredundante y mínima

Si bien se pueden formular otros criterios de parsimonia, en ausencia de otra información se preferirá la *irredundancia* por motivos conceptuales y computacionales. Esto hace que cuando hablemos de una covertura parsimoniosa nos estemos refiriendo en realidad a una covertura irredundante.

Por último, en muchos problemas de diagnóstico es interesante conocer todos los (mejores) diagnósticos posibles y no sólo el mejor, ya que este mayor conocimiento puede servir para decidir las acciones a realizar. Para recoger esta idea se formula la siguiente definición.

Definición 6. (Solución)
La *solución* a un problema clásico de diagnóstico $\mathcal{P} = (D, M, C, M^+)$, denotada por $sol(\mathcal{P})$, es el conjunto de todas las explicaciones de M^+. □

Continuando con el problema planteado en el ejemplo 1 tendremos que la solución es $sol(\mathcal{P}) = \{\{d_1, d_4\}, \{d_2, d_3, d_4\}\}$.

3.2 Extensión probabilística de la teoría del recubrimiento parsimonioso

Una limitación de la teoría del recubrimiento parsimonioso es que para un problema $\mathcal{P}$, el conjunto de soluciones $sol(\mathcal{P})$ puede contener un gran número de explicaciones potenciales. Esto hace que se tenga que introducir un nuevo criterio de selección, de forma que podamos elegir las mejores explicaciones. Para hacer esto Peng y Reggia [16,17] incorporan conocimiento probabilístico al modelo simbólico antes presentado. La idea ahora es calcular $P(D_I|M^+)$ para cada $D_I \in sol(\mathcal{P})$ y ordenar las explicaciones de acuerdo a esta medida, quedándonos, por tanto, con las *explicaciones más probables.*

De cara al desarrollo del modelo probabilístico, el conjunto M^+ representará que toda manifestación $m_j \in M^+$ está presente (estado m_j) y que el resto de las manifestaciones $(M \setminus M^+)$ están ausentes (estado $\neg m_j$). De forma análoga, el conjunto D_I representa que todo desorden $d_i \in D_I$ está presente y que el resto de los desórdenes $(D \setminus D_I)$ están ausentes. En adelante supondremos implícita esta notación para los conjuntos.

Como ya se ha comentado el objetivo ahora es establecer un orden entre las explicaciones $D_I \in sol(\mathcal{P})$, utilizando para ello la *probabilidad a posteriori* $P(D_I|M^+)$. Sin embargo, aplicando la regla de Bayes tenemos

$$P(D_I|M^+) = \frac{P(M^+|D_I)P(D_I)}{\sum_{D_J \in 2^D} P(M^+|D_J)P(D_J)} = \frac{P(M^+|D_I)P(D_I)}{P(M^+)},$$

de donde se puede observar que el denominador es constante para toda explicación D_I y, por tanto, para establecer un orden sólo necesitamos calcular el numerador. El objetivo ahora es desarrollar un método que permita abordar estos cálculos desde el punto de vista computacional.

Definiciones básicas y suposiciones

Veamos algunas definiciones básicas y las suposiciones que Peng y Reggia asumen para facilitar la tarea computacional y para que el modelo numérico sea coherente con el modelo simbólico antes descrito.

Definición 7. (Suceso causal)
Para cualquier $d_i \in D$ y $m_j \in M$, $d_i \rightarrow m_j$ representa el suceso de que d_i es la causa de m_j en este momento. Por tanto, $d_i \rightarrow m_j$ es cierto sii d_i y m_j están presentes y además en este momento m_j está siendo causado por d_i. □

De la definición anterior se deduce que $d_i \rightarrow m_j$ implica $d_i \wedge m_j$ pero no al revés. Además, $P(d_i \rightarrow m_j|d_i) \leq P(m_j|d_i)$, ya que en $P(m_j|d_i)$ también se recogen las situaciones en que $m_j \wedge d_i$ es cierto, pero m_j está siendo provocado por algún otro desorden d_k también presente. Por otra parte, $P(d_i \rightarrow m_j|d_i)$ puede interpretarse como la frecuencia media con que d_i provoca la presencia de m_j.

Definición 8. (Contexto de $d_i \rightarrow m_j$)
Sea X la conjunción de cualquier causa y sucesos causales o sus negaciones, excluyendo $d_i \rightarrow m_j$ y $\neg(d_i \rightarrow m_j)$. Entonces se dice que X es un contexto de $d_i \rightarrow m_j$. □

Teniendo en cuenta estas definiciones y tomando $p_i = P(d_i)$ y $c_{ij} = P(d_i \rightarrow m_j|d_i)$ Peng y Reggia formulan las siguientes suposiciones:

- *Suposiciones respecto a la base de conocimiento.* Para todo desorden d_i p_i es conocido y $0 < p_i < 1$. Para todo suceso causal $d_i \rightarrow m_j$, c_{ij} es conocido y $c_{ij} > 0$ sii $(d_i, m_j) \in C$.
- *Suposiciones de independencia.*

- Independencia entre desórdenes. Un desorden d_i puede ocurrir independientemente de cualquier otro desorden.

$$P(D_I) = \prod_{d_i \in D_I} p_i \prod_{d_k \in D \setminus D_I} (1 - p_k) \tag{3}$$

  Esta independencia puede observarse en el grafo aplicando el criterio de d-separación.
- Independencia causal. Si $d_i \in D$ ocurre, entonces el suceso causal $d_i \rightarrow m_j$ ocurre independientemente de sus contextos. Es decir, si X es un contexto de $d_i \rightarrow m_j$ y $P(X, d_i) \neq 0$, entonces

$$P(d_i \rightarrow m_j | d_i, X) = P(d_i \rightarrow m_j | d_i) = c_{ij}$$

  Esto significa que la influencia particular de d_i sobre m_j no se ve afectada por otros sucesos. Esta suposición sustituye la hipótesis de independencia entre manifestaciones formulada en otros modelos.

– Por último, se supone que ninguna manifestación m_j puede ocurrir si no es causada por algún desorden a través de un suceso causal.

Cálculo de las probabilidades

A partir de las suposiciones anteriores Peng y Reggia obtienen los siguientes resultados:

– Si X es un contexto para $d_i \rightarrow m_j$ entonces:
  - $P(d_i \rightarrow m_j, X | d_i) = c_{ij} P(X | d_i)$
  - $P(d_i \rightarrow m_j, X) = c_{ij} P(d_i, X)$
– Un único fallo. En este caso y puesto que sólo hay un desorden se tiene que $P(m_j | d_i) = c_{ij}$ y por tanto:

$$P(M_J | d_i) = \prod_{m_j \in M_J} c_{ij} \prod_{m_k \notin M_J} (1 - c_{ik}) \tag{4}$$

– Múltiples fallos.

$$P(m_j | D_I) = 1 - \prod_{d_i \in D_I} (1 - c_{ij}) \tag{5}$$

$$P(M_J | D_I) = \prod_{m_j \in M_J} P(m_j | D_I) \prod_{m_k \notin M_J} (1 - P(m_k | D_I)) \tag{6}$$

Estos resultados son importantes porque nos permiten calcular las probabilidades condicionales a partir de los datos c_{ij} y p_i que tenemos especificados a priori. Además, en el segundo producto de las expresiones 4 y 6 no es necesario recorrer todo el índice $d_k \notin M_J$, sino sólo aquellas manifestaciones $m_k \in efectos(D_I) \setminus M^+$, ya que para el resto $c_{ik} = 0$ al no existir la arista (d_i, m_k) en el grafo. Lo mismo ocurre en el cálculo de $p(D_I)$, ya que la expresión 3 puede ponerse como:

$$P(D_I) = \prod_{d_i \in D_I} \frac{p_i}{1-p_i} \prod_{d_k \in D} (1-p_k) \tag{7}$$

$$\propto \prod_{d_i \in D_I} \frac{p_i}{1-p_i} \tag{8}$$

ya que $\prod_{d_k \in D}(1-p_k)$ es una constante para todos los $D_I \subseteq D$ y, por tanto, puede ser ignorada para establecer el orden entre los distintos $D_I \in sol(\mathcal{P})$.

Por último, es fácil ver que si D_I no es una covertura de M_J entonces se cumple $P(M_J|D_I) = 0$, ya que para alguna manifestación m_j se tendrá $c_{ij} = 0$ para todo $d_i \in D_I$. Además, si d_i provoca m_j con total certeza ($c_{ij} = 1$) y $m_j \notin M^+$ entonces también $P(M_J|D_I) = 0$ y la explicación será rechazada.

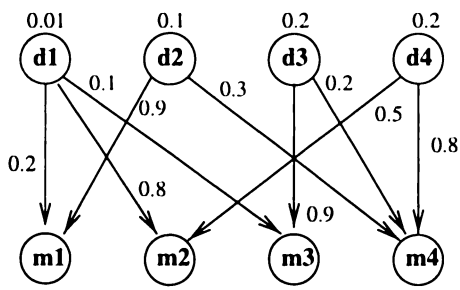


Figura 3. Problema clásico de diagnóstico con cuatro desórdenes y cuatro manifestaciones.

Ejemplo 2. Sea el problema clásico de diagnóstico dado por la figura 3 y el conjunto de manifestaciones $M^+ = \{m_1, m_3\}$. Aplicando la teoría del recubrimiento parsimonioso se obtiene el conjunto de soluciones $sol(\mathcal{P}) = \{\{d_1\}, \{d_2, d_3\}\}$. Queremos ahora ordenar estas explicaciones en función de su probabilidad.

Usando la expresión 8 tenemos:

$$P(\{d_1\}) \propto \frac{P(d_1)}{1-P(d_1)} = \frac{0.01}{0.99} = 0.0101$$

$$P(\{d_2,d_3\}) \propto \frac{P(d_2)\cdot P(d_3)}{(1-P(d_2))\cdot(1-P(d_3))} = \frac{0.1\cdot 0.2}{0.9\cdot 0.8} = 0.028$$

Usando las expresiones 4, 5 y 6 tenemos:

$$P(\{m_1,m_3\}|\{d_1\}) = c_{11}\cdot c_{13}\cdot(1-c_{12}) = 0.2\cdot 0.1\cdot(1-0.8) = 0.004$$

$$\begin{aligned} P(\{m_1,m_3\}|\{d_2,d_3\}) &= p(\{m_1\}|\{d_2,d_3\})\cdot p(\{m_3\}|\{d2,d3\})\cdot \\ &\quad \cdot(1-P(\{m_4\}|\{d_2,d_3\})) \\ &= 0.9\cdot 0.9\cdot 0.44 \\ &= 0.4536 \end{aligned}$$

Por tanto, tenemos que:

$$\begin{aligned} P(\{d_1\}|\{m_1,m_3\}) &\propto 0.0101\cdot 0.004 = 0.00004 \text{ y} \\ P(\{d_2,d_3\}|\{m_1,m_3\}) &\propto 0.028\cdot 0.4536 = 0.0127, \end{aligned}$$

siendo la explicación más probable de M^+ $\{d_2,d_3\}$. □

4 Abducción en redes causales Bayesianas

Peng y Reggia han generalizado con éxito la teoría del recubrimiento parsimonioso para poder trabajar con redes de más de dos niveles; sin embargo, no ocurre lo mismo con la extensión probabilística, que no es capaz de tratar los ciclos (debido a que violan la suposición de la independencia entre causas). Por tanto, para trabajar con redes causales Bayesianas sin restricciones en su topología es necesario el desarrollo de nuevos métodos. No obstante, antes de entrar en los métodos, veamos qué se entiende por abducción en redes causales Bayesianas.

El problema de hacer inferencia abductiva en redes causales, también llamado *revisión de creencias* o *búsqueda de la explicación más probable* por Pearl [14] y *búsqueda de la configuración máxima a posteriori* por Shimony y Charniak [28], consiste en encontrar la configuración de estados de mayor probabilidad para las variables no observadas. En general el problema se conoce como la búsqueda de las K explicaciones más probables, ya que habitualmente se pretenden encontrar las K mejores explicaciones a los hechos observados y no la primera únicamente. A continuación definimos formalmente los conceptos de *explicación* y *explicación más probable* dentro del contexto de las redes bayesianas.

Definición 9. (explicación)
Sea $G = (\mathcal{U}, E)$ una red bayesiana y x_O una observación del conjunto de variables $X_O \subset \mathcal{U}$. Decimos que $x \in \Omega_{\mathcal{U}}$ es una explicación de x_O si $x^{\downarrow X_O} = x_O$. □

Evidentemente esto produce un altísimo número de explicaciones, lo que hace necesario seleccionar entre ellas de alguna forma. El criterio de selección que se usa está basado en la probabilidad a posteriori de la explicación.

Definición 10. (explicación más probable)
Sea $G = (\mathcal{U}, E)$ una red bayesiana y x_O una observación del conjunto de variables $X_O \subset \mathcal{U}$. Decimos que $x \in \Omega_{\mathcal{U}}$ es la explicación más probable (EMP) de x_O si

$$x = \arg\max_{\mathcal{U}} P(\mathcal{U}|x_O) \tag{9}$$

□

La obtención de la explicación más probable x usando la expresión 9 no es equivalente a hacer:

$$x = x_1 \wedge x_2 \wedge \ldots \wedge x_{|\mathcal{U}|}, \text{ con } x_i = \max_{X_i} P(X_i|x_O), \ \forall X_i \in \mathcal{U} \setminus X_O$$

y, por tanto, no podemos resolver el problema de la inferencia abductiva en redes causales usando los métodos de propagación de probabilidades descritos en los capítulos anteriores.

4.1 Obtención de las K explicaciones más probables

En los últimos años se han desarrollado una serie de algoritmos para resolver de forma exacta el problema de la búsqueda de las K explicaciones más probables, sin embargo, en muchos de ellos o bien se restringe la topología de la red o se considera únicamente el caso de $K = 1$. En este trabajo nos vamos a limitar a comentar únicamente los métodos basados en propagación sobre el árbol de grupos maximales[4], por ser estos algoritmos los más usados en la actualidad [8,25]. En concreto nos referiremos a los algoritmos propuestos por Dawid [4] y por Seroussi y Goldmard [24].

Algoritmo de Dawid

El procedimiento propuesto por Dawid [4] consiste en aplicar el algoritmo Hugin utilizando el máximo como operador de marginalización en lugar de la

[4] También llamados árboles de conglomerados en este libro

suma, es decir, ahora la marginalización de un grupo G_i a su separador S_i se hace aplicando la siguiente expresión:

$$\psi(S_i) \leftarrow \psi(G_i)^{\downarrow S_i} = \max_{G_i \setminus S_i} \psi(G_i) \tag{10}$$

El procedimiento de realizar las fases de recolección y distribución usando el máximo como operador de marginalización recibe el nombre de *max-prop*. Sea $T = (\{G_1, \ldots, G_t\}, E)$, $\mathcal{U} = G_1 \cup \cdots \cup G_t$ y $P(\cdot)$ la distribución de probabilidad conjunta factorizada en T. Dawid indica que después de introducir la evidencia x_O en T y aplicar el procedimiento *max-prop* se cumple lo siguiente:

i) $\forall G_i \in T, \quad \max_{\mathcal{U}} P(\mathcal{U}, x_O) = \max_{G_i} \psi(G_i)$

ii) Sea $x = \arg \max_{\mathcal{U}} p(\mathcal{U}, x_O)$ la configuración de máxima probabilidad. Entonces x puede obtenerse mediante la composición de las g_i^* calculadas con el siguiente procedimiento:

1. $g_1^* = \arg \max_{G_1} \psi(G_1)$
2. Para $j = 2, \ldots, t$ hacer
   $i \quad = padre(G_j)$
   $s_{i,j}^* = g_i^{*\downarrow S_{i,j}}$
   $g_j^* \quad = \arg \max_{G_j \setminus S_{i,j}} \psi(G_j, s_{i,j}^*)$

La aplicación del método descrito en ii) es necesaria por si hay dos o más configuraciones de máxima probabilidad. Si sólo hay una configuración de máxima probabilidad ésta puede obtenerse directamente mediante la composición de las $g_i^* = \arg \max_{G_i} \psi(G_i)$. Por otra parte, para obtener la probabilidad asociada a la configuración de máxima probabilidad es necesario dividir el valor obtenido en i) por $P(x_O)$ que como se vió en el algoritmo Hugin puede calcularse sumando en el grupo raíz después de que haya finalizado la fase de recolección de evidencia invocada por este grupo.

Por último, indicar que como señala Nilsson [13] el algoritmo diseñado por Dawid no es válido (en general) para obtener la cuarta, quinta y sucesivas explicaciones más probables.

Algoritmo de Seroussi y Goldmard

Seroussi y Goldmard [24] plantean un algoritmo basado en árboles de grupos para obtener las K explicaciones más probables. La idea básica del algoritmo de obtención de la explicación más probable es utilizar un procedimiento ascendente que visita cada grupo del árbol, calculando en cada grupo $G_i = \{R_i, S_i\}$ la configuración de R_i que maximiza el potencial asociado al grupo G_i. Esto puede

hacerse debido a que la información relevante a las variables del conjunto residual R_i está contenida en el subárbol que tiene como raíz a G_i y no se ve afectada por la información contenida en el resto de los grupos. Ahora cada configuración g_i de un grupo G_i tiene asociado además de su potencial, la instanciación máxima de las variables pertenecientes a los conjuntos residuales del subárbol que tiene como raíz a G_i, es decir, de las variables que ya han sido *borradas*. Los autores denotan a esta configuración por $config(g_i)$.

En lugar de dividir por la probabilidad de la evidencia $P(x_O)$ como hace el algoritmo de Dawid, Seroussi y Goldmard ejecutan una fase previa en la que instancian la evidencia y modifican el potencial de cada grupo G_i del árbol a $P(R_i|S_i, x_O)$. Después de esta fase se realiza la propagación[5] en orden ascendente y al final la explicación más probable viene dada por:

$$x^* = g_1^* \cup config(g_1^*), \quad \text{con } g_1^* = \arg\max_{G_1} \psi(G_1)$$

Para obtener las K explicaciones más probables en lugar de la primera, se modifica el algoritmo de forma que cada vez que se marginaliza por máximo en vez de pasar un valor como mensaje se pasa un vector ordenado que contiene los K valores de máxima probabilidad junto con sus $config$ asociadas. Evidentemente esto aumenta la complejidad del algoritmo, ya que si un grupo G_i tiene H grupos hijos y $S_{j_1}, \ldots, S_{j_H}$ son los separadores que comunican G_i con sus grupos hijos entonces, en la búsqueda de la explicación más probable al hacer la operación de combinación se tiene

$$\psi(g_i) \leftarrow \psi(g_i) \cdot \psi(g_i^{\downarrow S_{j_1}}) \cdot \cdots \cdot \psi(g_i^{\downarrow S_{j_H}});$$

es decir, H multiplicaciones para obtener el valor de $\psi(g_i)$. Sin embargo, en la búsqueda de las K explicaciones más probables se tiene

$$\psi(g_i) \leftarrow \psi(g_i) \cdot \psi(g_i^{\downarrow S_{j_1}}[m_1]) \cdot \cdots \cdot \psi(g_i^{\downarrow S_{j_H}}[m_H]), \quad \text{con } m_1, \ldots, m_H \in 1 \ldots K.$$

Es decir, ahora hay que hacer HK^H multiplicaciones, ordenar los K^H valores obtenidos para g_i y quedarnos con los K primeros. No obstante, los autores modifican el método para evitar la complejidad exponencial quedando el número de multiplicaciones necesarias acotado por H^2K. Por simplicidad no describiremos aquí esta modificación.

Ejemplo 3. Sea la red causal de la figura 4.a formada por las variables bivaluadas $\{A, B, C\}$ y la variable D que puede tomar tres estados. Y sean sus probabilidades

[5] Si bien los autores describen el algoritmo como un procedimiento iterativo y no como una propagación propiamente dicha (con paso de mensajes)

condicionadas las mostradas en la tabla 1.a. Supongamos que no hay evidencia observada y que queremos obtener las dos explicaciones más probables. En primer lugar obtenemos el árbol de grupos maximales mostrado en la figura 4.b y sus potenciales iniciales (tabla 1.b).

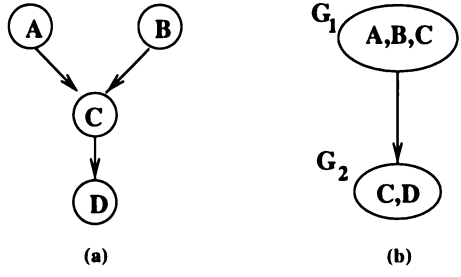


Figura 4. (a) Red causal con cuatro variables. (b) Un árbol de grupos maximales.

El siguiente paso es calcular el mensaje a enviar desde el grupo G_2 hacia el grupo G_1, teniendo en cuenta que como queremos obtener las dos explicaciones más probables hay que mandar un vector de dos posiciones por cada configuración. El mensaje a enviar es el siguiente:

$$M^{G_2 \to G_1} = \begin{cases} \psi[1](c_1) = 0.5 \rightsquigarrow config = (D = d_3) \\ \psi[2](c_1) = 0.4 \rightsquigarrow config = (D = d_2) \\ \psi[1](c_2) = 0.6 \rightsquigarrow config = (D = d_1) \\ \psi[2](c_2) = 0.3 \rightsquigarrow config = (D = d_2) \end{cases}$$

Por último tenemos que combinar la información recibida en el grupo G_1 con el potencial contenido en este grupo. El resultado de esta operación puede verse en la tabla 2.

Por tanto, las dos mejores explicaciones más probables son:

$$p(a_2, b_1, c_2, d_1) = 0.126$$
$$p(a_1, b_1, c_1, d_3) = 0.112$$

□

5 Abducción parcial en redes causales Bayesianas

En la sección anterior se ha caracterizado el problema de la abducción en las redes bayesianas, sin embargo, en ocasiones sólo queremos obtener la configuración

$p(a_1) = 0.4$
$p(a_2) = 0.6$
$p(b_1) = 0.7$
$p(b_2) = 0.3$
$p(c_1\|a_1, b_1) = 0.8$
$p(c_1\|a_1, b_2) = 0.5$
$p(c_1\|a_2, b_1) = 0.5$
$p(c_1\|a_2, b_2) = 0.0$
$p(c_2\|a_1, b_1) = 0.2$
$p(c_2\|a_1, b_2) = 0.5$
$p(c_2\|a_2, b_1) = 0.5$
$p(c_2\|a_2, b_2) = 1.0$
$p(d_1\|c_1) = 0.1$
$p(d_1\|c_2) = 0.6$
$p(d_2\|c_1) = 0.4$
$p(d_2\|c_2) = 0.3$
$p(d_3\|c_1) = 0.5$
$p(d_3\|c_2) = 0.1$

(a)

$\psi(a_1, b_1, c_1) = 0.224$
$\psi(a_1, b_1, c_2) = 0.056$
$\psi(a_1, b_2, c_1) = 0.06$
$\psi(a_1, b_2, c_2) = 0.06$
$\psi(a_2, b_1, c_1) = 0.21$
$\psi(a_2, b_1, c_2) = 0.21$
$\psi(a_2, b_2, c_1) = 0.0$
$\psi(a_2, b_2, c_2) = 0.18$
$\psi(c_1, d_1) = 0.1$
$\psi(c_1, d_2) = 0.4$
$\psi(c_1, d_3) = 0.5$
$\psi(c_2, d_1) = 0.6$
$\psi(c_2, d_2) = 0.3$
$\psi(c_2, d_3) = 0.1$

(b)

Tabla 1. (a) Distribuciones condicionadas. (b) Potenciales iniciales

g_1	$\psi[1](g_1)$	$\psi[2](g_1)$
a_1, b_1, c_1	$0.112 \rightsquigarrow config = (D = d_3)$	$0.0896 \rightsquigarrow config = (D = d_2)$
a_1, b_1, c_2	$0.0336 \rightsquigarrow config = (D = d_1)$	$0.0168 \rightsquigarrow config = (D = d_2)$
a_1, b_2, c_1	$0.03 \rightsquigarrow config = (D = d_3)$	$0.024 \rightsquigarrow config = (D = d_2)$
a_1, b_2, c_2	$0.036 \rightsquigarrow config = (D = d_1)$	$0.018 \rightsquigarrow config = (D = d_2)$
a_2, b_1, c_1	$0.105 \rightsquigarrow config = (D = d_3)$	$0.084 \rightsquigarrow config = (D = d_2)$
a_2, b_1, c_2	$0.126 \rightsquigarrow config = (D = d_1)$	$0.063 \rightsquigarrow config = (D = d_2)$
a_2, b_2, c_1	$0.0 \rightsquigarrow config = (D = d_3)$	$0.0 \rightsquigarrow config = (D = d_2)$
a_2, b_2, c_2	$0.108 \rightsquigarrow config = (D = d_1)$	$0.054 \rightsquigarrow config = (D = d_2)$

Tabla 2. Resultado de la combinación en G_1.

de estados más probable para un subconjunto de variables de la red llamado *conjunto explicación*. A este tipo de problema se le llama *abducción parcial*. La explicación más probable en un problema de abducción parcial se define como:

Definición 11. (explicación más probable (abducción parcial))
Sea $G = (\mathcal{U}, E)$ una red bayesiana y x_O una observación del conjunto de variables $X_O \subset \mathcal{U}$. Sea $X_E \subseteq \mathcal{U}$ el conjunto de variables de interés o *conjunto explicación*. Sea $X_R = \mathcal{U} \setminus X_E$. Decimos que $x_E \in \Omega_{X_E}$ es la explicación más probable (EMP) de x_O si

$$x_E = \arg\max_{X_E} \sum_{X_R} P(X_E, X_R | x_O)$$

□

Veamos un ejemplo de abducción en redes causales aplicado al análisis de fallos en circuitos lógicos.

Ejemplo 4. Vamos a modelar el circuito de la figura 5 con una red causal. En el circuito hay tres grupos diferenciados de variables:

- Tres variables de entrada $\{A, B, C\}$ que tomarán valores 0 o 1 y que vamos a suponer que ambos valores tienen la misma probabilidad (0.5).
- Tres puertas lógicas que tomarán valores c o i, correspondiendo a funcionamiento correcto e incorrecto respectivamente. Suponemos que la probabilidad de fallo para una puerta Y es del 10%, para una O del 5% y para una N del 2%.
- Tres variables $\{D, E, F\}$ que pueden tomar valores 0 o 1. El valor que toman estas variables depende de otros valores, ya que representan estados intermedios y la salida.

Para transformar el circuito en una red causal aplicamos el esquema[6] de la figura 6 y obtenemos la red de la figura 7.

Supongamos ahora que tenemos la observación $C = 0, F = 0$ y buscamos la explicación más probable a estos hechos, el resultado es:

$$(A = 0, B = 0, Y = c, N = c, D = 0, O = i, E = 1) \text{ con probabilidad } 0.173$$

De esta forma diagnosticamos que la explicación más probable a los hechos observados es un fallo de la puerta NO y un funcionamiento correcto de las puertas O e Y. Sin embargo, la probabilidad en que nos basamos para formular este diagnóstico no es muy elevada, ya que sólo es del 0.173. La cuestión que podemos

[6] Como señalan deKleer y Williams [6] al aplicar este esquema de transformación una salida correcta no garantiza el funcionamiento correcto de los componentes

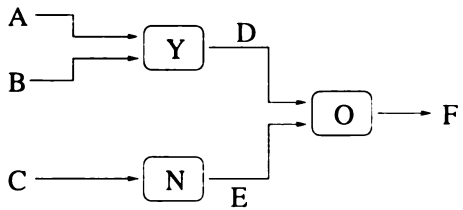
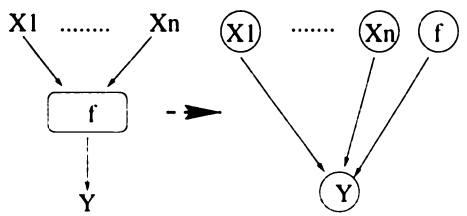


Figura 5. Circuito lógico con tres puertas



$$P(Y|X_0,\ldots,X_n,f) = \begin{cases} 1 & si\ f = correcto\ \text{e}\ Y = f(X_1,\ldots,X_n) \\ 0 & si\ f = correcto\ \text{e}\ Y \neq f(X_1,\ldots,X_n) \\ \frac{1}{|\Omega_Y|} & si\ f = incorrecto \end{cases}$$

Figura 6. Transformación de una puerta lógica.

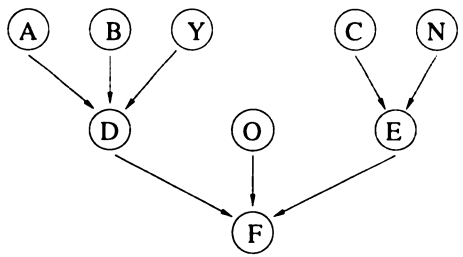


Figura 7. Red causal para el circuito de la figura 5

plantearnos es ¿por qué obtener la configuración de máxima probabilidad para todas las variables si sólo nos interesa conocer si las puertas lógicas funcionan correcta o incorrectamente?. De hecho si planteamos un problema de abducción parcial con $\{Y, N, O\}$ como conjunto explicación obtenemos que la explicación más probable es:

$$(Y = c, N = c, O = i) \text{ con probabilidad } 0.691$$

de donde se obtiene el mismo[7] diagnóstico pero podemos soportarlo con una probabilidad mucho mayor. □

5.1 Obtención de las K explicaciones más probables

Al igual que en la sección dedicada al problema de abducción total, nos centraremos en los métodos basados en realizar la propagación sobre un árbol de grupos maximales. La idea ahora es utilizar un método de propagación híbrido, usando la marginalización por suma para las variables que no pertenecen al conjunto explicación y la marginalización por máximo para las variables del conjunto explicación. Sin embargo, si bien la idea parece sencilla tiene el problema de que no todo árbol de grupos es válido para poder aplicarla, sino que el árbol de grupos debe cumplir ciertas condiciones. El problema viene dado por la no conmutatividad entre la suma y el máximo, que obliga a que no se haga ninguna suma sobre el resultado de un máximo. Por tanto, para poder aplicar los métodos de propagación híbridos se necesita que X_E constituya un subárbol T' de T.

Ejemplo 5. Sea el árbol de grupos de la figura 4.b, entonces si $X_E = \{A, B, C\}$ o $X_E = \{C, D\}$ se dan las condiciones necesarias para aplicar los algoritmos descritos; sin embargo, si $X_E = \{A, B, D\}$ no se dan estas condiciones y, por tanto, los algoritmos no pueden ser aplicados. □

El problema que se presenta ahora es qué hacer cuando X_E no constituye un subárbol del árbol de grupos. Nilsson [13] plantea usar el algorimto de Xu [31] para modificar el árbol de grupos de forma que los algoritmos puedan ser aplicados. De Campos y col. [5,7] describen cómo construir un árbol de grupos que cumpla las condiciones requeridas y proponen un método híbrido basado en el algoritmo de Seroussi y Golmard. Sin embargo, estos autores han estudiado que los árboles de grupos obtenidos tienen un tamaño muy superior a los que se obtienen cuando no hay ninguna restricción, lo que hace que los algoritmos sean más ineficientes que en el caso de la abducción total.

[7] En general esto no tiene por qué ocurrir

6 Conclusiones

En este trabajo hemos comenzado introduciendo el problema de la abducción usando para ello su aplicación más conocida: el diagnóstico. Inicialmente hemos visto cómo se abordaba el problema en algunos de los sistemas expertos desarrollados hace una década. Posteriormente hemos estudiado la formalización del problema propuesta por Peng y Reggia, así como los métodos propuestos por estos autores para resolverlo. Sin embargo, en ambos modelos se hace uso de suposiciones que no permiten aplicar el modelo en cualquier sistema probabilístico, lo que nos ha llevado a plantear el problema en las redes causales.

Ya dentro del formalismo de las redes causales se han planteado dos problemas relativos a la abducción: abducción total (sobre todas las variables) y abducción parcial (sobre un subconjunto distinguido de las variables). El caso de la abducción total ha sido más estudiado y los métodos de búsqueda de la explicación más probable pueden construirse sustituyendo en los métodos de propagación de probabilidades la marginalización mediante suma por la marginalización mediante máximo. La búsqueda de las K explicaciones más probables es más compleja y en este problema es donde se centran las investigaciones actuales. Por otra parte, la abducción parcial ha sido menos estudiada y plantea una serie de restricciones que hace más complicada la resolución del problema.

En cuanto a las líneas de investigación futuras, podemos dividirlas en dos grupos: desarrollo de algoritmos de carácter aproximado para el problema de la abducción parcial; y estudio de métodos que nos permitan obtener explicaciones más simples desde el punto de vista del número de literales incluidos en la explicación ([2,26,27]).

Referencias

1. D.E. Appelt y M. Pollack. Weighted abduction for plan ascription. Technical report, Artificial Intelligence Center and Center for the Study of Language and Information, SRI International, Menlo Park, California, 1990.
2. U. Chajewska y J. Y. Halpern. Defining explanation in probabilistic systems. En: *Proceedings of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence (UAI-97)*, págs. 62–71, San Francisco, CA, 1997. Morgan Kaufmann Publishers.
3. E. Charniak y D. McDermott. *Introduction to Artificial Intelligence.* Addison-Wesley, 1985.
4. A.P. Dawid. Applications of a general propagation algorithm for probabilistic expert systems. *Statistics and Computing*, **2**:25–36, 1992.
5. L.M. De Campos, S. Moral y Gámez J.A. Un método exacto para realizar abducción parcial en redes bayesianas. En: V. Botti (ed.), *Actas de la VII Conferencia de la Asociación Española para la inteligencia artificial*, págs. 621–633, Málaga, 1997.
6. J. deKleer y B.C. Williams. Diagnosing multiple faults. *Artificial Intelligence*, **32**(1):97–130, 1987.
7. J.A. Gámez. *Inferencia abductiva en redes causales.* Tesis doctoral, En preparación.
8. F.V. Jensen. *An introduction to Bayesian Networks.* UCL Press, 1996.
9. U.P. Kumar y U.B. Desai. Image interpretation using bayesian networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **18**(1):74–78, 1996.
10. H. Levesque. A knowledge-level account of abduction. En: *Proceedings of the 11th International Joint Conference on Artificial Intelligence*, 1989.
11. R. Miller, H. Pople y J. Meyers. Internist-1: An experimental computer-based diagnostic consultant for general internal medicine. *New England Journal of Medicine*, **307**:468–476, 1982.
12. R. E. Neapolitan. *Probabilistic Reasoning in Expert Systems. Theory and Algorithms.* Wiley Interscience, New York, 1990.
13. D. Nilsson. An efficient algorithm for finding the m most probable configurations in bayesian networks. Technical Report R-96-2020, Institute for Electronic Systems. Department of Mathematics and Computer Science. University of Aalborg, 1996.
14. J. Pearl. Distributed revision of composite beliefs. *Artificial Intelligence*, **33**:173–215, 1987.
15. C.S. Peirce. *Abduction and Induction.* Dower, 1955.
16. Y. Peng y J.A. Reggia. A probabilistic causal model for diagnostic problem solving. partes I y II. *IEEE Transactions on Systems, Man, and Cybernetics*, **17**(2), 1987.
17. Y. Peng y J.A. Reggia. *Abductive Inference Models for Diagnostic Problem-Solving.* Springer-Verlag, 1990.
18. H.E. Pople. On the mechanization of abductive logic. En: *Proceedings of the 3rd International Joint Conference on Artificial Intelligence*, 1973.
19. H.E. Pople. The formation of composite hypoteses in diagnostic problem solving: An exercise in synthetic reasoning. En: *Proc. of IJCAI*, págs. 1030–1037, 1977.

20. H.E. Pople. *Artificial Intelligence in Medicine*, cap. Heuristic Methods for improving structure on ill-structured problems: The structuring of medical diagnosis, págs. 119–190. ., 1982.
21. J. Reggia. Knowledge-based decision support system: Development through kms. Technical Report TR-1136, Dept. of Computer Science, University of Maryland, 1982.
22. J.A. Reggia. Diagnostic expert systems based on a set covering model. *International Journal of Man-Machine Studies*, **83**, 1988.
23. R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, **32**, 1987.
24. B. Seroussi y J.L. Goldmard. An algorithm directly finding the k most probable configurations in bayesian networks. *International Journal of Approximate Reasoning*, **11**:205–233, 1994.
25. P.P. Shenoy y G.R. Shafer. Axioms for probability and belief-function propagation. En: R.D. Shachter, T.S. Levitt, L.N. Kanal y J.F. Lemmer (eds.), *Uncertainty in Artificial Intelligence, 4.*, págs. 169–198. Elsevier Science Publishers B.V. (North-Holland), 1990.
26. S.E. Shimony. The role of relevance in explanation i: Irrelevance as statistical independence. *International Journal of Approximate Reasoning*, **8**:281–324, 1993.
27. S.E. Shimony. The role of relevance in explanation ii: Disjunctive assignments and approximate independence. *International Journal of Approximate Reasoning*, **13**:27–60, 1995.
28. S.E. Shimony y E. Charniak. A new algorithm for finding map assignments to belief networks. En: *Proceedings of the 6th Conference on Uncertainty in Artificial Intelligence*, Cambridge, MA, 1990.
29. H. Shubin y J. Ulrich. Idt: An intelligent diagnostic tool. En: *Proc. of National Conference on Artificial Intelligence (AAAI)*, págs. 290–295, 1982.
30. M.E. Stickel. A prolog-like inference system for computing minimum-cost abductive explanations in natural language interpretation. Technical Report 451, AI Center, SRI International, 1988.
31. H. Xu. Computing marginals for arbitrary subsets from marginal representation in markov trees. *Artificial Intelligence*, **74**:177–189, 1995.

Aprendizaje Automático de Modelos Gráficos I: Métodos Básicos

Luis M. de Campos

Departamento de Ciencias de la Computación e I.A.
Universidad de Granada
Granada. 18071
correo-e: lci@decsai.ugr.es

Resumen

El objetivo de este trabajo es dar una introducción al problema del aprendizaje automático de modelos gráficos, poniendo de relieve los conceptos más importantes y las ideas generales de los distintos métodos propuestos. También se consideran algunos métodos específicos de aprendizaje, con el propósito de ilustrar de forma más detallada las distintas metodologías. Después de destacar la importancia creciente que van adquiriendo las técnicas de aprendizaje automático en general, nuestro estudio se restringe al aprendizaje de un tipo concreto (pero posiblemente el más conocido y empleado) de modelos gráficos: las redes de creencia. Se consideran las dos tareas básicas que necesita realizar cualquier algoritmo de aprendizaje de redes de creencia: aprendizaje de la estructura gráfica y aprendizaje de los parámetros numéricos correspondientes, centrándonos principalmente en la primera de ellas.

1 Introducción

Los métodos de aprendizaje automático (*Machine Learning* [58]) han jugado un papel central en la Inteligencia Artificial desde sus comienzos. Probablemente, esto se debe a que la habilidad para aprender, adaptarse y modificar el comportamiento es un componente fundamental de la inteligencia humana, y por tanto ¿cómo podría entonces decirse de una máquina que es inteligente, si no es capaz de mejorar su funcionamiento?

En la mayoría de los campos de conocimiento, el volumen de datos que se pueden recoger y almacenar está aumentando a gran velocidad, gracias al desarrollo de las tecnologías de la información. Surge entonces la necesidad de disponer de herramientas computacionales capaces de asistir a las personas en la tarea de extraer información útil (conocimiento) a partir de esas ingentes cantidades de datos. Se trata pues de desarrollar métodos capaces de extraer sentido de los datos.

El problema básico es el de poder relacionar datos de bajo nivel (datos 'en bruto'), que habitualmente son demasiado numerosos para comprenderlos y asimilarlos fácilmente, con otras formas de representación que puedan ser más compactas (pero conserven lo esencial de los datos), más abstractas (un modelo que describa el proceso que generó los datos) y más útiles (un modelo que pueda servir para predecir o estimar los valores de interés en situaciones o casos aún no observados).

Por ejemplo, imaginemos una base de datos que contenga diferentes tipos de información sobre pacientes (sexo, edad, síntomas, resultados de exploraciones y pruebas de laboratorio, patologías que padece,...). Esa base de datos contiene gran cantidad de información en forma latente, pero no contiene 'conocimiento'. En primer lugar, la información que contiene puede ser demasiado voluminosa para ser manejable por una persona, de ahí que resulte importante 'condensarla' (algo así como separar la paja del grano, o destilar la esencia de una flor o planta). En segundo lugar, aunque la información está allí, no constituye conocimiento porque el conocimiento significa 'comprender' (saber los mecanismos que gobiernan las cosas, las relaciones entre ellas, etc); en definitiva, comprender implica disponer de un modelo que describa cómo funciona el fenómeno que se está estudiando. En nuestro ejemplo, un modelo describiría cuáles son las relaciones entre síntomas y enfermedades, qué enfermedades causan qué síntomas, etc. Finalmente, la información contenida en la base de datos no es útil tal cual: si queremos diagnosticar cuál es la enfermedad que padece un nuevo paciente en función de sus características y sintomatología, no podemos consultar la base de datos para obtener la respuesta, necesitamos que el modelo descriptivo antes mencionado también sea predictivo, pueda ser manipulado para, en función del conocimiento general que el modelo describe y del conocimineto específico sobre ese paciente en particular, podamos inferir la respuesta.

El método tradicional de transformar datos en conocimiento se basa en análisis manuales de los mismos y su interpretación, es decir, uno o varios analistas profundamente familiarizados con un tipo de datos (expertos) sirven como interfaz entre los datos y los usuarios. Esta forma de trabajo es lenta, cara y muy subjetiva. Es bien conocido que el principal cuello de botella en el proceso de construcción de sistemas expertos es este método tradicional de elicitación del conocimiento mediante la interacción entre los expertos y el ingeniero de conocimiento. De hecho, cuando el volumen de datos aumenta de forma importante, este tipo de análisis manual está resultando completamente impracticable en muchos casos. Por tanto resulta necesario automatizar, aunque sea parcialmente, este trabajo de análisis.

En muchos aspectos, el proceso de obtención de conocimiento tiene un componente fundamentalmente estadístico. La estadística proporciona un lenguaje para cuantificar la incertidumbre que aparece cuando se tratan de inferir patrones genéricos a partir de una muestra (una fracción) de una población. En

nuestro caso, la base de datos a partir de la cual queremos aprender un modelo de representación y predicción apropiado, constituye la muestra de una población posiblemente mucho mayor: no es casi nunca previsible que tengamos datos suficientes de todas las situaciones posibles; por ejemplo, una base de datos de pacientes, por muy grande que pueda ser, sólo contiene información de los pacientes realmente tratados, no de todos los hipotéticos pacientes que pudieran atenderse alguna vez (que constituirían la población completa).

El objetivo de este trabajo es dar una panorámica de las diversas técnicas existentes para el aprendizaje automático de modelos gráficos a partir de datos. Como es ya sabido, los modelos gráficos son herramientas de representación del conocimiento capaces de representar relaciones de dependencia/independencia así como incertidumbre en nuestro conocimiento, y constan de un componente cualitativo y otro cuantitativo. El componente cualitativo es un grafo (dirigido o no, o incluso un hipergrafo) que representa relaciones de dependencia e independencia: la ausencia de algún enlace significa la existencia de ciertas relaciones de independencia entre variables, y la presencia de enlaces puede representar la existencia de relaciones de dependencia directa. El componente cuantitativo es una colección de parámetros numéricos, que dan idea de la fuerza de las dependencias y miden nuestra incertidumbre.

Concretamente, nos vamos a referir casi exclusivamente a un tipo específico de modelos gráficos: las *redes de creencia*. El modelo cualitativo es en este caso un grafo dirigido y acíclico (si se pretende dar una interpretación causal a las direcciones de los arcos del grafo, se suele utilizar el nombre de *redes causales*); el modelo cuantitativo es una colección de distribuciones condicionadas de un nodo dados sus padres en el grafo, siendo lo más frecuente emplear distribuciones de probabilidad (si se desea enfatizar este hecho entonces es habitual emplear el término *redes bayesianas*).

El interés en los métodos de aprendizaje de redes de creencia (y de modelos gráficos en general) es el resultado de la unión entre las técnicas de aprendizaje automático desarrolladas dentro de la Inteligencia Artificial (originalmente centradas en el aprendizaje de sistemas basados en reglas), los métodos estadísticos clásicos de análisis de datos, y la cada vez mayor apreciación de las redes de creencia como un formalismo de representación del conocimiento con incertidumbre coherente y potente.

El trabajo se articula en las siguientes secciones: la sección 2 plantea de forma muy breve las dos tareas básicas a realizar para el aprendizaje de redes de creencia. La sección 3 estudia el problema del aprendizaje de los parámetros numéricos de la red, mientras que la sección 4 considera el problema del aprendizaje de la estructura de la red. En los diferentes apartados dentro de esta sección se consideran las distintas metodologías generales existentes para abordar este problema,

y se estudia un algoritmo representativo de cada una de ellas. Las referencias bibliográficas son abundantes (aunque en ningún modo son exhaustivas), y atestiguan el creciente interés y popularidad que el aprendizaje de modelos gráficos suscita. También se pretende que sirvan de guía para los lectores interesados en el tema.

2 Generalidades sobre Aprendizaje de Redes de Creencia

Puesto que las redes de creencia constan de dos componentes diferentes (pero estrechamente relacionados), el modelo gráfico y el modelo numérico, los algoritmos de aprendizaje automático de redes de creencia necesariamente tienen que realizar dos tareas bien diferenciadas, aunque altamente dependientes entre sí:

- El aprendizaje de la estructura gráfica (un grafo dirigido y acíclico).
- El aprendizaje de los parámetros numéricos (las distribuciones de probabilidad de cada nodo del grafo condicionadas a los posibles valores de sus nodos padres).

Estas dos tareas no se pueden realizar de forma completamente independiente. Por un lado, para poder aprender o estimar las distribuciones de probabilidad condicionadas que contendrá la red, es preciso primero conocer cuál es la estructura gráfica: sólamente cuando conozcamos, por ejemplo, que el grafo contiene los arcos $X \rightarrow Z$ e $Y \rightarrow Z$, es cuando sabemos que tenemos que calcular los valores $P(Z = z|X = x, Y = y)$ de las distribuciones de probabilidad de Z dados X e Y. Por otro lado, para poder determinar si el grafo que intentamos encontrar contiene o no determinados arcos, tendremos que realizar algún test de independencia condicional entre los nodos extremos de esos arcos, o calcular de algún modo la intensidad de la relación entre tales nodos (esto dependerá en gran medida del tipo de algoritmo de aprendizaje de la estructura que se emplee, como veremos más adelante), pero en cualquier caso tendremos que estimar ciertas distribuciones de probabilidad a partir de los datos disponibles.

En este trabajo nos vamos a centrar fundamentalmente en los métodos de aprendizaje de la estructura de la red de creencia, pero comentaremos en la siguiente sección algunas ideas respecto al aprendizaje de los parámetros. Excelentes trabajos introductorios al tema del aprendizaje de redes se pueden encontrar en [14,41,47].

3 Aprendizaje de los Parámetros de una Red de Creencia

El problema que se plantea aquí es, dado un grafo dirigido y acíclico G, que representa la estructura de una red de creencia, y una base de datos que contiene

datos de las variables asociadas a los nodos de la red [1], determinar los parámetros numéricos de la red de creencia a partir de la base de datos. Más concretamente, la topología del grafo determina, para cada nodo X_i, el conjunto de padres de dicho nodo, $\Pi_G(X_i)$; entonces la distribución de probabilidad conjunta de todos los nodos se factoriza de la siguiente forma:

$$P(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | \pi_G(x_i))$$

donde x_i representa un valor de la variable X_i y $\pi_G(x_i)$ representa una asignación de valores a cada una de las variables del conjunto $\Pi_G(X_i)$. Entonces el problema consiste en estimar los valores de las distribuciones de probabilidad condicionadas $P(x_i|\pi_G(x_i))$ a partir de los datos disponibles.

Para ilustrar las ideas que se expondrán en este apartado, supongamos por ejemplo que disponemos de la base de datos, mostrada en la tabla 1, que contiene 6 casos para un problema con 4 variables binarias.

Caso	X_1	X_2	X_3	X_4
1	1	0	0	1
2	0	1	0	1
3	0	1	1	0
4	0	1	0	0
5	1	0	0	1
6	1	1	0	1

Tabla 1. Una sencilla base de datos para cuatro variables binarias.

Supongamos también que el grafo que queremos completar, estimando las distribuciones de probabilidad apropiadas, es el mostrado en la figura 1.

La forma más simple, y muy habitual de estimar las distribuciones de probabilidad es mediante el cálculo de las frecuencias relativas de ocurrencia de los correspondientes sucesos. Así, si por ejemplo queremos calcular la probabilidad $P(X_2 = 0|X_1 = 1)$, las frecuencias relativas son:

$$P(X_2 = 0|X_1 = 1) = \frac{P(X_2 = 0, X_1 = 1)}{P(X_1 = 1)} = \frac{2}{3}$$

[1] A partir de ahora hablaremos indistintamente de las variables del problema y de los nodos de la red.

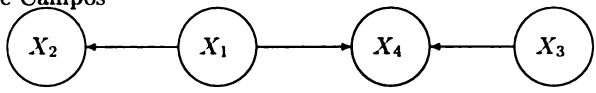


Figura 1. Grafo con cuatro nodos.

En el caso general, si $n(\pi_G(x_i))$ y $n(x_i, \pi_G(x_i))$ representan respectivamente el número de casos de la base de datos en que las variables de $\Pi_G(X_i)$ toman los valores $\pi_G(x_i)$ y en que las variables X_i y $\Pi_G(X_i)$ toman simultáneamente los valores x_i y $\pi_G(x_i)$, entonces el valor estimado de probabilidad es

$$P(x_i|\pi_G(x_i)) = \frac{n(x_i, \pi_G(x_i))}{n(\pi_G(x_i))}$$

En términos más formales, este método se corresponde con utilizar un estimador de máxima verosimilitud [14].

Los métodos de máxima verosimilitud presentan dos problemas: datos dispersos y sobreajuste. El primero se debe a que el estimador puede no estar definido si el número de datos de cierto tipo es cero. Por ejemplo, si queremos estimar la probabilidad $P(X_4 = 0|X_1 = 1, X_3 = 1)$, al no existir en la base de datos casos en los que $X_1 = 1$ y simultáneamente $X_3 = 1$, el estimador de máxima verosimilitud no está bien definido. En un grafo con un nodo que tenga k padres (todos binarios), necesitaríamos un mínimo de 2^k casos para que el estimador tuviera alguna posibilidad de estar definido. Por otro lado, el problema del sobreajuste es el siguiente: si por ejemplo calculamos el estimador de máxima verosimilitud de la probabilidad $P(X_2 = 1|X_1 = 0)$, obtenemos un valor de 1.0, puesto que en la base de datos todos los casos en que X_1 es 0 (que son tres) tienen un valor 1 para la variable X_2. Este estimador está basado en tan sólo tres casos. Podría ser razonable pensar que el 'verdadero' valor de esa probabilidad fuese por ejemplo 0.9 en lugar de 1.0, pero por el azar no hemos observado casos en que $X_1 = 0$ y $X_2 = 0$. El estimador de máxima verosimilitud tiende a sobreajustarse a los datos. Cuando el tamaño muestral es bastante grande, este estimador tiende al valor verdadero; sin embargo para muestras pequeñas la diferencia puede ser considerable. Esto se debe a que el estimador de máxima verosimilitud se ajusta demasiado a los datos disponibles (tiene poca capacidad de generalización; es como si para ajustar, mediante regresión, un polinomio de 10 pares de puntos en el plano usásemos un polinomio de grado 9: el ajuste será perfecto, pero posiblemente fuese mucho más razonable emplear un polinomio de grado 2 o 3, y suponer que los pequeños errores de ajuste se deben a ruido de datos).

Existen otros métodos de estimación o aprendizaje de probabilidades que intentan paliar esos problemas. Uno de ellos está basado en lo que se llama la *ley*

de la sucesión de Laplace [38]: esta ley dice que si en una muestra de N casos encontramos k casos que verifican una determinada propiedad Q (por ejemplo que el valor de la variable X_i es igual a x_i), entonces la probabilidad de que el siguiente caso que observemos exhiba la misma propiedad es $(k+1)/(N+|Q|)$, donde $|Q|$ representa el número de alternativas posibles que se consideran para la propiedad Q (por ejemplo el número de distintos valores posibles que la variable X_i puede tomar).

En nuestro caso, para estimar la probabilidad $P(x_i|\pi_G(x_i))$ con este método, obtendríamos el valor

$$P(x_i|\pi_G(x_i)) = \frac{n(x_i, \pi_G(x_i)) + 1}{n(\pi_G(x_i)) + |X_i|}$$

donde $|X_i|$ es el número de distintos valores posibles que la variable X_i puede tomar. Obsérvese que si la muestra es muy grande, las diferencias con respecto al estimador de máxima verosimilitud serán muy pequeñas, y cuando la muestra sea pequeña, la distribución tiende a parecerse a una distribución uniforme (en el caso extremo en que $n(\pi_G(x_i))$ sea cero, el resultado es exactamente la distribución uniforme). En nuestro ejemplo, la estimación resultante de aplicar este método a las distribuciones antes empleadas es:

$$P(X_2 = 0|X_1 = 1) = \frac{2+1}{3+2} = 0.60$$
$$P(X_4 = 0|X_1 = 1, X_3 = 1) = \frac{0+1}{0+2} = 0.5$$
$$P(X_2 = 1|X_1 = 0) = \frac{3+1}{3+2} = 0.8$$

Este método de estimación es realmente un método bayesiano (se parte de cierta información a priori, y se actualiza dicha información a la luz de nuevos datos empleando la fórmula de Bayes), en el que la distribución a priori es uniforme. Se puede considerar como un caso particular de un método bayesiano de estimación más general, basado en distribuciones de Dirichlet [41,42] (que son generalizaciones de las distribuciones beta, que a su vez incluyen a la distribución uniforme como caso particular).

Sin entrar en detalles técnicos, vamos a exponer muy brevemente el resultado de este método bayesiano más general, que también se denomina a veces de *m-estimación* [20,21]. Suponiendo que nuestras distribuciones a priori son Dirichlet, el m-estimador para el valor de probabilidad $P(x_i|\pi_G(x_i))$ es:

$$P(x_i|\pi_G(x_i)) = \frac{n(x_i, \pi_G(x_i)) + s\frac{n(x_i)}{N}}{n(\pi_G(x_i)) + s}$$

donde s es un parámetro, que se suele interpretar en términos de *tamaño muestral equivalente* (es como si la distribución a priori se hubiese estimado a partir de una muestra de tamaño s), y N es el número total de datos. Una formulación equivalente, pero quizás algo más intuitiva, es la siguiente [31,32]:

$$P(x_i|\pi_G(x_i)) = \alpha \frac{n(x_i, \pi_G(x_i))}{n(\pi_G(x_i))} + (1-\alpha)\frac{n(x_i)}{N}$$

donde $\alpha = \frac{n(\pi_G(x_i))}{n(\pi_G(x_i))+s}$. En otras palabras, el estimador es el resultado de realizar una combinación convexa entre la probabilidad condicional de X_i dados sus padres y la distribución marginal de X_i, ambas obtenidas por máxima verosimilitud (frecuencias relativas). Continuando con nuestro ejemplo, los resultados de aplicar este método de m-estimación a las distribuciones anteriores son (suponiendo que $s = 5$):

$$P(X_2 = 0|X_1 = 1) = 0.458$$
$$P(X_4 = 0|X_1 = 1, X_3 = 1) = 0.333$$
$$P(X_2 = 1|X_1 = 0) = 0.792$$

En la discusión anterior hemos supuesto que conocíamos los valores de todas las variables en todos los casos de la base de datos. Es frecuente que esto no ocurra: se habla de *variables perdidas* cuando para algunos casos de la base de datos desconocemos el valor de alguna o algunas variables (no se registró su valor por alguna razón). También puede ocurrir que algunas variables no puedan ser observadas nunca, y en este caso hablamos de *variables latentes*. Existen métodos para tratar estas situaciones, algunos exactos [26,42,69] pero altamente costosos, y otros aproximados, entre los que hay métodos determinísticos y estocásticos (Monte-Carlo) [13,28,29,39,46,52,56,69].

También podemos distinguir entre la construcción de un modelo inicial y la revisión de los parámetros en un modelo ya existente. Al primer caso se le suele llamar *entrenamiento*: disponemos de todos los datos desde el principio, y los empleamos para estimar las probabilides necesarias. El segundo caso se denomina *adaptación* [60]: sobre un grafo ya existente y unas distribuciones de probabilidad asociadas (extraídas de datos previos o de opiniones de expertos), se desea usar los nuevos datos que puedan ir apareciendo para revisar las probabilidades. Diversas técnicas para tratar este problema de la adaptación, principalmente basadas también en el uso de distribuciones de Dirichlet, pueden encontrarse en [69–71].

4 Aprendizaje de la Estructura de una Red de Creencia

En esta sección analizaremos las técnicas existentes para el aprendizaje de la estructura de una red de creencia. Aunque este es un problema relativamente

nuevo (en el sentido de que ha comenzado a ser estudiado hace pocos años: la inmensa mayoría de los trabajos de investigación al respecto se han publicado en esta década), hay ya una gran cantidad de algoritmos para resolverlo. No obstante, casi todos esos algoritmos están esencialmente basados en dos únicos enfoques (con múltiples variaciones), que pueden aplicarse al aprendizaje de distintos tipos de grafos. En todo caso, la idea común a todos ellos es efectuar una búsqueda (que en la mayoría de los casos es explícita, pero en algunos casos es implícita) en un espacio de posibles soluciones (el formado por todos los grafos del tipo deseado), y tratan de encontrar una solución óptima o aproximadamente óptima desde algún punto de vista.

Antes de explicar los dos enfoques básicos, comentaremos brevemente algunos tipos específicos de grafos que, por diversas razones, resultan interesantes. En general, a mayor complejidad del tipo de estructura que se desea utilizar, mayor es también la complejidad de los algoritmos de aprendizaje.

En primer lugar, y aunque no dan lugar a redes de creencia (pero guardan una estrecha relación con éstas), existen algoritmos para aprender grafos no dirigidos o *redes Markovianas* [8,13,17,81,83,84], particularmente los llamados grafos *cordales* (que son grafos en los que todo ciclo de longitud cuatro o más tiene una cuerda, es decir, una arista uniendo dos nodos no adyacentes en el ciclo). Los grafos cordales son importantes por diversas razones: constituyen la clase de modelos que puede representarse tanto mediante grafos dirigidos como no dirigidos [15,61]; también poseen propiedades muy útiles relativas a factorización y estimación de parámetros [82]. La figura 2 representa dos grafos no dirigidos, uno cordal y el otro no.

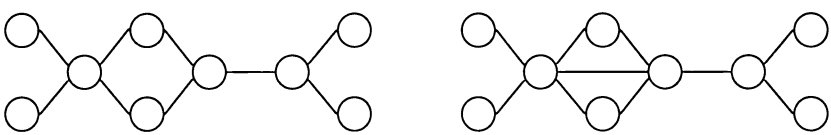


Figura 2. Grafo no dirigido (izqda.) y Grafo no dirigido cordal (dcha.)

Dentro ya de las redes de creencia, los tipos de grafos más sencillos son las redes simplemente conectadas: *poliárboles* (que incluyen a los árboles como caso particular). Los poliárboles son grafos en los que no existe más de un camino (no dirigido) que conecte cualesquiera dos nodos; en otras palabras, grafos que no tienen ningún ciclo no dirigido. La figura 3 representa un poliárbol (que fue aprendido por el algoritmo PA [16] a partir de la base de datos Alarm [9]). Sólamente para poliárboles son válidos los procedimientos de propagación puramente locales

[61]. De hecho, varios métodos de propagación para redes generales (condicionamiento y agrupamiento de variables) están basados en la idea de transformar el grafo y convertirlo en simplemente conectado [53,61]. Además, su estructura sencilla permite el aprendizaje de poliárboles de una forma mucho más eficiente que en el caso general. El precio que hay que pagar por estas ventajas es una pérdida de capacidad expresiva, puesto que el tipo de relaciones de independencia que pueden representarse es mucho más restringido en el caso de poliárboles que en el caso de redes generales (múltiplemente conectadas) [16]. Así pues, el aprendizaje de redes simplemente conectadas han sido objeto de mucho interés, desde distintos puntos de vista: causalidad [16,36,45,64], clasificación [31,34], compresión de datos [25], modelos aproximados [1-3,16,66].

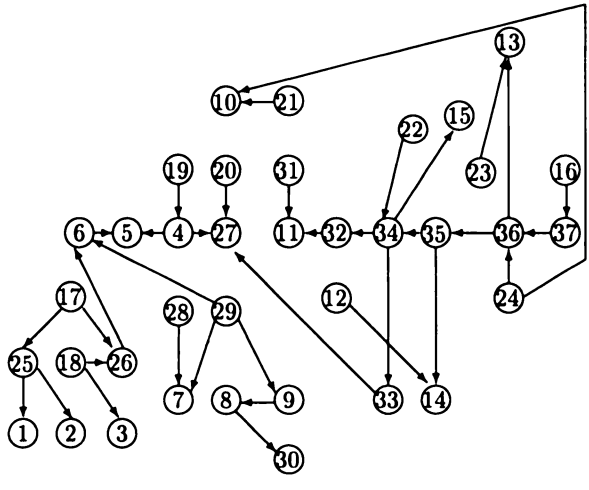


Figura 3. Poliárbol con 37 nodos.

Otro tipo especial de redes de creencia, más general que los poliárboles, a los que incluyen, son los *grafos simples*. Son grafos dirigidos acíclicos donde cada par de nodos con un hijo común no tienen antecesores comunes ni uno es antecesor del otro. Esto significa que en un grafo simple sólo están permitidos un tipo especial de ciclos no dirigidos: los que contienen al menos dos nodos cabeza-cabeza (ver figura 4). Los grafos simples permiten representar un conjunto más amplio de relaciones de independencia que los poliárboles, admiten métodos de inferencia

(propagación) más eficientes [40] y pueden también ser aprendidos de forma más eficiente que las redes de tipo general [18,37].

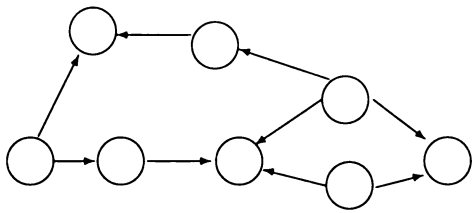


Figura 4. Grafo simple.

Comentaremos ahora los dos enfoques básicos comunmente utilizados para diseñar algoritmos de aprendizaje de redes de creencia:

- Métodos basados en funciones de evaluación y técnicas de búsqueda heurística.
- Métodos basados en detección de independencias.

En el primer tipo de métodos la idea es encontrar un grafo que, teniendo el menor número de arcos posible, represente 'adecuadamente' los datos. El grado de adecuación del grafo a los datos, es decir, la 'calidad' de cada red candidata, se cuantifica mediante algún tipo de medida (la función de evaluación, ajuste, puntuación o métrica). Esta medida es utilizada por algún procedimiento de búsqueda (implícito o explícito), habitualmente heurística (debido al tamaño más que exponencial del espacio de búsqueda), que vaya explorando el espacio de posibles soluciones, tratando de seleccionar la mejor, desde el punto de vista de la métrica empleada. Cada algoritmo de esta clase se caracteriza por el tipo de métrica y de búsqueda específicas que utiliza.

Por otra parte, el objetivo inmediato de los métodos basados en tests de independencia condicional no es encontrar una red que cuantitativamente se ajuste muy bien a los datos disponibles (según alguna métrica), sino que intentan realizar un estudio cualitativo de las relaciones de independencia existentes en el dominio (obviamente a través de los datos), y entonces tratan de encontrar una red que represente esas relaciones de independencia. Así, los datos de entrada básicos que emplean estos métodos son las relaciones de independencia condicional entre conjuntos de variables del modelo, y la salida es un grafo que representa la mayor parte de esas relaciones (o todas, si es posible). Después se estiman las diferentes

distribuciones condicionales de probabilidad para cada variable a partir de la base de datos o de un experto.

También existen enfoques híbridos, que utilizan de forma conjunta una técnica de búsqueda orientada por una métrica y la detección de independencias.

En los siguientes apartados comentaremos con más detalle las características generales de estos métodos, los algoritmos a que dan lugar y, a título de ejemplo, describiremos un algoritmo representativo de cada metodología.

4.1 Aprendizaje de la Estructura mediante la Detección de Independencias

Como ya hemos comentado, estos métodos tratan de determinar la estructura del grafo comprobando si son ciertas o no algunas relaciones de independencia condicional entre variables. Los algoritmos de este tipo pueden tener como información de entrada:

- Una lista de relaciones de independencia condicional que se conoce que son ciertas (ver figura 5),
- Una distribucion de probabilidad P sobre la que se comprueban las relaciones de independencia (figura 6),
- Una base de datos sobre la que se estima directamente la veracidad o no de las relaciones de independencia mediante tests estadísticos de independencia condicional (figura 7).

$$
\begin{array}{|c|}
\hline
I(X_2, X_3|X_1) \\
I(X_1, X_4|\{X_2, X_3\}) \\
I(X_2, X_5|X_4) \\
I(X_3, X_5|X_4) \\
I(X_1, X_5|\{X_2, X_3, X_4\}) \\
\ldots\ldots \\
\hline
\end{array}
$$

Figura 5. Lista de relaciones de independencia condicional.

Desde un punto de vista formal, no hay diferencias en cuanto al tipo de información de entrada, pero existen diferencias muy importantes de tipo práctico, en cuanto:

- Al costo de efectuar los tests (complejidad): detectar independencias condicionales de orden elevado es computacionalmente costoso (el orden de un test

X_1	X_2	X_3	X_4	X_5	$P(x_1, x_2, x_3, x_4, x_5)$
0	0	0	0	0	0.12
0	0	0	0	1	0.05
0	0	0	1	0	0.0
0	0	0	1	1	0.03
...	...	...	...	...	...
1	1	1	1	1	0.2

Figura 6. Distribución de probabilidad conjunta.

X_1	X_2	X_3	X_4	X_5
0	1	1	0	0
1	1	0	1	0
1	0	1	1	1
0	1	1	0	1
1	0	0	0	0
1	0	1	1	1
...	...	...	...	...

Figura 7. Base de datos.

hace referencia al número de variables que intervienen en el conjunto al que se condiciona).
- A la fiabilidad del resultado de los tests (robustez): la detección fiable de independencias de orden elevado requiere gran número de datos.

Muchos de estos métodos requieren ademas información adicional: una ordenación total (o a veces parcial) de las variables, condiciones de isomorfía, etcétera. Existen también diferencias entre estos algoritmos en cuanto a:

- El tipo de grafo que recuperan.
- Su eficiencia:
  - número de independencias que hay que chequear,
  - el orden de estas independencias.
- Su garantía de solución.
- Su robustez frente a errores de muestreo.

Existen muy diversos algoritmos de aprendizaje basados en detección de independencias. Algunos recuperan árboles [16,36], poliárboles [16,45], y otros grafos simples [18,37]. De entre los algoritmos que recuperan grafos dirigidos acíclicos sin restricciones, destacamos los desarrollados por Spirtes y colaboradores [72–74]

(uno de los cuales, el algoritmo PC, nos servirá para ejemplificar estos métodos), y los propuestos por Pearl y sus colaboradores [62,77–79]. Existen por supuesto otros trabajos muy interesantes al respecto [12,19,22,57,75,81].

Un Método de Aprendizaje de la Estructura mediante la Detección de Independencias: El Algoritmo PC El algoritmo que vamos a describir [73,74] es uno de los más conocidos y utilizados de los que emplean el método de detección de independencias. El algoritmo PC presupone que el modelo que se pretende recuperar es isomorfo a un grafo dirigido acíclico (esto quiere decir que todas las relaciones de independencia condicional del modelo se corresponden con relaciones de independencia gráfica o d-separación [61] en el grafo correspondiente y viceversa). Bajo esta condición el algoritmo garantiza encontrar el verdadero grafo subyacente (siempre y cuando no se cometan errores al realizar los tests de independencia condicional requeridos). Los resultados básicos que justifican el algoritmo son los siguientes:

(i) En un grafo dirigido acíclico G, $X—Y \notin G$ si y solo si $\exists S \subseteq Ady_G(X,Y)$ (vértices adyacentes a X o a Y) tal que X e Y estan d-separados por S.

(ii) En un grafo dirigido acíclico G, si $X—Y$, $Y—Z \in G$, pero $X—Z \notin G$, entonces o bien Y está en cualquier subconjunto de variables que d-separa X y Z, o no está en ningún subconjunto de variables que d-separa X y Z.

(iii) En un grafo dirigido acíclico G, si $X—Y$, $Y—Z \in G$, pero $X—Z \notin G$, entonces $X \rightarrow Y \leftarrow Z \in G$ si y solo si Y no está en ningún subconjunto de variables que d-separa X y Z

Como la mayoría de los algoritmos de este tipo, PC comienza formando el grafo completo no dirigido. Entonces va reduciéndolo: primero eliminando las aristas que unen nodos que verifican una independencia condicional de orden cero, despues las que unen nodos que satisfacen una independencia condicional de orden uno, y así sucesivamente. El conjunto de nodos candidatos a formar parte del conjunto separador (el conjunto al que se condiciona) es el de los nodos (todavía) adyacentes a alguno de los nodos que se pretenden separar (nótese que este conjunto de nodos adyacentes está continuamente cambiando conforme el algoritmo progresa). Como todos los algoritmos que recuperan grafos generales, en el peor caso la complejidad de PC es exponencial, aunque es razonablemente eficiente para aprender grafos poco densos. El algoritmo PC se detalla en la figura 8.

1. Formar el grafo completo no dirigido G.
2. $n = 0$
3. repetir
   repetir
   (a) Seleccionar un par de vértices X, Y adyacentes en G tales que $|Ady_G(X, Y)| \geq n$, y seleccionar un subconjunto $S(X, Y) \subseteq Ady_G(X, Y)$ de cardinal igual a n.
   (b) Si $I(X, Y|S(X, Y))$, eliminar X—Y de G, y guardar $S(X, Y)$.
   hasta que todos los pares X, Y hayan sido comprobados.
   $n = n + 1$.
   hasta que para cada par de vértices adyacentes X, Y, $|Ady_G(X, Y)| < n$.
4. Sea G el grafo resultante de los pasos anteriores. Para cada terna X, Y, Z tal que X—Y—$Z \in G$, pero X—$Z \notin G$, orientar como $X \rightarrow Y \leftarrow Z$ si y solo si $Y \notin S(X, Z)$.

Figura 8. El algoritmo PC.

4.2 Aprendizaje de la Estructura usando Métricas y Técnicas de Búsqueda

Los métodos de este tipo usan técnicas de búsqueda para ir obteniendo modelos (redes candidatas), que son entonces evaluados mediante una métrica. Todos los algoritmos emplean algún tipo de búsqueda heurística, la mayoría búsquedas de tipo 'ávido' (*greedy*), debido al tamaño super-exponencial del espacio de búsqueda. En cambio, el tipo de métrica que emplean es muy variado, aunque basado en unos pocos principios diferentes: entropía, ideas bayesianas y descripción de longitud mínima, principalmente.

Entropía Los métodos basados en entropía tratan de encontrar la red cuya entropía cruzada con los datos sea mínima. La entropía se puede considerar como una forma de medir el grado de dependencia entre variables, y en ese sentido estos métodos lo que hacen es buscar configuraciones que favorezcan la presencia de conexiones entre variables que manifiesten un alto grado de dependencia. De alguna manera, se reemplaza la dicotomía de dependencia/independencia de los métodos basados en detección de independencias por una idea gradual de dependencia (en la que la independencia no es más que dependencia a grado mínimo (cero) y la máxima dependencia corresponde a dependencia funcional: conocido el valor de una variable, se puede conocer con certeza el valor de la otra).

Entre estos métodos, los hay que aprenden estructuras sencillas, como árboles [16,25,34,66] y poliárboles [1–3,16,59,64]. En esos casos, debido a las características peculiares de estas estructuras, es posible reemplazar el proceso de optimización mediante una búsqueda explícita por un proceso analítico (una búsqueda implícita), lo que da lugar a algoritmos mucho más eficientes. Para el caso de redes cualesquiera también existen algoritmos de este tipo [44].

Descripción de Longitud Mínima Este principio [65] establece que la mejor representación de un conjunto de datos es aquélla que minimiza la suma de las longitudes de codificación del modelo y de los datos dado el modelo. Normalmente, elegir un modelo muy complejo conllevará una longitud de codificación del mismo elevada (es como si para representar 101 puntos (x_i, y_i) de un plano, y utilizando como tipo de modelo un polinomio $p(x)$, se emplease un polinomio de grado 100). Por otro lado, un modelo complejo también resultará exacto o muy aproximado, con lo que la codificación de los datos dado el modelo posiblemente será sencilla (en el ejemplo anterior, para codificar los datos, los puntos del plano, dado que el modelo es un polinomio de grado 100, sólo se necesitan las abscisas x_i de los puntos; en cambio, empleando un modelo más sencillo, por ejemplo un polinomio de grado 4, la longitud de codificación del modelo es menor, pero la longitud de codificación de los datos dado el modelo aumenta: ahora se necesitan las abscisas x_i de los puntos y además las diferencias (errores) entre las verdaderas ordenadas y_i, y las ordenadas predichas por el modelo $p(x_i)$). Así pues, el principio de descripción de longitud mínima pretende encontrar un modelo que represente un compromiso entre la exactitud del resultado obtenido y la simplicidad del mismo.

En nuestro caso, los modelos complejos son redes densamente conectadas, que son muy precisas (en el caso extremo, el grafo completo da lugar a una precisión absoluta) pero presentan dificultades computacionales y de comprensión. Por tanto se pretende encontrar redes que tal vez sean algo menos precisas pero con la ventaja de ser más simples. Existen diversos algoritmos de aprendizaje de redes que emplean el principio de descripción de longitud mínima como base para definir la métrica [10,30,49,76,80]. Para codificar el modelo, se necesita codificar la estructura gráfica (por ejemplo la lista de padres de cada nodo) y las distribuciones de probabilidad. Ambas codificaciones aumentan conforme el grafo es más denso. Para codificar los datos dado el modelo, se emplea una codificación cuya longitud disminuye conforme aumenta la precisión. Por ejemplo, en [49] se emplea una codificación basada en códigos de Huffman (a los datos más frecuentes se le asignan códigos más cortos).

Métodos Bayesianos En general, los algoritmos más habituales de este tipo buscan la estructura que maximiza la probabilidad de obtener una red condicio-

nada a la base de datos de que se dispone, empleando para ello la fórmula de Bayes. En otras palabras, el tipo de métrica que emplean está basado en la probabilidad de la red condicionada a la base de datos $P(\text{Red}|\text{Datos})$. Empleando la fórmula de Bayes, tenemos que

$$P(\text{Red}|\text{Datos}) = \frac{P(\text{Datos}|\text{Red})P(\text{Red})}{P(\text{Datos})}$$

y como los datos son siempre los mismos para cualquier red, el denominador en la expresión anterior es constante y puede obviarse. El término $P(\text{Red})$ representa la distribución a priori de cada red candidata (en muchos casos se supone uniforme y por tanto puede obviarse también), y el término $P(\text{Datos}|\text{Red})$, llamado *evidencia*, es la verosimilitud muestral promedio, que puede calcularse bajo ciertas suposiciones (diferentes suposiciones dan lugar a diferentes métricas). Existe una gran cantidad de trabajos sobre este tipo de técnicas [13,24,23,26,27,31,32,35,42,43,55,63], así como estudios comparativos [7,11,23,28,54].

Un Método de Aprendizaje de la Estructura usando una Métrica Bayesiana: El Algoritmo K2 El algoritmo que vamos a describir, llamado K2 [26], es tal vez el más conocido entre los algoritmos de aprendizaje de redes basados en ideas Bayesianas, y ha sido fuente de inspiración para muchos trabajos posteriores.

Supuestas ciertas una serie de condiciones (independencia de los casos de la base de datos, inexistencia de casos en la base de datos con datos perdidos, uniformidad de las distribuciones de probabilidad de los parámetros de una red, dada ésta) es posible derivar una fórmula que establece cuál es la distribucion de probabilidad conjunta de una estructura G y una base de datos BD. Esta fórmula se usa como métrica en un algoritmo de búsqueda local. Dicha métrica es la siguiente:

$$P(G, BD) = P(G) \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}!$$

donde

- r_i =número de casos de la variable X_i
- x_{ik} = k-ésimo valor de X_i
- q_i =número de casos de $\Pi_G(X_i)$
- w_{ij} = j-ésimo valor de $\Pi_G(X_i)$
- N_{ijk} =número de casos en la base de datos tales que $X_i = x_{ik}$ y $\Pi_G(X_i) = w_{ij}$
- $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$

Para hacer tratable el espacio de búsqueda se exige también una ordenación de las variables y la uniformidad de la distribución a priori sobre las distintas estructuras (por lo que el término $P(G)$ desaparece).

Puesto que, debido al orden introducido, se puede maximizar la métrica anterior trabajando separadamente con cada nodo X_i y su conjunto de padres $\Pi_G(X_i)$, el algoritmo va recorriendo las variables según el orden preestablecido, y para cada una de ellas, partiendo de un conjunto de padres inicialmente vacío, va paso a paso incluyendo aquellos padres que más incrementan la probabilidad de la estructura resultante, que se mide mediante la función:

$$g(X_i, \Pi_G(X_i)) = \prod_{j=1}^{q_i} \frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \prod_{k=1}^{r_i} N_{ijk}!$$

El algoritmo también emplea un valor u que representa el máximo número de padres que se le permite tener a cada nodo. V_i denota el conjunto de nodos anteriores a X_i en el orden fijado. El algoritmo K2 se detalla en la figura 9.

---

for $i = 1$ to n do

1. $\Pi_G(X_i) = \emptyset$
2. $P_{\text{old}} = g(X_i, \Pi_G(X_i))$
3. Ok = True
4. while Ok and $|\Pi_G(X_i)| < u$ do
   (a) Sea Z el nodo de $V_i \setminus \Pi_G(X_i)$ que maximiza $g(X_i, \Pi_G(X_i) \cup \{Z\})$
   (b) $P_{\text{new}} = g(X_i, \Pi_G(X_i) \cup \{Z\})$
   (c) if $P_{\text{new}} > P_{\text{old}}$
       then
         i. $P_{\text{old}} = P_{\text{new}}$
         ii. $\Pi_G(X_i) = \Pi_G(X_i) \cup \{Z\}$
       else Ok = False
5. return($\Pi_G(X_i)$)

Figura 9. El algoritmo K2.

Como se puede observar en la figura 9, la estrategia de búsqueda empleada es totalmente local: va eligiendo de entre los nodos anteriores a X_i en el orden, aquél que al ser incluido en el conjunto de padres más aumenta el valor de la métrica,

y este proceso se repite hasta que no se produce ninguna mejora, en cuyo caso se devuelve el conjunto de padres actual. Otros algoritmos de aprendizaje emplean la misma métrica de K2, pero cambian la técnica de búsqueda, utilizando por ejemplo algoritmos genéticos [50,51].

4.3 Aprendizaje de la Estructura usando Métodos Híbridos

También se han desarrollado algoritmos de aprendizaje de redes de creencia que utilizan una metodología híbrida, en el sentido que usan una técnica de búsqueda guiada por una métrica pero también hacen uso de tests de independencia condicional de una u otra forma [5,6,67,68].

Así, por ejemplo, algunos algoritmos [67,68] emplean como métrica la misma utilizada por el algoritmo K2, y para eliminar la restricción de disponer de una ordenación inicial de las variables que K2 requiere[2] utilizan la técnica del algoritmo PC: chequean independencias condicionales de un orden dado (empezando con orden cero), eliminan las aristas correspondientes y obtienen un grafo, que da lugar a un orden parcial entre las variables; se transforma el orden parcial en una ordenación completa, y se aplica el algoritmo K2. Este proceso se itera, pasando a chequear independencias condicionales de un orden mayor, hasta que no se producen mejoras.

Otros algoritmos [5,6] emplean un método de hibridación diferente, como es el caso del que vamos a describir en el siguiente apartado.

Un Método Híbrido de Aprendizaje de la Estructura: El Algoritmo BENEDICT BENEDICT [5,6], acrónimo compuesto por las palabras BElief NEtworks DIscovery using Cut-set Techniques, es una metodología híbrida para el aprendizaje de redes de creencia: utiliza una métrica específica y un método de búsqueda, pero también emplea explícitamente las relaciones de independencia condicional representadas en la red para definir la métrica, y utiliza tests de independencia para limitar el proceso de búsqueda.

La idea básica de los algoritmos de este tipo es cuantificar la discrepancia entre cualquier red candidata y la base de datos, midiendo para ello las discrepancias entre las independencias condicionales representadas en la red (a través del concepto de d-separación, separación direccional o independencia gráfica [61]) con las correspondientes independencias condicionales que puedan deducirse de la base de datos. La agregación de todas estas discrepancias será la métrica que utilicen

[2] Una técnica genérica para estimar una ordenación completa de las variables, que puede ser utilizada por cualquier algoritmo, basada en chequear independencias de orden cero y uno, y en el empleo de algoritmos genéticos, se describe en [19].

los algoritmos. En cuanto al proceso de búsqueda, BENEDICT emplea una técnica greedy: inicialmente se parte de un grafo completamente inconexo, y en cada iteración se prueba a insertar cada uno de los arcos posibles, eligiendo aquél que produce una mayor disminución de la discrepancia, e incluyéndolo en el grafo de forma permanente. Se continua con este proceso hasta que finalmente se satisface una condición de parada.

La versión de BENEDICT que comentaremos aquí determina la estructura de la red bajo la suposición de que se dispone de una ordenación total de las variables (como sucede con otros algoritmos de aprendizaje [26,44,78]).

Puesto que la idea básica del algoritmo es medir las discrepancias entre las independencias condicionales representadas en cualquier red candidata y aquéllas que reflejan los datos, lo primero que hay que plantear es qué independencias representa una red. Esta cuestión tiene, en principio, una respuesta muy clara: todas las relaciones de independencia que pueden deducirse del grafo mediante el criterio de d-separación. Sin embargo, el número de asertos de d-separación representados en un grafo puede ser muy alto (crece exponencialmente con el tamaño del mismo), y por razones de eficiencia y fiabilidad interesa excluir gran parte de ellos y utilizar sólo un subconjunto 'representativo' de todas las d-separaciones presentes. Una opción muy razonable se basa en utilizar el hecho de que en un grafo dirigido acíclico G, cualquier nodo X_j que no sea un descendiente de X_i está d-separado de X_i mediante el conjunto de padres de X_i en el grafo. Por tanto, se puede emplear como conjunto de independencias el formado por las sentencias de la forma $I(X_i, X_j|\Pi_G(X_i))$, para cada par de variables no adyacentes X_i y X_j; se supone que $X_j < X_i$ en el orden dado.

Sin embargo también importa el número de variables implicadas en esas independencias: cada uno de los asertos de independencia extraídos del grafo ha de ser contrastado con los datos mediante una medida de discrepancia, *Dep*. Así pues, interesa reducir lo más posible el tamaño de los conjuntos d-separadores: dados dos nodos X_i y X_j, tal que $X_j < X_i$, en lugar de utilizar el conjunto $\Pi_G(X_i)$, BENEDICT usa un conjunto de tamaño mínimo que consiga d-separar X_i de X_j. Encontrar este conjunto supondrá un esfuerzo adicional, pero se verá compensado con un decrecimiento en la computación de la medida de discrepancia; también se obtendrán unos resultados más fiables, ya que se necesitan menos datos para estimar fiablemente una medida de orden menor.

Por tanto, dada una red candidata G, las relaciones de independencia cuya discrepancia con los datos se va a calcular son: $I(X_i, X_j|S_G(X_i, X_j))$, para cualquier par de nodos no adyacentes X_i, X_j en G, tal que $X_j < X_i$, donde $S_G(X_i, X_j)$ es el mínimo conjunto d-separador de X_i y X_j. El método empleado para encontrar los conjuntos $S_G(X_i, X_j)$ está basado en una modificación del conocido algoritmo de Ford-Fulkerson para problemas de máximo flujo en redes [4]. En el algorit-

mo BENEDICT, el cálculo de los conjuntos d-separadores mínimos se lleva a cabo mediante la función *Mínimo-Corte*.

En cuanto a la forma de medir la discrepancia entre cualquier sentencia gráfica de independencia condicional representada en el grafo y los datos, se emplea la entropía cruzada de Kullback-Leibler [48], que mide el grado de dependencia entre X e Y dado que conocemos el valor de Z:

$$Dep(X, Y|Z) = \sum_{x,y,z} P(x, y, z) \log \frac{P(x, y|z)}{P(x|z)P(y|z)}$$

donde P representa la distribución de probabilidad estimada a partir de la base de datos. Esta medida toma el valor 0 cuando X e Y son realmente independientes dado Z, y es tanto mayor cuanto más dependientes entre sí son X e Y dado Z.

En lo que se refiere a la medida de discrepancia global entre el grafo G y la base de datos BD, $g(G, BD)$, que emplea el algoritmo para puntuar los méritos relativos de cada red candidata seleccionada por el proceso de búsqueda, se define de la siguiente manera:

$$g(G, BD) = \sum_{X_j < X_i,\, X_j \notin \Pi_G(X_i)} Dep(X_i, X_j | S_G(X_i, X_j))$$

El algoritmo BENEDICT se describe en la figura 10.

En la descripción del algoritmo anterior no se ha especificado la forma concreta en que se detiene el proceso de aprendizaje. Se utilizan tests de independencia para ir eliminando arcos del conjunto de arcos candidatos, y detener el proceso de forma natural cuando dicho conjunto llegue a ser vacío [6] (se eliminan arcos candidatos bien porque se insertan en la estructura o bien porque sus nodos extremos se hallan independientes). También se realiza un proceso final de poda de arcos (similar a los métodos de poda empleados habitualmente para árboles de clasificación): una vez terminado el proceso de inserción de arcos, se procede a una revisión de cada uno de ellos: se prueba a eliminarlos uno a uno, empleando también para ello un test de independencia.

Agradecimientos

Este trabajo ha sido financiado por la CICYT mediante el proyecto TIC96-0781.

1. Comenzar con un grafo G sin arcos ($G = \emptyset$)
2. Se fija $L = \{X_j \rightarrow X_i | X_j < X_i\}$; $g := 0$
3. Para cada par de nodos $X_s < X_t$ hacer $g := g + Dep(X_t, X_s|\emptyset)$
4. $min := g$
5. Hasta *detenerse* hacer
   (a) Para cada enlace $X_j \rightarrow X_k \in L$ hacer
      i. $G' := G \cup \{X_j \rightarrow X_k\}$; $g := 0$
      ii. Para cada nodo X_t hacer
         Para cada nodo $X_s < X_t$ tal que $X_s \notin \Pi_{G'}(X_t)$ hacer
            $S_{G'}(X_t, X_s) := Minimo\text{-}Corte(X_t, X_s)$
            $g := g + Dep(X_t, X_s|S_{G'}(X_t, X_s))$
      iii. Si $g < min$ entonces
         $min := g$
         $X := X_k$; $Y := X_j$
   (b) $G := G \cup \{Y \rightarrow X\}$
   (c) $L := L \backslash \{Y \rightarrow X\}$

Figura 10. El algoritmo BENEDICT.

Referencias

1. S. Acid, L.M. de Campos, A. González, R. Molina, N. Pérez de la Blanca, CASTLE: A tool for bayesian learning, en: Proceeding of the ESPRIT'91 Conference, Commission of the European Communities (1991) 363-377.
2. S. Acid, L.M. de Campos, A. González, R. Molina, N. Pérez de la Blanca, Learning with CASTLE, en: R. Kruse, P. Siegel, eds., Symbolic and Quantitative Approaches to Uncertainty, Lecture Notes in Computer Science 548, (Springer Verlag, Berlin, 1991) 99-106.
3. S. Acid, L.M. de Campos, Approximations of causal networks by polytrees: an empirical study, en: B. Bouchon-Meunier, R.R. Yager, L.A. Zadeh, eds., Advances in Intelligent Computing, Lecture Notes in Computer Science 945, (Springer Verlag, Berlin, 1995) 149-158.
4. S. Acid, L.M. de Campos, An algorithm for finding minimum d-separating sets in belief networks, en: E. Horvitz, F. Jensen, eds., Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence (Morgan Kaufmann, San Mateo, 1996) 3-10.
5. S. Acid, L.M. de Campos, Benedict: An algorithm for learning probabilistic belief networks, en: Proceedings of the Sixth IPMU Conference (1996) 979-984.

6. S. Acid, L.M. de Campos, Algoritmos híbridos para el aprendizaje de redes de creencia, en: Actas de la VII Conferencia de la Asociación Española para la Inteligencia Artificial (1997) 499-508.
7. C.F. Aliferis, G.F. Cooper, An evaluation of an algorithm for inductive learning of Bayesian belief networks using simulated data sets, en: R. López de Mántaras, D. Poole, eds., Uncertainty in Artificial Intelligence: Proceedings of the Tenth Conference (Morgan Kaufmann, San Francisco, 1994) 8-14.
8. L.R. Andersen, J.H. Krebs, J. Damgaard, STENO: an expert system for medical diagnosis based on graphical models and model search, Journal of Applied Statistics 18 (1991) 139-153.
9. I.A. Beinlich, H.J. Suermondt, R.M. Chavez, G.F. Cooper, The Alarm monitoring system: A case study with two probabilistic inference techniques for belief networks, en: Proceedings of the Second European Conference on Artificial Intelligence in Medicine (1989) 247-256.
10. R.R. Bouckaert, Belief network construction using the minimum description length principle, en: M. Clarke, R. Kruse, S. Moral, eds., Symbolic and Quantitative Approaches to Reasoning and Uncertainty, Lecture Notes in Computer Science 747, (Springer Verlag, Berlin, 1993) 41-48.
11. R.R. Bouckaert, Properties of Bayesian belief network learning algorithms, en: R. López de Mántaras, D. Poole, eds., Uncertainty in Artificial Intelligence: Proceedings of the Tenth Conference (Morgan Kaufmann, San Francisco, 1994) 102-109.
12. R.R. Bouckaert, Optimizing causal orderings for generating DAGs from data, en: D. Dubois, M.P. Wellman, B.D. D'Ambrosio, P. Smets, eds., Uncertainty in Artificial Intelligence: Proceedings of the Eighth Conference (Morgan and Kaufmann, San Mateo, 1992) 9-16.
13. W. Buntine, Operations for learning with graphical models, Journal of Artificial Intelligence Research 2 (1994) 159-225.
14. W. Buntine, A guide to the literature on learning probabilistic networks from data, IEEE Transactions on Knowledge and Data Engineering 8 (1996) 195-210.
15. L.M. de Campos, Characterizations of decomposable dependency models, Journal of Artificial Intelligence Research 5 (1996) 289-300.
16. L.M. de Campos, Independency relationships and learning algorithms for singly connected networks, por aparecer en Journal of Experimental and Theoretical Artificial Intelligence (1998). Disponible también como DECSAI Technical Report 96-02-04.
17. L.M. de Campos, J.F. Huete, Algorithms for learning decomposable models and chordal graphs, en: D. Geiger, P.P. Shenoy, eds., Uncertainty in Artificial Intelligence: Proceedings of the Thirteenth Conference (Morgan and Kaufmann, San Mateo, 1997) 46-53.
18. L.M. de Campos, J.F. Huete, On the use of independence relationships for learning simplified belief networks, International Journal of Intelligent Systems 12 (1997) 495-522.
19. L.M. de Campos, J.F. Huete, Aproximación de una ordenación de variables en redes causales mediante algoritmos genéticos, en: Actas de la VII Conferencia de la Asociación Española para la Inteligencia Artificial (1997) 155-164.

20. B. Cestnik, Estimating probabilities: A crucial task in Machine Learning, Proceedings of the European Conference on Artificial Intelligence (ECAI'90) (1990) 147-149.
21. B. Cestnik, I. Bratko, On estimating probabilities in tree pruning, en: Y. Kodratoff, ed., Lecture Notes in Artificial Intelligence (1991) 138-150.
22. J. Cheng, D.A. Bell, W. Liu, An algorithm for Bayesian belief network construction from data, en: Proceedings of the Seventh International Workshop on Artificial Intelligence and Statistics (1997) 83-90.
23. D.M. Chickering, Learning equivalence classes of Bayesian network structures, en: E. Horvitz, F. Jensen, eds., Uncertainty in Artificial Intelligence: Proceedings of the Twelfth Conference (Morgan Kaufmann, San Mateo, 1996) 150-157.
24. D.M. Chickering, D. Geiger, D. Heckerman, Learning bayesian networks is NP-Hard, Technical Report MSR-TR-94-17, Microsoft Research (1994).
25. C.K. Chow, C.N. Liu, Approximating discrete probability distribution with dependence trees, IEEE Transactions on Information Theory 14 (1968) 462-467.
26. G.F. Cooper, E. Herskovits: A bayesian method for the induction of probabilistic networks from data, Machine Learning 9 (1992) 309-347.
27. G.F. Cooper, A method for learning belief networks that contain hidden variables, Journal of Intelligent Information Systems 4 (1995) 71-88.
28. R.G. Cowell, A.P. Dawid, P. Sebastiani, A comparison of sequential learning methods for incomplete data, Research Report 135, Department of Statistical Science, University College (1994).
29. A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, Journal of the Royal Statistical Society B 39 (1977) 1-38.
30. N. Friedman, M. Goldszmidt, Learning Bayesian networks with local structure, en: E. Horvitz, F. Jensen, eds., Uncertainty in Artificial Intelligence: Proceedings of the Twelfth Conference (Morgan Kaufmann, San Mateo, 1996) 252-262.
31. N. Friedman, M. Goldszmidt, Building classifiers using bayesian networks, en: Proceedings of the National Conference on Artificial Intelligence (AAAI Press, Menlo Park, 1996) 1277-1284.
32. N. Friedman, D. Geiger, M. Goldszmidt, Bayesian network classifiers, Machine Learning 29 (1997) 131-163.
33. R.M. Fung, S.L. Crawford, Constructor: A system for the induction of probabilistic models, en: Proceedings of the Eighth National Conference on Artificial Intelligence (1990) 762-769.
34. D. Geiger, An entropy-based learning algorithm of bayesian conditional trees, en: D. Dubois, M.P. Wellman, B.D. D'Ambrosio, P. Smets, eds., Uncertainty in Artificial Intelligence: Proceedings of the Eighth Conference (Morgan and Kaufmann, San Mateo, 1992) 92-97.
35. D. Geiger, D. Heckerman, A characterisation of the Dirichlet distribution with application to learning Bayesian networks, en: P. Besnard, S. Hanks, eds., Uncertainty in Artificial Intelligence: Proceedings of the Eleventh Conference (Morgan and Kaufmann, San Francisco, 1995) 196-207.

36. D. Geiger, A. Paz, J. Pearl, Learning causal trees fron dependence information, Proceedings of the Eighth National Conference on Artificial Intelligence (1990) 770-776.
37. D. Geiger, A. Paz, J. Pearl, Learning simple causal structures, International Journal of Intelligent Systtems 8 (1993) 231-247.
38. I.J. Good, The Estimation of Probabilities (MIT Press, Cambridge, 1965).
39. W. Hastings, Monte Carlo sampling methods using Markov chains and their applications, Biometrika 57 (1970) 97-109.
40. D. Heckerman, A tractable inference algorithm for diagnosing multiple diseases, en: R.D. Shachter, T.S. Levitt, L.N. Kanal, J.F. Lemmer, eds., Uncertainty in Artificial Intelligence 5 (North-Holland, Amsterdam, 1990) 163-171.
41. D. Heckerman, A tutorial on learning bayesian networks, Technical Report MSR-TR-95-06, Microsoft Research, Advanced Technology Division (1995).
42. D. Heckerman, Bayesian networks for knowledge discovery, en: U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy, eds., Advances in Knowledge Discovery and Data Mining (MIT Press, Cambridge, 1996) 273-305.
43. D. Heckerman, D. Geiger, D.M. Chickering, Learning bayesian networks: The combination of knowledge and statistical data, Machine Learning 20 (1995) 197-243. También en: R. López de Mántaras, D. Poole, eds., Uncertainty in Artificial Intelligence: Proceedings of the Tenth Conference (Morgan Kaufmann, San Francisco, 1994) 293-301.
44. E.H. Herskovits and G.F. Cooper, Kutató: An entropy-driven system for the construction of probabilistic expert systems from Databases, en: P. Bonissone, ed., Uncertainty in Artificial Intelligence: Proceedings of the Sixth Conference (Cambridge, 1990) 54-62. También en Report KSL-90-22 Stanford University (1990).
45. J.F. Huete, L.M. de Campos, Learning causal polytrees, en: M. Clarke, R. Kruse, S. Moral, eds., Symbolic and Quantitative Approaches to Reasoning and Uncertainty, Lecture Notes in Computer Science 747 (Springer Verlag, Berlin, 1993) 180-185.
46. R. Jirousek, S. Preucil, On the effective implementation of the iterative proportional fitting procedure, Computational Statistics and Data analysis 19 (1995) 177-189.
47. P.J. Krause, Learning probabilistic networks, Technical Report, Philips Research Laboratories (1998).
48. S. Kullback, Information Theory and Statistics (Dover Publications, 1968).
49. W. Lam, F. Bacchus, Learning belief networks: an approach based on the MDL principle, Computational Intelligence 10 (1994) 269-293.
50. P. Larrañaga, M. Poza, Y. Yurramendi, R.H. Murga, C.M.H. Kuijpers, Structure learning of Bayesian networks by genetic algorithms: A performance analysis and control parameters, IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (1996) 912-926.
51. P. Larrañaga, R.H. Murga, M. Poza, C.M.H. Kuijpers, Structure learning of Bayesian networks by hybrid genetic algorithms, en: D. Fisher, H.J. Lenz, eds., Learning from Data: AI and Statistics V (springer Verlag, 1996) 165-174.
52. S.L. Lauritzen, The EM algorithm for graphical association models with missing data, Computational Statistics and Data Analysis 19 (1995) 157-224.

53. S.L. Lauritzen, D.J. Spiegelhalter, Local computations with probabilities on graphical structures and their application to expert systems, Journal of the Royal Statistical Society B 50 (1988) 157-224.
54. S.L Lauritzen, B. Thiesson, D. Spiegelhalter, Diagnostic systems created by model selection methods: A case study, en: P. Cheeseman, R. Oldford, eds., AI and Statistics IV, Lecture Notes in Statistics 89 (Springer Verlag, New York, 1994) 143-152.
55. D. Madigan, A. Raftery, Model selection and accounting for model uncertainty in graphical models using Occam's window, Journal of the American Statistics Association 89 (1994) 1535-1546.
56. D. Madigan, J. York, Bayesian graphical models for discrete data, International Statistical Review 63 (1995) 215-232.
57. C. Meek, Causal inference and causal explanation with background knowledge, en: P. Besnard, S. Hanks, eds., Uncertainty in Artificial Intelligence: Proceedings of the Eleventh Conference (Morgan and Kaufmann, San Francisco, 1995) 403-410.
58. D. Michie, D.J. Spiegelhalter, C.C. Taylor, eds., Machine Learning, Neural and Statistical Classification (Ellis Horwood, London, 1994).
59. R. Molina, L.M. de Campos, J. Mateos, Using Bayesian Algorithms for Learning Causal Networks in Classification Problems, en: B. Bouchon-Meunier, L. Valverde, R.R. Yager, eds., Uncertainty in Intelligent Systems (North-Holland, 1993) 49-59.
60. K.G. Olesen, S.L. Lauritzen, F.V. Jensen, aHugin: A system creating adaptive causal probabilistic networks, en: D. Dubois, M.P. Wellman, B.D. D'Ambrosio, P. Smets, eds., Uncertainty in Artificial Intelligence, Proceedings of the Eighth Conference (Morgan and Kaufmann, San Mateo, 1992) 223-229.
61. J. Pearl, Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference (Morgan and Kaufmann, San Mateo, 1988).
62. J. Pearl, T.S. Verma, A theory of inferred causation, en: J.A. Allen, R. Fikes, E. Sandewall, eds., Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference (Morgan and Kaufmann, San Mateo, 1991) 441-452.
63. M. Ramoni, P. Sebastiani, Discovering Bayesian networks in incomplete databases, KMI-TR-46 Technical Report, The Open University (1997).
64. G. Rebane, J. Pearl, The recovery of causal polytrees from statistical data, en: L.N. Kanal, T.S. Levitt, J.F. Lemmer, eds., Uncertainty in Artificial Intelligence 3 (North-Holland, Amsterdam, 1989) 175-182.
65. J. Rissanen, Stochastic complexity, Journal of the Royal Statistical Society B 49 (1987) 223-239.
66. S. Sarkar, Using tree-decomposable structures to approximate belief networks, en: D. Heckerman, A. Mamdani, eds., Uncertainty in Artificial Intelligence: Proceedings of the Ninth Conference (Morgan and Kaufmann, San Mateo, 1993), 376-382.
67. M. Singh, M. Valtorta, An algorithm for the construction of Bayesian network structures from data, en: D. Heckerman, A. Mamdani, eds., Uncertainty in Artificial Intelligence: Proceedings of the Ninth Conference (Morgan Kaufmann, San Francisco, 1993) 259-265.

68. M. Singh, M. Valtorta, Construction of Bayesian network structures from data: A brief survey and an efficient algorithm, International Journal of Approximate Reasoning 12 (1995) 111-131.
69. D.J. Spiegelhalter, R. Cowell, Learning in probabilistic expert systems, en: J.M. Bernardo, J.O. Berger, A.P. Dawid, A.F. Smith, eds., Bayesian Statistics 4 (Oxford University Press, 1992) 447-465.
70. D.J. Spiegelhalter, A.P. Dawid, S.L. Lauritzen, R.G. Cowell, Bayesian analysis in expert systems (with discussion), Statistical Science 8 (1993) 219-283.
71. D.J. Spiegelhalter, S.L. Lauritzen, Sequential updating of conditional probabilities on directed graphical structures, Networks 20 (1990) 579-605.
72. P. Spirtes, Detecting causal relations in the presence of unmeasured variables, en: B. D'Ambrosio, P. Smets, P.P. Bonissone, eds., Uncertainty in Artificial Intelligence: Proceedings of the Seventh Conference (Morgan and Kaufmann, 1991) 392-397.
73. P. Spirtes, C. Glymour, R. Scheines, An algorithm for fast recovery of sparse causal graphs, Social Science Computing Reviews 9 (1991) 62-72.
74. P. Spirtes, C. Glymour, R. Scheines, Causation, Prediction and Search, Lecture Notes in Statistics 81 (Springer Verlag, New York, 1993).
75. S. Srinivas, S. Russell, A. Agogino, Automated construction of sparse bayesian networks from unstructured probabilistic models and domain information, en: M. Henrion, R.D. Shachter, L.N. Kanal, J.F. Lemmer, eds., Uncertainty in Artificial Intelligence 5 (North-Holland, Amsterdam, 1990) 295-308.
76. J. Suzuki, A construction of bayesian networks from databases based on the MDL principle, en: D. Heckerman, A. Mamdani, eds., Uncertainty in Artificial Intelligence: Proceedings of the Ninth Conference (Morgan Kaufmann, San Francisco, 1993) 266-273.
77. T. Verma, J. Pearl, Equivalence and synthesis of causal models, en: Uncertainty in Artificial Intelligence: Proceedings of the Sixth Conference (Mass, 1990) 220-227.
78. T. Verma, J. Pearl, Causal networks: Semantics and expressiveness, en: R.D. Shachter, T.S. Levitt, L.N. Kanal, J.F. Lemmer, eds., Uncertainty in Artificial Intelligence 4 (North-Holland, Amsterdam, 1990), 69-76.
79. T. Verma, J. Pearl, An algorithm for deciding if a set of observed independencies has a causal explanation, en: D. Dubois, M.P. Wellman, B. D'Ambrosio, P. Smets, eds., Uncertainty in Artificial Intelligence: Proceedings of the Eighth Conference (Morgan and Kaufmann, San Mateo, 1992) 323-330.
80. D. Wedelin, Efficient algorithms for probabilistic inference, combinatorial optimization and the discovery of causal structure from data, Ph.D. Thesis, Chalmers University of Thechnology, Goteborg (1993).
81. N. Wermuth, S.L. Lauritzen, Graphical and recursive models for contingency tables, Biometrika 70 (1983) 537-552.
82. J. Whittaker, Graphical Models in Applied Multivariate Statistics (Wiley, Chichester, 1991).
83. S.K.M Wong, Y. Xiang, Construction of a Markov network from data for probabilistic inference, en: Proceedings of the Third International Workshop on Rough Sets and Soft Computing (1994) 562-569.

84. Y. Xiang, S.K.M. Wong, N. Cercone, A 'microscopic' study of minimum entropy search in learning decomposable Markov models, Machine Learning 26 (1997) 65-92.

Aprendizaje Automático de Modelos Gráficos II. Aplicaciones a la Clasificación Supervisada

Pedro Larrañaga

Dpto. de Ciencias de la Computación e Inteligencia Artificial
Universidad del País Vasco - Euskal Herriko Unibertsitatea
Paseo Manuel de Lardizabal 1
20080 San Sebastián
correo-e: ccplamup@si.ehu.es

Resumen

En este artículo se plantea - con un enfoque divulgativo - el abordar el problema del reconocimiento de patrones - también denominado clasificación supervisada - por medio de Redes Bayesianas. En primer lugar se introducen muy brevemente algunos paradigmas provenientes de la Estadística - Ánalisis Discriminante, Regresión Logística, Árboles de Clasificación, K-NN - como de la Inteligencia Artificial - Inducción de Reglas, Redes Neuronales - los cuales vienen siendo utilizados en dicho problema. Se exponen distintos criterios de evaluación - tasa de error, rapidez de la clasificación, interpretatibidad y simplicidad - de los modelos construidos, asi como diferentes métodos de estimación de la tasa de error - método H, remuestreo, booststrapping -. A continuación se presentan distintas aproximaciones al mismo - General, Naive - Bayes, Pazzani, Estructura de Árbol Aumentada, Markov Blanket, Markov Blanket Reducido - basadas en el paradigma de Redes Bayesianas, asi como distintos criterios de ajuste - porcentaje de bien clasificados, sensibilidad, especifidad -. Finalmente se presenta una comparación empírica de algunos de los métodos anteriores en un ejemplo real de predicción de la supervivencia en pacientes aquejados de Melanoma, llevándose a cabo la estimación de la bondad de los distintos modelos por medio de validaciones cruzadas.

1 Introducción

La palabra *clasificación* se usa con distintos significados, de ahí que me parece conveniente aclarar desde un principio la terminología a utilizar en el contenido de este artículo. Desde un punto de vista general, podemos distinguir entre la denominada *clasificación no supervisada* (*cluster* análisis, o reconocimiento de patrones no supervisado) y la *clasificación supervisada* (reconocimiento de patrones supervisado).

La *clasificación no supervisada* (Figura 1) - véase por ejemplo Kaufman y Rousseeuw (1990) [17] - se refiere al proceso de definir clases de objetos. Es decir, partiendo de una colección de N objetos, $O_1, O_2, ..., O_i, ..., O_N$, caracterizados por p variables $X_1, X_2, ..., X_j, ..., X_p$ - discretas y/o continuas - se trata de encontrar una estructura de clases en los mismos, de tal manera que los objetos que pertenecen a una misma clase presenten una gran homogeneidad, mientras que, por otra parte, las distintas clases construidas sean muy heterogéneas entre sí. Si bien estas técnicas surgieron en el dominio de la Biología - tratando de agrupar plantas - hoy en dia se vienen aplicando en muy diversos dominios, entre los cuales cabe citar el tratamiento digital de imágenes y el marketing. Con objeto de clarificar la terminología diremos que mientras que en textos estadísticos se habla de *taxonomía numérica* para referirse a este tipo de métodos, en áreas cercanas a la Inteligencia Artificial se utiliza la denominación de *formación de conceptos.*

Por lo que se refiere a la *clasificación supervisada* (Hand (1997) [13]) - tambien denominada *reconocimiento de patrones* - cada objeto se encuentra descrito por un vector de p características - variables predictoras - $X_1, X_2, ..., X_j, ..., X_p$ - discretas y/o continuas - así como por la clase a la que pertenece, la cual viene indicada en la variable C. Se conoce por tanto la clase verdadera para una muestra de objetos, y el ejercicio consiste en construir un modelo - formular una "regla"en sentido amplio - para asignar a nuevos objetos, de los que se conocen las p características anteriores - o algunas de ellas - , un valor de la variable C (Tabla 1). La muestra de objetos que sirve para construir el clasificador - es decir el modelo - se denomina *conjunto de entrenamiento*, ya que a partir de ella se determina la estructura y los parámetros del modelo clasificador. Otras denominaciones habituales para dicho conjunto de entrenamiento son las de *conjunto de aprendizaje* o *conjunto de diseño.* Teniendo en cuenta que el objetivo de la clasificación supervisada es el clasificar adecuadamente nuevos casos, suele ser habitual plantearse la validación de los modelos construidos. Dicha validación se puede efectuar de muy diversas maneras - veáse Sección 2.5 -, ilustrándose con el ejemplo que se describe a continuación el procedimiento más simple de validación.

Imaginemos que una determinada entidad financiera se plantee el construir un sistema clasificador que les ayude a decidir acerca de la concesión o denegación de un crédito solicitado por sus clientes. Para ello decide utilizar información de los 6000 últimos clientes a los que se les concedió un crédito. Dicha información - veáse Tabla 2 - incluye como variables predictoras las siguientes: X_1: Edad; X_2: Estado Civil; X_3: Nivel de Estudios; X_4: Propietario de Casa; X_5: Nivel de Ingresos; X_6: Crédito Solicitado. En la información a utilizar para construir el clasificador se incluye también la clase a la que pertenece - valor de la variable C - cada uno de los 6000 casos; es decir para cada individuo si fué capaz de hacer frente al crédito $C = 0$, o no $C = 1$. Supongamos que los 5000 primeros casos

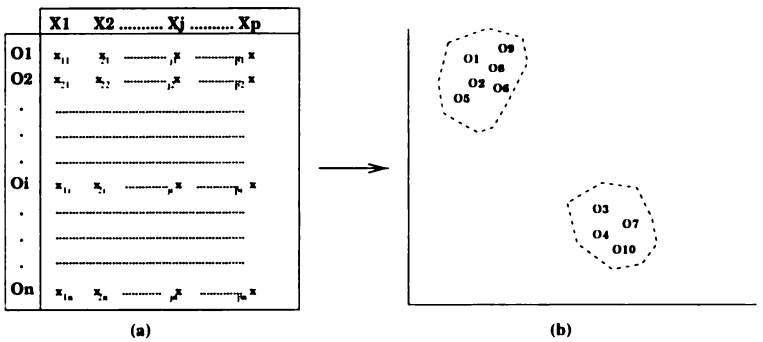


Figura 1. (a) Matriz de datos inicial; (b) Resultado de una clasificación no supervisada

van a servir para construir el modelo de clasificación. Diremos que dichos casos serán el conjunto de entrenamiento. Una vez construido dicho modelo, y tratando de estudiar su bondad para una posterior utilización del mismo, nos podemos plantear el medir de una manera sencilla dicha bondad a partir del porcentaje de casos bien clasificados por el modelo construido en los siguientes 1000 casos del fichero, los cuales en este ejemplo jugarán el papel de conjunto de testeo. Obviamente podemos tambien tener en cuenta las diferencias existentes entre los dos tipos de error que se pueden llegar a cometer. Es decir, clasificar como "capaz de hacer frente al crédito" un individuo que en la realidad pertenece a la clase $C = 1$, o clasificar como "no capaz de hacer frente al crédito" un individuo que en realidad pertenece a la clase $C = 0$.

Este planteamiento de clasificación supervisada es común a gran cantidad de problemas que surgen en diversos dominios. Así por ejemplo, en medicina se puede estar interesado en un sistema de ayuda al diagnóstico o al pronóstico de un enfermo que presenta una serie de síntomas, en finanzas podemos tratar de construir modelos que sean capaces de determinar a partir de los datos financieros de una empresa si ésta es candidata a sufrir una quiebra o no. Otros problemas en los que estos modelos han demostrado su validez son: el reconocimiento de voz, la verificación de firmas, la clasificación de cromosomas humanos con objeto de detectar anormalidades, detección del fraude en compañías de seguros, ...

En este trabajo se expondrán de manera breve algunas características básicas de métodos que para tratar el problema de la clasificación supervisada se han venido desarrollando tanto en la Estadística como en el denominado Aprendizaje Automático, este último dentro de la Inteligencia Artificial. Si bien en sus inicios existía una diferencia clara entre las aproximaciones provenientes de ambas

	X_1	X_2	...	X_j	...	X_p	C
O_1	x_1^1	x_2^1	...	x_j^1	...	x_p^1	c^1
O_2	x_1^2	x_2^2	...	x_j^2	...	x_p^2	c^2
...	...	...	...	...	...	...	....
O_i	x_1^i	x_2^i	...	x_j^i	...	x_p^i	c^i
...	...	...	...	...	...	...	....
O_N	x_1^N	x_2^N	...	x_j^N	...	x_p^N	c^N
O_{N+1}	x_1^{N+1}	x_2^{N+1}	...	x_j^{N+1}	...	x_p^{N+1}	?
...	...	...	...	...	...	...	....
O_{N+M}	x_1^{N+1}	x_2^{N+1}	...	x_j^{N+1}	...	x_p^{N+1}	?

Tabla 1. Matriz de datos inicial previa a la clasificación supervisada

	X_1	X_2	X_3	X_4	X_5	X_6	C	C_M
O_1	34	soltero	bachiller	si	200.000	1.000.000	0	
O_2	40	casado	licenciado	si	250.000	1.500.000	0	
...	...	...	...	...	...	...	...	...
O_{5000}	46	casado	sin estudios	no	150.000	2.500.000	1	
O_{5001}	36	soltero	sin estudios	si	250.000	3.500.000	1	1
...	...	...	...	...	...	...	...	...
O_{6000}	46	casado	sin estudios	no	150.000	2.500.000	1	0

Tabla 2. Conjunto de entrenamiento y conjunto de testeo para el problema d e la inducción de un clasificador de concesión de créditos

disciplinas, ya que mientras que en los métodos desarrollados alrededor del Aprendizaje Automático se asumía que se trabajaba en dominios sin ruido - individuos con las mismas características pertenecen a la misma clase -, hoy en dia la frontera entre ambas disciplinas se viene haciendo cada vez más difusa. De hecho en el paradigma que se desarrollará con más detalle en este artículo - Redes Bayesianas - las aproximaciones provenientes de ambas disciplinas han servido para un mejor desarrollo del mismo.

La estructura del trabajo es como sigue: en la sección 2 se introducen brevemente algunos paradigmas de clasificación supervisada - Análisis Discriminate, Regresión Logística, *K-NN*, Árboles de Clasificación, Inducción de Reglas, Redes Neuronales -, así como distintos criterios de evaluación de los mismos, y métodos de validación de los modelos clasificatorios creados. La sección 3 presenta distintas aproximaciones al problema de la clasificación basadas en el paradigma de las Redes Bayesianas - General, *Naive - Bayes*, Estructura de Árbol Aumentada, Pazzani, *Markov Blanket*, *Markov Blanket* Reducido - . La sección 4 presenta un ejemplo real de pronóstico de la supervivencia en pacientes aquejados de melanoma, en que se han aplicado algunos de los modelos expuestos en relación con el paradigma de Redes Bayesianas. Se finaliza con las conclusiones y posibles líneas de trabajo en este campo.

2 Paradigmas de Clasificación Supervisada

2.1 Introducción

Los paradigmas que se presentan de manera breve se han agrupado en paradigmas clasificatorios provenientes de la Estadística y en provenientes de la Inteligencia Artificial. Tal y como se ha comentado con anterioridad esta clasificación puede llegar a resultar difusa. En concreto los Árboles de Clasificación y el *K-NN* que aquí se presentan como provenientes de la Estadística, han sido motivo de estudio por parte de gran número de investigadores provenientes de la Inteligencia Artificial.

2.2 Paradigmas provenientes de la Estadística

Análisis Discriminante El Análisis Discriminante - introducido por Fisher (1936) [11] - crea factores - variables sintéticas - que son funciones discriminantes - lineales o cuadráticas - de las p variables explicativas. El peso asignado a cada variable indica la importancia de la misma en la discriminación y se calcula a partir de operaciones algebráicas realizadas sobre la matriz de varianzas-covarianzas de los datos. Una vez obtenido el modelo, la asignación a las clases de los nuevos

individuos se lleva a cabo calculando la puntuación obtenida por dicho individuo en la función discriminante obtenida, y comparando dicha puntuación con las que se obtienen por los baricentros de los distintos grupos.

El método funciona adecuadamente en el caso en que las clases sean linealmente separables, o separables por una función cuadrática. Una ventaja adicional del mismo es que la mayoría de los paquetes estadísticos más utilizados - SPSS, SAS, BMDP, SYSTAT, .. - incorporan procedimientos para construir este tipo de modelos.

Regresión Logística Sean $X_1, X_2, ..., X_p$ variables explicativas, C variable a explicar (dicotómica), la Regresión Logística - Kleinbaum (1994) [18] - crea un modelo del tipo siguiente:

$$\pi(\boldsymbol{X}) = P(C = 1 | X_1 = x_1, ..., X_p = x_p) = \frac{1}{1 + e^{-\beta_0 + \sum_{i=1}^{p} \beta_i x_i}}$$

donde $\beta_0, \beta_1, ..., \beta_p$ son parámetros que se estiman a partir del método de estimación máximo verosímil.

El modelo resulta de gran atractivo en el mundo médico, debido a la fácil interpretación, en términos de riesgo, que tienen los parámeros $\beta_i; i = 0, 1, .., p$.

Árboles de Clasificación Los Árboles de Clasificación constituyen un método de particionamiento recursivo o de segmentación del conjunto de objetos, cuyo objetivo es ir particionando el conjunto de casos en base a un criterio - la mayoría de las veces relacionado con la entropía - , habitualmente basado en una única variable, hasta que al final del proceso - en una situación ideal - en los distintos grupos creados no haya más que individuos que pertenezcan a una de las clases de la variable C.

Este paradigma se ha ido desarrollando tanto en la Estadística - Breiman y col. (1984) [3] - como en el Aprendizaje Automático - Quinlan (1986) [27] - . Mientras que en la Estadística los modelos construidos tenian en cuenta la existencia de variables tanto discretas como continuas, así como el hecho de que los dominios tenian ruido, en las aproximaciones desarrolladas en el Aprendizaje Automático se presuponia que en el dominio no existia ruido y todas las variables predictoras eran discretas.

K-NN Esta aproximación es quizás la más intuitiva de las que se van a comentar. Se basa en la idea de que objetos que tienen vectores de características cercanos

van a tener el mismo valor para la variable a clasificar. *K-NN* - Cover y Hart (1967) [8] - asigna a un nuevo individuo O_q a clasificar, la clase más frecuente de los K ejemplos más cercanos a O_q en el fichero de casos de entrenamiento.

Existen varios refinamientos del algoritmo básico. Uno utilizado habitualmente, consiste en pesar la contribución de cada uno de los K vecinos, en función de la distancia al individuo a clasificar, O_q, dando más peso a los vecinos más cercanos.

Si bien estos métodos de *K-NN* surgieron dentro de la Estadística, hoy en dia existen métodos muy similares dentro del Aprendizaje Automático, bajo la denominacion de *Instance Based Learning* (*IBL*) - Aha y col. (1991) [1] - .

2.3 Paradigmas provenientes de la Inteligencia Artificial

Inducción de Reglas Las reglas, dada su transparencia, son un tipo de paradigma que goza de gran popularidad dentro de la Inteligencia Artificial. Si bien en sus comienzos dichas reglas se obtenian a partir de la infomación proporcionada por el experto en el dominio a tratar, desde hace varios años se vienen desarrollando, con relativo éxito, sistemas capaces de inducir reglas. Ejemplos de los mismos pueden ser: GABIL [9] - De Jong y col. (1993) - y SIA [29] - Venturini (1993) - . Ambos pertenecen respectivamente a las denominadas aproximaciones de Pittsburgh y Michigan a los sistemas clasificadores. Dichos sistemas clasificadores utilizan a los algoritmos genéticos - Holland (1975) [15] - como heurísticos de búsqueda dentro del espacio de todas las reglas posibles.

Redes Neuronales Las Redes Neuronales (Bishop (1996) [2]) - inspiradas en sus comienzos en los sistemas neurológicos biológicos - modelan el conocimiento, en problemas clasificatorios, por medio de una estructura que presenta como nodos de entrada a las variables predictoras, como nodos de salida a las distintas categorías de la variable a clasificar, y presentan varias capas intermedias de nodos - denominadas capas ocultas - con las que pueden atacar problemas no lineales. Los nodos de un determinado nivel se conectan con los del nivel siguiente, cuantificándose dicha conexión por medio de unos pesos, los cuales en el proceso de aprendizaje - habitualmente llevado a cabo por un algoritmo de retropropagación del error - se van ajustando.

Si bien en algunos problemas reales han demostrado su eficacia, su poca transparencia hace que sean pobres candidatas para problemas en los que se trata de extraer conocimiento y que el algoritmo de inducción ayude a entender mejor el problema en cuestión.

2.4 Criterios de evaluación

Varios son los criterios en los que nos podemos basar para medir la bondad del modelo creado. Entre los mismos podemos destacar:

(i) *Tasa de error* obtenida por el clasificador. En la siguiente sección de validación veremos con detalle diferentes aproximaciones al problema de tratar de estimar de manera "honesta"dicha tasa de error.

(ii) *Rapidez de la clasificación.* Para algunos problemas en los que el sistema debe de responder en tiempo real, ésta será una característica a tener en cuenta.

(iii) *Interpretatibidad* del modelo obtenido por el clasificador. En algunos dominios interesa que el modelo ayude a entender mejor el problema que se está tratando.

(iv) *Simplicidad del modelo.* Guiándose por criterios de parsimonia interesa construir modelos lo más simples posibles, para por una parte ganar en interpretatibidad y por otra parte en rapidez de razonamiento. Es por ello por lo que una estrategia habitual suele consistir en utilizar una función de evaluación - de los distintos modelos que se van obteniendo durante la búsqueda - que pondere negativamente la complejidad del modelo en base a distintos criterios - Akaike, MDL, BIC, ... -.

Entre las diferentes estrategias para guiar la búsqueda de un modelo, podemos hablar de manera general de 4 grandes aproximaciones:

(i) *Inclusión iterativa.* Se comienza con un modelo vacío, para en cada paso seleccionar para su inclusión, aquella variable - de entre aquellas que no están incluidas en el modelo - que más poder clasificatorio tenga. El proceso de inclusión de variables termina cuando la inclusión de cualquiera de las variables que están fuera del modelo no mejore el mismo de manera significativa.

(ii) *Exclusión iterativa.* Se comienza con un modelo que incluye todas las variables. En cada paso se elimina una de ellas - la que menos contribuye al poder clasificatorio del modelo -. El proceso de exclusión se detiene cuando la exclusión de cualquiera de las variables que se han mantenido, hace disminuir significativamente el poder clasificatorio del modelo.

(iii) *Procedimiento mixto de inclusión - exclusión paso a paso.* Consiste en una combinación de los anteriores. En cada etapa se evalúa tanto la posibilidad de incluir una nueva variable como la exclusión de alguna de las variables ya presentes en el modelo.

(iv) *Búsqueda en el espacio de modelos.* En lugar de utilizar una aproximación que se fundamenta en un algoritmo voraz - como ocurre con cualquiera de las tres propuestas anteriores - se trata en este caso, de utilizar una técnica heurística de optimización global - algoritmos genéticos, enfriamiento estadístico, búsqueda tabú, ... - para ir moviéndose en el espacio de todos los posibles modelos, tratando de encontrar el "óptimo"para un criterio determinado.

2.5 Validación

En esta sección se tratará el problema de como estimar la bondad de un método de clasificación supervisada. Teniendo en cuenta que el objetivo de un método de clasificación supervisada es clasificar correctamente casos nuevos, no parece lógico tratar de estimar dicha bondad sobre el mismo fichero de casos que ha servido para crear el clasificador. Por otra parte aunque la tasa de error - número de errores / número de casos - es la medida más habitual del éxito de un clasificador - error entendido como clasificación incorrecta - , hay algunos dominios de aplicación en los cuales es interesante distinguir entre los dos tipos de error asociados - no todos los errores igual importancia - a un clasificador. Es por ello por lo que resulta interesante definir la denominada matriz de confusión. Dicha matriz - véase Tabla 3 - es una tabla de contingencia cruzando la variable derivada de la clasificación obtenida, con la variable que guarda la verdadera clasificación.

		Clase real		
		0 (+)	1(-)	
Clase predicha	0 (+)	a	b	p_0
	1(-)	c	d	p_1
		π_0	π_1	n

Tabla 3. Matriz de confusión para el caso de 2 clases

En algunos dominios - por ejemplo en ejemplos médicos - conviene tener presentes los conceptos de sensibilidad y especifidad, definidos a continuación:

Sensibilidad $Se = a/(a+c)$ y *Especifidad* $Es = d/(b+d)$. Es decir la sensibilidad mide la proporción de verdaderos positivos, mientras que la especifidad tiene en cuenta la proporción de verdaderos negativos. De manera análoga podemos definir la proporción de *falsos positivos* $(c/(a+c))$ o la proporción de *falsos negativos* $(b/(b+d))$.

Se denomina *tasa de error aparente* a la tasa de error obtenida por el clasificador en el conjunto de entrenamiento, mientras que el indicador de la bondad del clasificador viene habitualmente dado por la - denominada *tasa de error verdadera* - probabilidad de que el clasificador construido clasifique incorrectamente nuevos casos. Se trata de efectuar una *estimacion "honesta"de la tasa de error verdadera*, a partir de casos que constituyen una muestra aleatoria, lo cual puede llevarse a cabo por medio de los siguientes procedimientos:

- *Método H (Holdout)* Se trata de particionar la muestra aleatoriamente en dos grupos de casos: conjunto de entrenamiento, usado para inducir el modelo clasificador, y conjunto de testeo, usado para evaluar la bondad - estimar la tasa de error verdadera - del clasificador.
- *Remuestreo*
  Existen dos variantes fundamentales:
  - *Random Subsampling.* Se efectúan múltiples experimentos utilizando el método H, con particiones independientes. La estimación de la tasa de error se calcula a partir de la media de las tasas de error obtenidas en los experimentos.
  - *k-Fold Cross-Validation.* Divide el conjunto total de casos en k subconjuntos disjuntos de aproximadamente el mismo tamaño. $k-1$ de dichos subconjuntos los utiliza para entrenar el modelo, testándolo con el restante subconjunto. Esto se efectúa k veces. La estimación de la tasa de error como media de las k tasas de error obtenidas. Una variante de este procedimiento utilizada con ficheros de tamaño reducido se lleva a cabo haciendo $k = N$ (N= número de casos), y se denomina *leave-one-out.*
- *Bootstrapping.* Se escoge una muestra aleatoria con reemplazamiento del mismo tamaño que el conjunto total de casos. Se utiliza dicha muestra como conjunto de entrenamiento. Los casos no seleccionados se utilizan como conjunto de testeo. Se repite el proceso un número determinado de veces. La media de las tasas de error sirve como estimación de la tasa de error verdadera.

3 Redes Bayesianas en Clasificación

3.1 Introducción

Una aproximación Bayesiana al problema de la clasificación supervisada podría formularse de la siguiente manera:

Sean: j clase; $\boldsymbol{x}$ vector de características de un objeto; $P(j \mid \boldsymbol{x})$ probabilidad de que un objeto con características $\boldsymbol{x}$ pertenezca a la clase j

Se trata de encontrar la clase j^* verificando: $P(j^* \mid \boldsymbol{x}) = \max_j P(j \mid \boldsymbol{x})$. Utilizando el teorema de Bayes, tenemos que:

$$P(j \mid \boldsymbol{x}) = \frac{P(\boldsymbol{x} \mid j)\pi_j}{\sum_k P(\boldsymbol{x} \mid k)\pi_k}$$

donde π_k denota la probabilidad a priori de cada clase.

Existe una diferenciación clara entre las distintas aproximaciones al problema de inducir un clasificador usando el paradigma de Redes Bayesianas, en base a si el algoritmo de aprendizaje tiene en cuenta o no la existencia de una variable

especial, como es la variable que se trata de predecir. De entre las aproximaciones que vamos a introducir en este trabajo, podemos decir que exceptuando la que se expondrá en el apartado siguiente, el resto - *Naive-Bayes*, Pazzani, *Markov Blanket*, *Markov Blanket* Reducido - tienen en cuenta, en el tipo de estructura que buscan, la existencia de una variable especial, como es la variable que se trata de clasificar.

Por otra parte desde una perspectiva más general, todos los métodos de aprendizaje que se expondrán a continuación se enmarcan dentro de los denominados *métrica + búsqueda*. Es decir se propondrá una función que servirá para valorar cada una de las estructuras candidatas, y se procederá a efectuar una búsqueda dentro del espacio de posibles estructuras.

3.2 General. Métrica CH

Cualquiera de los algoritmos de aprendizaje estructural diseñados para Redes Bayesianas tanto con estructura de árbol, poliárbol o múltiplemente conectadas, que se pueden consultar en la literatura - véase por ejemplo Buntine (1996) [4], Heckerman y col. (1995) [14] - puede servir para aprender una distribución de probabilidad $p + 1$ dimensional expresable en forma de Red Bayesiana, la cual puede usarse con finalidad clasificatoria, instanciando los valores de las variables predictoras.

A modo de ejemplo comentaremos algunas características de la métrica propuesta por Cooper y Herskovits (1993) [6], ya que la misma es utilizada en los experimentos efectuados en el caso práctico que presentamos en la sección siguiente.

La manera de evaluar la bondad de una estructura de Red Bayesiana se fundamenta en el siguiente teorema probado por los autores anteriores.

Teorema 1. Sea Z un conjunto de n variables discretas. Sea una variable $X_i \in Z$ con r_i posibles valores: $(v_{i1}, \ldots, v_{ir_i})$. Sea D una base de datos conteniendo m casos, donde cada caso está especificado por la asignación de un valor para cada variable en Z. Denotamos por B_S una estructura de Red Bayesiana conteniendo exactamente las variable de Z. Cada variable X_i en B_S tiene un conjunto de variables padres, que representamos con una lista de variables denotada por Π_i. Denotemos por w_{ij} la j-ésima instanciación distinta de Π_i relativa a D. Supongamos que existen q_i instanciaciones distintas de Π_i. Definimos N_{ijk} como el número de casos en D en los cuales la variable X_i toma el valor v_{ik} y Π_i se encuentra instanciada como w_{ij}. Sea $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$.

Si dado un modelo de Red Bayesiana, se verifica que los casos ocurren de manera independiente, no existen casos con valores perdidos y la función de densidad $f(B_P|B_S)$ es uniforme, entonces se tiene que:

$P(B_S, D) = P(B_S) \prod_{i=1}^{n} g(i, \Pi_i)$, donde $g(i, \Pi_i) = \prod_{j=1}^{q_i} \frac{(r_i-1)!}{(N_{ij}+r_i-1)!} \prod_{k=1}^{r_i} N_{ijk}!$
□

Cooper y Herskovits han desarrollado un algoritmo voraz, *K2*, de aprendizaje de Redes Bayesianas, con el objetivo de encontrar la estructura de Red Bayesiana, que maximice $P(B_S, D)$. *K2* necesita definir previamente un orden total entre las variables, y asume que todas las estructuras son a priori igualmente probables. Busca para cada nodo, el conjunto de nodos padres que maximiza $g(i, \Pi_i)$. Para ello, comienza asumiendo que cada nodo no tiene ningún nodo padre, para a continuación en cada paso añadir aquel nodo padre cuya inclusión produce un mayor incremento de la probabilidad de la estructura resultante. *K2* dejará de añadir padres a un determinado nodo, cuando la adición de cualquier padre simple no incremente la probabilidad. Obviamente esta aproximación no garantiza la obtención de la estructura que tenga asociada la mayor probabilidad.

Para solventar los problemas anteriores, en nuestro grupo de trabajo, hemos desarrollado diferentes aproximaciones basadas en un heurístico de optimización global como son los algoritmos genéticos. Por una parte hemos tratado de encontrar por medio de dichos algoritmos genéticos, el mejor orden entre las variables, efectuándose la búsqueda en el espacio de órdenes, y utilizándose para ello operadores genéticos de cruce y mutación ligados al problema del viajante de comercio. Véase para más detalles, Larrañaga y col. (1996) [21]. Otra solución al problema, ha consistido en efectuar la búsqueda en el espacio de estructuras posibles de Redes Bayesianas. Para ello, si se asume un orden entre las variables, los operadores habituales de cruce y mutación genéticos resultan ser operadores cerrados - Larrañaga y col. (1996) [20] -, mientras que cuando la búsqueda se lleva a cabo sin ningún tipo de restricción en cuanto al orden de las variables, se hace necesaria la introducción de un operador de reparación - Larrañaga y col. (1996) [19] - que garantice la aciclidad de las estructuras obtenidas.

3.3 Naive-Bayes

Uno de los modelos más simples, y que por otra parte dada su facilidad de utilización se ha convertido en una especie de standard con el que comparar las bondades de los diferentes métodos, es el denominado *Naive-Bayes* (Duda y Hart 1973) [10]. Su denominación proviene de la hipótesis ingenua sobre la que se construye, es decir las variables predictoras son condicionalmente independientes dada la variable a clasificar - véase la Figura 2-. Dicha hipótesis tiene una serie de implicaciones geométricas, que han sido estudiadas por Minsky (1961) [23] - en el caso de variables predictoras binarias- y por Peot (1996)[26] - en el caso más general.

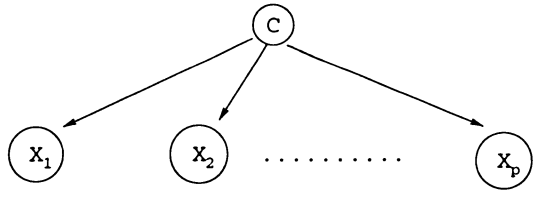


Figura 2. Naive-Bayes

La probabilidad de que el j -ésimo ejemplo pertenezca a la clase i-ésima de la variable C, puede calcularse, sin más que aplicar el teorema de Bayes, de la siguiente manera:

$$P(C = c_i \mid X_1 = x_{1j}, ..., X_p = x_{pj}) \propto P(C = c_i) \times P(X_1 = x_{1j}, ..., X_p = x_{pj} \mid C = c_i).$$

En el caso de que las variables predictoras sean condicionalmente independientes dada la variable C, se obtiene que:

$$P(C = c_i \mid X_1 = x_{1j}, ..., X_p = x_{pj}) \propto P(C = c_i) \times \prod_{r=1}^{p} P(X_r = x_{rj} \mid C = c_i).$$

El modelo *Naive-Bayes* presenta un comportamiento muy dependiente del tipo de dominio. Así por ejemplo, en dominios médicos donde el conocimiento sobre el problema es elevado y por tanto tan sólo se recoge información relativa a variables que podríamos decir que se complementan, el *Naive-Bayes* proporciona resultados aceptables, mientras que en dominios poco estructurados, en los que las variables del sistema se encuentran altamente correlacionadas, el comportamiento del *Naive-Bayes* suele ser más bien pobre.

3.4 Estructura de Árbol Aumentado. TAN

Recientemente Friedman y col. (1997) [12] presentan un método de construcción de lo que ellos denominan estructuras *TAN (Tree Augmented Naive Bayes)*, que obtiene mejores resultados que los obtenidos por el *Naive-Bayes*, a la vez que mantiene la simplicidad computacional y la robustez del anterior.

Podemos decir que un modelo *TAN* es una Red Bayesiana donde el conjunto de padres de la variable a clasificar, C, es vacio, mientras que el conjunto de variables padres de cada una de las variables predictoras, X_i, contiene necesariamente a la variable a clasificar, y como mucho otra variable. Véase por ejemplo la Figura 3.

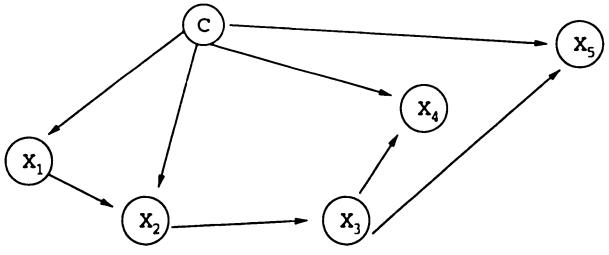


Figura 3. Estructura de Árbol Aumentado

Los anteriores autores proponen un algoritmo - adaptación del de Chow y Liu (1968) [5] - que utiliza el concepto de información mútua entre variables predictoras condicionada a la variable a clasificar. La función se define como:

$$I_P(X, Y \mid C) = \sum_{x,y,c} P(x, y, c) \log \frac{P(x, y, c)}{P(x \mid c)P(y \mid c)}.$$

De manera simple, podemos decir que la función anterior mide la información que la variable Y proporciona sobre la variable X cuando el valor de C es conocido.

El algoritmo propuesto por Friedman y col. (1997) [12] - el cual garantiza que la estructura de *TAN* obtenida tiene asociada la máxima verosimilitud entre todas las posibles estructuras de *TAN* - es como sigue:

1. Calcular $I_P(X_i, X_j \mid C)$ para cada par de variables predictoras, con $i \neq j$.
2. Construir un grafo no dirigido completo en el cual los vértices son las variables predictoras $X_1, ..., X_p$. Asignar a cada arista conectando las variables X_i y X_j un peso dado por $I_P(X_i, X_j \mid C)$.
3. Construir un árbol expandido de máximo peso.
4. Transformar el árbol resultante no dirigido en uno dirigido, escogiendo una variable raiz, y direccionando todas las aristas partiendo del nodo raiz.
5. Construir un modelo *TAN* añadiendo un nodo etiquetado como C, y posteriormente un arco desde C a cada variable predictora X_i.

3.5 Pazzani

Pazzani (1996) [24] presenta un modelo que puede considerarse que se posiciona en un lugar intermedio entre los modelos extremos, en los que, por una parte se tienen que calcular las $(r-1)2^p$ distribuciones de probabilidad - para el caso

de que la variable C admita r posibles valores, y las variables predictoras sean dicotómicas -, es decir el modelo necesita las siguientes probabilidades:

$$P(C = c_i \mid X_1 = x_{1j}, ..., X_p = x_{pj})$$

y por otra parte el modelo que hemos denominado *Naive-Bayes*, en el cual se hace necesario el calcular:

$$P(C = c_i \mid X_1 = x_{1j}, ..., X_p = x_{pj}) \propto P(C = c_i) \times \prod_{r=1}^{p} P(X_r = x_{rj}),$$

y por tanto no necesitaríamos más que $(r-1)+p$ probabilidades.

Veamos escuetamente lo propuesto por Pazzani, apoyándonos en un simple ejemplo. Supongamos un dominio con 4 variables predictoras X_1, X_2, X_3, X_4 y una variable a predecir C. Supongamos asimismo que la variable X_2 no es relevante para C, y que además las variables X_1 y X_3 son condicionalmente dependientes dada C. Tendríamos una situación que gráficamente puede ser expresada según la Red Bayesiana central de la Figura 4.

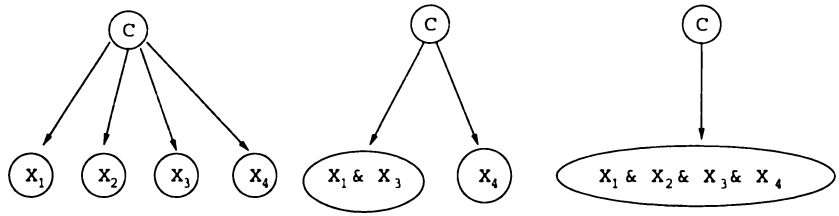


Figura 4. Pazzani

A nivel de fórmulas lo expresaríamos:

$$P(C = c_i \mid X_1 = x_{1j}, ..., X_4 = x_{4j}) \propto$$

$$P(C = c_i) \times P((X_1 = x_{1j}, X_3 = x_{3j}) \mid C = c_i) \times P(X_4 = x_{4j} \mid C = c_i).$$

Lo que queda por determinar es que variables son no relevantes, y por otra parte que variables van a agruparse y necesitar que se calcule para las mismas las probabilidades condicionadas correspondientes.

Pazzani propone para la selección del modelo dos algoritmos voraces, siguiendo la filosofía Estadística de modelización hacia adelante y modelización hacia atrás. Exponemos a continuación los pasos a seguir en uno de ellos.

Algoritmo FSSJ (Forward Sequential Selection and Joining)

1. Inicializar el conjunto de variables a utilizar a vacio. Clasificar todos los ejemplos en la clase más frecuente.
2. **Repetir** en cada paso la mejor operación entre:
   (a) Considerar cada variable no usada como un nueva variable a incluir en el modelo, condicionalmente independiente de las variables ya incluidas, dada la variable a clasificar.
   (b) Juntar cada variable no utilizada con una variable ya incluida en el clasificador.

   Evaluar cada clasificador candidato por medio de *leave-one-out*.
3. **Hasta que** ninguna operación produzca mejoras.

El procedimiento de búsqueda del modelo propuesto por el algoritmo anterior puede mejorarse si en lugar del mismo, la búsqueda se llevase a efecto por medio de un algoritmo que al menos de manera teórica garantice que el modelo creado es el óptimo global.

3.6 Markov Blanket

Teniendo en cuenta que en una Red Bayesiana - véase Figura 5 - cualquier variable tan sólo se encuentra influenciada por el denominado *Markov Blanket* relativo a la misma - es decir por el conjunto de sus variables padres, sus variables hijas, y por las variables que son padres de las hijas -, parece intuitivo tener en cuenta modelos clasificatorios que sean *Markov Blanket* de la variable a clasificar.

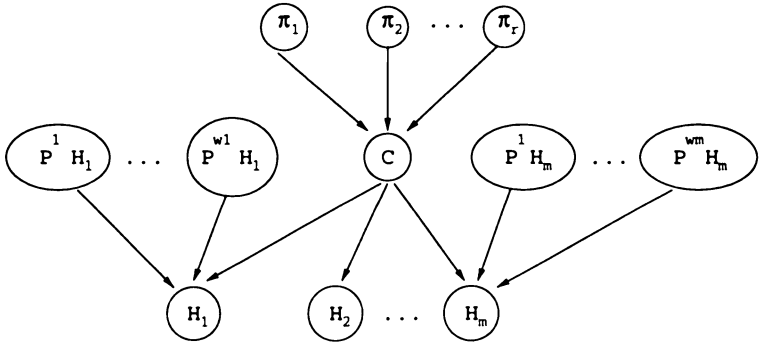


Figura 5. Markov-Blanket

El concepto de *Markov Blanket* asociado a una variable se ha utilizado en el denominado muestreo de Gibbs - véase por ejemplo Pearl (1987) [25] -, y puede ser establecido de manera formal por el siguiente teorema.

Teorema 2. La distribución de probabilidad de cada variable, X_i, en una Red Bayesiana, condicionada por el estado del resto de las variables, se puede obtener a través de la siguiente expresión:

$$P(x_i|\mathbf{Z}_{x_i}) = \alpha P(x_i|\pi_{x_i}) \prod_j P(\omega_{ij}|\pi_{ij}(x_i))$$

donde α es una constante normalizadora, independiente de X_i, y $x_i, \mathbf{Z}_{x_i}, \pi_{x_i}, \omega_{ij}$ and $\pi_{ij}(x_i)$ denotan respectivamente instanciaciones consistentes de $X, Z_X = Z - X, \Pi_X, \Omega_j$ y Π_{ij}, siendo Z el conjunto de todas las variables, Π_i el conjunto de variables padres de X, Ω_i el conjunto de variables hijas de X, y Π_{ij} el conjunto de padres de Ω_i. □

Existen varios procedimientos para buscar dentro del espacio de posibles *Markov Blanket* de la variable a clasificar. Por ejemplo Sierra y Larrañaga (1998) [28] utilizan los algoritmos genéticos para llevar a efecto tal búsqueda.

3.7 Markov Blanket Reducido

Debido a los problemas de sobreajuste - constatados en algunos experimentos - del que sufre la aproximación anterior, se pueden reducir las condiciones impuestas por el modelo anterior, con el objetivo de obtener Redes Bayesianas más simples pero a la vez con mayor poder generalizador. Para ello - véase Sierra y Larrañaga (1998) [28] - se pueden, por ejemplo, efectuar las siguientes dos relajaciones del modelo anterior:

1. No todas las variables tienen por que ser necesariamente parte del *Markov Blanket* de la variable a clasificar.
2. Una variable que sea padre de la variable a clasificar no puede ser padre de un hijo de la variable a clasificar, y además una variable tan sólo puede ser padre de un hijo de la variable a clasificar.

4 Predicción de la supervivencia en Melanoma

4.1 Introducción

Un área de interes dentro de la Inteligencia Artificial trata de comparar empíricamente las bondades de diferentes algoritmos de inducción de modelos, proveniendo estos tanto de la Estadística como del Aprendizaje Automático. A lo largo

de los últimos 5 años se han producido numerosos intentos de caracterización de las bondades y deméritos de los distintos algoritmos. El lector interesado puede consultar, por ejemplo, el trabajo desarrollado dentro del proyecto *ESPRIT Stat-Log* (Michie y col (1993) [22]), o más recientemente el llevado a cabo por Cooper y col. (1997) [7].

En esta sección presentamos los resultados de la aplicación de algunos de los modelos descritos en la sección anterior, al problema de la predicción de la supervivencia en pacientes aquejados de cáncer de piel maligno.

A pesar de los avances desarrollados en los últimos años en el tratamiento del cáncer, el pronóstico de pacientes que han desarrollado melanoma, ha cambiado muy poco. Por otra parte la incidencia de la enfermedad ha crecido continuamente en la última década, y en caso de que se siga produciendo la reducción de la capa de ozono, no es de esperar una disminución del número de casos relacionados con esta enfermedad.

Los resultados de estudios experimentales sugieren dos factores de riesgo fundamentales: la exposición al sol, junto con las características fenotípicas del individuo. Así por ejemplo, la exposición continua al sol multiplica por 9 el riesgo de padecer la enfermedad, mientras que si dicha exposición se hace de manera intermitente pero intensamente, dicho riesgo se ve incrementado por un factor de 5.7.

El melanoma de piel maligno es un tumor no muy común en nuestro entorno. Se relacionan con el mismo entre el 8% y el 10% de los tumores malignos que afectan a la piel. De acorde con el Registro de Cáncer del País Vasco, (Izarzugaza (1994) [16]), en 1990 la tasa de incidencia era de 2.2 por cada 100.000 hombres, incrementándose dicha cantidad al 3 por 100.000 para las mujeres.

La base de datos analizada contenía 311 casos - diagnosticados en el Instituto Oncológico de Gipuzkoa entre el 1 de Enero de 1988 y el 31 de Diciembre de 1995 -, para cada uno de los cuales se tenía información recogida en su mayor parte en el momento del diagnóstico y relativa a 8 variables. Las 5 variables predictoras son: sexo (2 categorias), edad (5 categorias), estadío (4 categorias), densidad (4 categorias) y número de nodos positivos (2 categorias). La variable a predecir tiene dos categorias y tiene en cuenta si la persona sobrevive o no, una vez transcurrido una año, tres años y cinco años desde el momento del diagnóstico.

4.2 Resultados obtenidos

Se han tenido en cuenta 4 modelos a la hora de efectuar los experimentos. En primer lugar hemos inducido una Red Bayesiana usando algoritmos genéticos para efectuar la búsqueda, y tomando como función objetivo la propuesta por Cooper y Herskovits. El segundo modelo trata de encontrar la mejor estructura de *Markov*

Blanket de la variable a clasificar, siendo el objetivo del algoritmo genético el maximizar el porcentaje de casos correctamente clasificados. El tercer modelo se relaciona con una relajación del concepto de *Markov Blanket*, se obtiene por medio de un algoritmo genético que guia la búsqueda, tratando de encontrar el *Markov Blanket* de la variable a clasificar que tenga asociado el mayor porcentaje de casos bien clasificados. Finalmente el cuarto modelo es el denominado *Naive Bayes*. En todos los modelos las estimaciones del porcentaje de individuos bien clasificados, que se muestra en la Tabla 4, se ha obtenido por medio de validaciones cruzadas *(10-fold cross-validation)*. La propagación de la evidencia se ha llevado a cabo por medio del software *HUGIN*.

Supervivencia del Melanoma de Piel Maligno			
	1 año	3 años	5 años
CH-GA	93.06%	81.95%	69.57%
MB	94.28%	83.90%	78.88%
MBR	93.47%	83.85%	74.53%
N-B	91.43%	79.02%	71.43%

Tabla 4. Estimación del porcentaje de bien clasificados para la predicción de la supervivencia al año, a los tres años y a los cinco años desde el diagnóstico

5 Conclusiones

Después de una breve introducción a distintos paradigmas clasificadores - Análisis Discriminante, Regresión Logística, Árboles de Clasificación, *K-NN*, Inducción de Reglas, Redes Neuronales - así como de las distintas medidas de calidad - porcentaje de bien clasificados, sensibilidad, especificidad, .. - de los clasificadores, y de distintos métodos de validación de los mismos - método H, validaciones cruzadas, *bootstrapping* -, se han introducido las distintas aproximaciones al problema basadas en Redes Bayesianas.

Entre los modelos propuestos dentro del paradigma de Redes Bayesianas, se han tratado los siguientes: el General, *Naive-Bayes*, Pazzani, *Markov Blanket*y *Markov Blanket* Reducido.

Finalmente se han mostrado los resultados obtenidos con varios de los modelos anteriores en un problema de clasificación con datos reales provenientes de un dominio médico. Se ha tratado de predecir la supervivencia de individuos aquejados

de melanoma maligno al año, a los tres años y a los cinco años del momento del diagnóstico. Se han efectuado estimaciones de la probabilidad de individuos bien clasificados a partir del *10-fold cross-validation*.

Por lo que respecta a posibles líneas de trabajo relacionadas con este problema, citaremos varias. En primer lugar parece interesante - ya que en buena parte de las aplicaciones prácticas así ocurre - que el paradigma sea capaz de tratar adecuadamente *información temporal*, es decir que pueda trabajar con datos longitudinales. Por otra parte, dada la magnitud - tanto en número de casos, como en número de variables predictoras - de algunas aplicaciones reales, una línea de trabajo consiste en desarrollar procedimientos que efectúen de manera automática tanto la *selección de las variables predictoras* - procedimientos independientes del paradigma, asi como que tengan en cuenta el inductor a utilizar (*wrappers*) -, como la *selección de casos* con los que efectuar el aprendizaje. Una tercera línea de investigación radica en el desarrollo de *modelos híbridos* que conjuguen las bondades de más de un paradigma. Asi por ejemplo se podria pensar en paradigmas que conjugasen los Árboles de Clasificación con las Redes Bayesianas, las cuales se construirían en cada una de las hojas terminales del árbol. Otra manera de hibridar podría ser utilizar la información proporcionada por el experto para reducir el espacio de búsqueda del paradigma. Finalmente, una línea de trabajo que está siendo estudiada por distintos grupos investigadores, consiste en el desarrollo de *multiclasificadores* que conjugan de manera adecuada la información proporcionada por varios modelos construidos para los mismos datos.

6 Agradecimientos

Este trabajo se ha efectuado gracias a las subvenciones concedidas por el CICYT (TIC97-1135-C04-03), así como por el Gobierno Vasco - Departamento de Educación, Universidades e Investigación (PI 96/12).

Referencias

1. Aha, D., Kibler, D., Albert, M. (1991). Instance-based learning algorithms. *Machine Learning*, **6(1)**, 37-66.
2. Bishop, C. M. (1996). *Neural networks for pattern recognition.* Oxford University Press.
3. Breiman, L., Freidman, J. H., Olshen, R. A., Stone, C. J. (1984). *Classification and Regression Trees.* Wadsworth.
4. Buntine, W. (1996). A guide to the literature on learning graphical models. *IEEE Transactions on Knowledge Data Engeniering.* **8**, 195-210.
5. Chow, C. K. , Liu, C. N. (1968). Approximating discrete probability distributions. *IEEE Transactions on Information Theory*, **14**, 462-467.
6. Cooper, G. F., and Herskovits, E.A. (1993). A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, **9**, 309-347.
7. Cooper, G. F., Aliferis, C. F., Ambrosino, R., Aronis, J., Buchanan, B. G., Caruana, R., Fine, M. J., Glymour, C., Gordon, G., Hanusa, B. H., Janosky, J. E., Meek, C., Mitchell, T., Richardson, T., Spirtes, P. (1997). An evaluation of machine-learning methods for predicting pneumonia mortality. *Artificial Intelligence in Medicine*, **9 (2)**, 107-138.
8. Covert, T. M., Hart, P. E. (1967). Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, **13 (1)**, 21-27.
9. De Jong, K. A., Spears, W. M., Gordon, F. D. (1993). Using genetic algorithms for concept learning. *Machine Learning*, **13**, 161-188.
10. Duda, R. O., Hart, P. E. (1973) *Pattern classification and scene analysis.* John Wiley Sons.
11. Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, **7**, 179-188.
12. Friedman, N., Geiger, D., Goldszmidt, M. (1997). Bayesian Network Classifiers. *Machine Learning*, **29**, 131-163.
13. Hand, D. J. (1997). *Construction and Assessment of Classification Rules.* John Wiley Sons.
14. Heckerman, D., Geiger, D., Chickering, D. (1995). Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, **20**, 197-243.
15. Holland, J. (1975). *Adaptation in Natural and Artificial Systems.* University of Michigan Press.
16. Izarzugaza, M.I. (1994). Informe del registro de Cáncer de Euskadi 1990. *Osasunkaria*, 8-11.
17. Kaufman, L., Rousseeuw, P. J. (1990). *Finding Groups in Data.* John Wiley Sons.
18. Kleinbaum, D. G. (1994). *Logistic Regression. A Self-Learning Text.* Springer-Verlag.
19. Larrañaga, P., Poza, M., Yurramendi, Y., Murga, R., and Kuijpers, C. (1996). Structure Learning of Bayesian Networks by Genetic Algorithms: A Performance Analysis of Control Parameters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **18**, 912-926.

20. Larrañaga, P., Murga, R., Poza, M., and Kuijpers, C. (1996). Structure Learning of Bayesian Networks by Hybrid Genetic Algorithms. *Learning from Data: AI and Statistics V, Lecture Notes in Statistics 112.* D. Fisher, H.-J. Lenz (eds.), Sprigер-Verlag, 165-174.
21. Larrañaga, P., Kuijpers, C., Murga, R., and Yurramendi, Y. (1996). Learning Bayesian Network Structures by searching for the best ordering with genetic algorithms. *IEEE Transactions on System, Man and Cybernetics*, **26**, 487-493.
22. Michie, D., Spiegelhalter, D. J., Taylor, C. C. (1994). *Machine Learning, Neural and Statistical Classification.* Ellis Horwood.
23. Minsky, M. (1961). Steps toward Artificial Intelligence. *Transactions of IRE*, **49**, 8-30.
24. Pazzani, M. J. (1996). Searching for dependencies in Bayesian classifiers. *Learning from data: Artificial intelligence and statistics V*, D. Fisher, H.-J. Lenz (eds.), Springer-Verlag, 239-248.
25. Pearl, J. (1987). Evidential Reasoning Using Stochastics Simulation of Causal Models. *Artificial Intelligence*, **32**, 245-257.
26. Peot, M. A. (1996). Geometric Implications of the Naive Bayes Assumption. *Uncertainty in Artificial Intelligence. Proceedings of the Twelfth Conference*, Horvitz, E., Jensen, F. (eds.), Morgan Kaufmann, 414-419.
27. Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning*, **1**, 81-106.
28. Sierra, B., Larrañaga, P. (1998). Predicting the survival in malignant skin melanoma using Bayesian networks automatically induced by genetic algorithms. An empirical comparision between different approaches. *Artificial Intelligence in Medicine.* En prensa.
29. Venturini, G. (1993). SIA: a supervised inductive algorithm with genetic search for learning attribute based concepts. *Proc. European Conference on Machine Learning*, Vienna, Austria, 280-296.

Modelos Gráficos para la Toma de Decisiones

Concha Bielza[1] y David Ríos Insua[2]

[1] Grupo de Análisis de Decisiones
Dpto. Inteligencia Artificial
Universidad Politécnica de Madrid
Campus de Montegancedo s/n
Boadilla del Monte, 28660 Madrid.
correo-e: mcbielza@fi.upm.es
[2] ESCET
Universidad Rey Juan Carlos
Móstoles, 28936 Madrid.
correo-e: d.rios@escet.urjc.es

Resumen

El Análisis de Decisiones proporciona el único marco coherente para la resolución de problemas de toma de decisiones. En problemas complejos, típicamente debemos proceder a modelizar el problema con ayuda de una representación gráfica. En el capítulo se revisan las principales ideas sobre modelos gráficos para la toma de decisiones, con énfasis en los diagramas de influencia.

1 Introducción

El resto de capítulos del curso se refieren básicamente a modelización de incertidumbre y problemas de inferencia basados en redes bayesianas, siendo las aplicaciones típicas a problemas de diagnóstico y predicción. Nosotros nos centraremos en problemas de toma de decisiones en condiciones de incertidumbre.

El marco que adoptamos para la modelización y resolución de estos problemas es el del Análisis de Decisiones (AD). Suponemos que una persona o grupo de personas tiene que elegir una alternativa de un conjunto. El problema de decisión al que se enfrentan es complejo, debido a la presencia de factores como objetivos múltiples y conflictivos, la presencia de incertidumbre, un entorno cambiante,... En estas condiciones, se requiere un marco racional para resolver estos problemas, como el proporcionado por el AD.

El responsable de la decisión o *decisor* es normalmente capaz de dar sus objetivos –no muy bien definidos– e ideas sobre las consecuencias de las distintas acciones. Aporta información sobre lo que espera que revelará el futuro, basado en experiencias previas, y sobre sus preferencias y actitudes frente al riesgo. La presencia de incertidumbre le obliga a tomar decisiones sin conocer con seguridad

determinados factores que no se controlan (estados). El AD le ayudará a organizar toda esta información de forma coherente, calculando su mejor curso de acción, dadas sus creencias y objetivos, consiguiendo, a su vez, que adquiera confianza y mayor profundización en el problema, al comprender las implicaciones y posibles inconsistencias de sus juicios [7].

Específicamente, el marco del AD, basado en el marco normativo de la Teoría de la Decisión, sugiere:

- modelizar las creencias del decisor sobre la ocurrencia de los estados θ mediante una distribución de probabilidad $\pi(\theta)$, que, en presencia de información adicional $f(x|\theta)$ se actualiza mediante la fórmula de Bayes $\pi(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\int f(x|\theta)\pi(\theta)d\theta}$;
- modelizar las preferencias del decisor sobre las consecuencias y sus actitudes frente al riesgo mediante una función de utilidad (afín única) $u(a,\theta)$, que indica la utilidad obtenida cuando se toma la alternativa a y se da el estado θ;
- escoger la alternativa de máxima utilidad esperada, i.e., aquella a que resuelve el problema de optimización $\max_{a\in\mathcal{A}} \int u(a,\theta)\pi(\theta|x)d\theta$, donde $\mathcal{A}$ es el conjunto de alternativas.

El resto del capítulo se estructura como sigue. Introducimos en la siguiente sección el proceso del AD y tres métodos de representación de problemas de AD. La sección 3 analiza en detalle la resolución de problemas de AD representados mediante un diagrama de influencia. En la sección 4 exponemos algunas aplicaciones reales de estos métodos, indicando el software existente en la actualidad. Concluimos comentando temas avanzados, algunos aún en desarrollo, y planteando algunas cuestiones abiertas.

2 Introducción al Análisis de Decisiones

El proceso del AD comienza con la modelización cualitativa del problema, es decir, la estructura del proceso de decisión con sus elementos: objetivos, alternativas y fuentes de incertidumbre. El primer aspecto importante es por tanto proporcionar técnicas para representar toda esa información. Introducimos en esta sección los principales métodos de representación de problemas de decisión. Matemáticamente son equivalentes, pero desde el punto de vista práctico hay grandes diferencias entre ellos. Revisamos tres métodos: tablas de decisión, árboles de decisión, y diagramas de influencia. Otros métodos se mencionan en la sección 5.

La especificación completa del problema debe incluir también el conocimiento cuantitativo sobre él, reflejando los valores que toman sus elementos (decisiones e incertidumbres), las relaciones entre ellos, y los juicios del decisor a través de probabilidades y utilidades. En esta sección expondremos la forma en que esos métodos de representación incluyen tal información cuantitativa.

2.1 Tablas de decisión

Las *tablas de decisión* constituyen la forma más elemental de representación de un problema de decisión. Permiten ilustrar los conceptos básicos de forma sencilla, pero son muy limitadas desde el punto de vista práctico.

Definición 1. (Tabla de decisión)
La idea básica que expresa una tabla de decisión es que las consecuencias de elegir una decisión no dependen sólo de ésta, sino también de una serie de factores externos no controlables por el decisor y parcialmente desconocidos por él al tomar la decisión. A los valores que pueden tomar tales factores externos los llamaremos *estados*. Si el decisor conociese el verdadero estado podría predecir la consecuencia de su elección con certeza:

$$\text{decisión} + \text{estado} \rightarrow \text{consecuencia}$$

Si $\mathcal{A}$ es el conjunto de decisiones a, Θ es el conjunto de estados θ (exhaustivo y mutuamente excluyente) y $\mathcal{C}$ es el conjunto de consecuencias c, la asignación se representa

$$a + \theta \rightarrow c(a, \theta) \in \mathcal{C}$$

que, escrito en forma tabular, constituirá la tabla de decisión. □

Ejemplo 1. (Inversión)
Consideremos un problema de inversión en el que un decisor tiene tres opciones: $\mathcal{A}$ ={Bonos del Estado, Petróleos, Telefónicas}. El retorno de las inversiones después de un año depende del estado futuro de la economía que puede ser Θ = {malo, regular, bueno, muy bueno}. Este problema de inversión se expresa mediante la tabla 1. Por ejemplo, si invertimos en Petróleos y la economía va bien, el retorno será del 9%.

	Malo	Regular	Bueno	Muy Bueno
BE	14	14	14	14
Pet	0	1	9	20
Tel	3	10	15	17

Tabla 1. Tabla de decisión para el problema de inversión

□

Los valores de la tabla que indican consecuencias podrán ser, en general, preferencias sobre las consecuencias, expresadas mediante una función de utilidad. Así, en el ejemplo 1, la función de utilidad de un retorno r podría venir dada, por ejemplo, por $u(r) = r^2$. La asignación de la función de utilidad se lleva a cabo con métodos estándar de la Teoría de la Utilidad [12]. Se comprueba primero que las preferencias del decisor satisfacen los axiomas requeridos por esta teoría, y en tal caso se asigna la utilidad de algunas consecuencias (con métodos como los de [12]). Después, mediante herramientas de análisis numérico se ajusta una función a los datos obtenidos. La concavidad/convexidad de esta función indicará aversión/afición al riesgo. Se han identificado ciertas formas funcionales adecuadas para modelizar tales actitudes frente al riesgo (exponenciales, logarítmicas,...) y se han determinado condiciones, denominadas de independencia de preferencias, que aseguran un cierto tipo de descomposición de la función, facilitando la tarea en casos más complejos, como el de consecuencias vectoriales.

Además de las preferencias, han de asignarse las creencias del decisor, según se indicó en la introducción. La asignación de probabilidades (subjetivas), como se ha visto en otros capítulos, sigue un procedimiento que está incluido dentro de un protocolo general realizado conjuntamente con expertos, con fases de motivación, preparación del problema, eliminación de sesgos, asignación y validación (ver el proceso SRI y sus extensiones en e.g. [21]). Puede utilizarse una *rueda de la fortuna* u otro mecanismo de aleatorización. En el ejemplo 1, el inversor tendrá que revelar sus creencias sobre el estado futuro de la economía, pudiendo ser, e.g., π(malo)=1/6, π(regular)=1/4, π(bueno)=1/2, y π(muy bueno)=1/12.

2.2 Árboles de decisión

La representación de un problema de decisión mediante tablas es estática, con un solo momento de elección. Sin embargo, los problemas de decisión reales suelen ser dinámicos, existiendo varias decisiones encadenadas a tomar. La representación de este encadenamiento de decisiones y de la ocurrencia de distintos fenómenos aleatorios resulta engorrosa mediante tablas, pues implicaría la enumeración de las posibles estrategias. Por ello, adoptamos una representación alternativa más flexible y expresiva: la de árboles de decisión.

Definición 2. (Árbol de decisión)
Un *árbol de decisión* es un árbol con tres tipos de nodos:

- de decisión (nodos rectangulares), del que emergen ramas que representan las decisiones posibles que se pueden tomar en ese instante;
- de azar (nodos circulares), cuyas ramas representan los estados posibles que se pueden dar en ese instante;

– de valor (terminales), que representan la utilidad de las consecuencias asociadas a la sucesión de decisiones y estados desde el nodo raíz hasta ese nodo.

□

Para construir el árbol se comienza identificando el nodo raíz, que corresponderá al primer hecho que se observa en el tiempo: una toma de decisión o la presencia de un factor de incertidumbre. Se continúa desde la raíz incluyendo nodos de azar o de decisión marcando los distintos caminos a seguir, hasta alcanzar un nodo terminal, en el que se indicará la consecuencia correspondiente. Así, el árbol equivalente a la tabla de decisión del ejemplo 1 sería el de la figura 1. Nótese en los nodos terminales, la presencia de las utilidades de las consecuencias de cada camino del árbol, y en los nodos de azar, las probabilidades de los posibles estados de la economía.

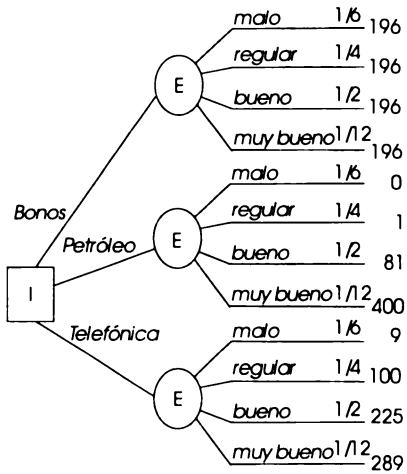


Figura 1. Árbol de decisión para el problema de inversión

Veamos otro ejemplo que se utilizará a lo largo del capítulo.

Ejemplo 2. (Reactor)
Una compañía eléctrica debe decidir (D_2) si construye un reactor de diseño convencional (c) o de diseño avanzado (a). El reactor avanzado conlleva más riesgo pero, en caso de éxito, proporciona más beneficios. Los beneficios en unidades monetarias para c son 8, si no falla (ce) una vez construido (con probabilidad

.98), y -4, si falla (cf) (.02). Los beneficios para el avanzado son 12, si no falla (ae) (.660), -6 si tiene un accidente de importancia leve (al) (.244), y -10, si ocurre un accidente de importancia mayor (am) (.096). Antes de tomar esta decisión, la compañía puede realizar un test ($D_1 = t$), con coste 1 UM, o no hacerlo ($D_1 = nt$), sobre las componentes críticas del reactor de diseño avanzado, que reducirá la incertidumbre sobre el mismo. Los resultados del test (T) se pueden clasificar en malos (m), buenos (b) o excelentes (e) y están muy relacionados con el éxito o fracaso del reactor avanzado. La figura 2 describe un modelo probabilístico causal para A y T. Si los resultados del test son malos, la opción avanzada no es viable y se construirá un reactor convencional.

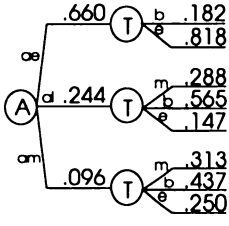


Figura 2. Modelo causal probabilístico para A y T en el problema del reactor

En este ejemplo, el proceso secuencial de decisión es: primero, decidir si se realiza o no el test, observar después sus resultados (en caso de realizarlo), y, a continuación, decidir qué tipo de reactor se va a construir. Finalmente, se desvelará el estado en que se encuentra el reactor escogido. La figura 3 proporciona el árbol de decisión asociado. Nótese que las probabilidades $\pi(A|T)$ y $\pi(T)$ incluidas en los nodos de azar A y T, no son las del enunciado, y se han tenido que calcular para poder ser representadas en el árbol. Además, para compactar la representación, se ha aprovechado la replicación del subárbol C, dibujándose sólo una vez. Los costes del test aparecen en la primera decisión, contribuyendo también a esta replicación o *coalescencia*, ver [5].

Se podría pensar que parece lógico que la compañía deba construir el reactor avanzado (convencional) después de obtener resultados excelentes (buenos), pero no siempre es así. Dependerá de las creencias específicas de la compañía sobre el resultado de cada reactor y sobre cómo percibe la fiabilidad del test, así como de la valoración de las posibles consecuencias. Los cuatro nodos D_2 de la figura 3 no son idénticos: el conocimiento de la compañía es diferente en cada caso y, por tanto, sus creencias sobre el estado de cada reactor diferirán. □

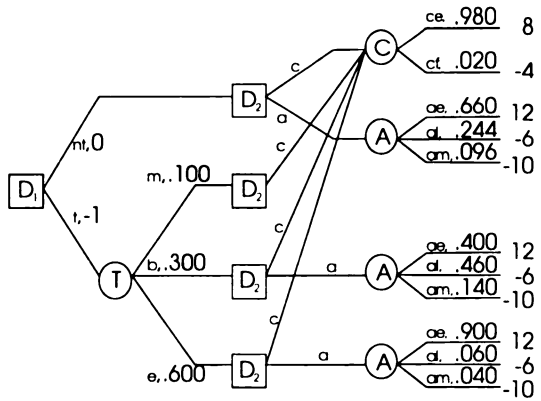


Figura 3. Árbol de decisión para el problema del reactor

A pesar de la capacidad descriptiva de los árboles de decisión, que indican explícitamente la cronología del proceso de decisión y el estado de información disponible en cada instante de decisión, se vuelven excesivamente complejos cuando aumenta el número de nodos de azar y/o de decisión. Cada nodo añadido al árbol expande su tamaño exponencialmente, de forma que solamente pueden mostrarse a nivel de detalle modelos pequeños y relativamente simples. Por ejemplo, sería infactible representar un problema con 30 variables, entre nodos de decisión y de azar. Se han propuesto algunas soluciones, e.g., representar el árbol de forma esquemática, ver [5], pero se pierde mucha información en la mayoría de los casos (por ejemplo, cuando se dan estructuras altamente asimétricas y dependientes). El árbol esquemático del ejemplo 2 se muestra en la figura 4.

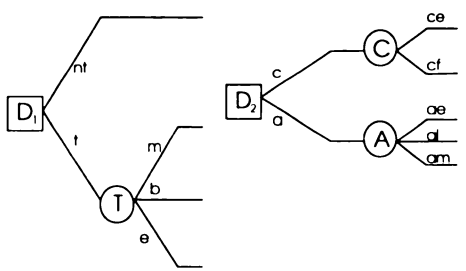


Figura 4. Árbol esquemático para el problema del reactor

2.3 Diagramas de influencia

Los diagramas de influencia salvan muchos de los inconvenientes de los árboles de decisión. Inicialmente fueron concebidos como método de representación más compacta de problemas [14], que después se traducían a un árbol para ser evaluados. Posteriormente se crearon algoritmos de evaluación que operan con el propio diagrama, e.g. [25]. Hoy en día constituyen un lenguaje gráfico de modelización que puede utilizarse tanto en Análisis de Decisiones como en Inferencia probabilística, como en [26].

Definición 3. (Diagrama de influencia)
Un *diagrama de influencia* (DI) es un grafo dirigido $G = (N, A)$ donde:

- el conjunto N de nodos, se particiona en conjuntos D, C, V que designan, respectivamente,
  - D, al conjunto de nodos de decisión (rectangulares), que modelizan decisiones a tomar;
  - C, al conjunto de nodos de azar (circulares), que modelizan, como antes, cantidades inciertas que influyen en el problema;
  - V, al conjunto de nodos de valor (romboidales), que modelizan las utilidades (esperadas);
- el conjunto A de arcos, incluye arcos de dos clases, dependiendo del tipo de nodo al que van dirigidos:
  - *informativos*, si van a nodos de decisión, e implican precedencia temporal, es decir, la variable en el origen del arco es información disponible y conocida en el momento de tomar la decisión que se encuentra en el nodo destino del arco;
  - *condicionales*, si van a nodos de valor o de azar, y representan dependencia, funcional o probabilística, respecto de los valores de los nodos antecesores.

□

Definición 4. Los predecesores directos de un nodo i de valor o de azar se denominan *predecesores condicionales* $C(i)$; los de un nodo i de decisión, *predecesores informativos* $I(i)$. □

El DI del ejemplo 1 de inversión es demasiado simple como para deducir sus ventajas respecto al árbol, ver la figura 5. En la figura 6 se muestra el DI del ejemplo 2. La información de los arcos indica que se toma antes D_1 que D_2, se conoce T al elegir D_2, la distribución de probabilidad de T está condicionada por A, y también por T, y la función de utilidad depende de las cuatro variables que

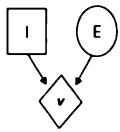


Figura 5. Diagrama de influencia del problema de inversión

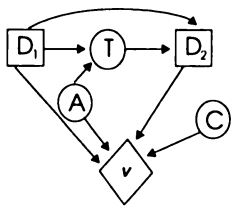


Figura 6. Diagrama de influencia del problema del reactor

tienen arco hacia el nodo de valor v. Con los DI la representación del problema es mucho más compacta. Cada variable añadida al problema expande su tamaño linealmente.

Dibujado el grafo que representa la descripción cualitativa del problema, se procede a incluir la información cuantitativa. Para cada nodo i del DI, se especifica un conjunto Ω_i, una variable X_i, y una aplicación. Si i es de decisión, X_i es la decisión que se toma del conjunto Ω_i; si i es de azar, X_i es la variable aleatoria asociada con espacio muestral Ω_i, sobre la que se define la distribución de probabilidad dada por $\pi_i(x_i|x_{C(i)})$; si i es de valor, X_i es la utilidad esperada, con dominio en $\Omega_{C(i)}$ y la aplicación es $U : \Omega_{C(i)} \rightarrow \Omega_i$, la utilidad esperada en función de los predecesores directos.

La tabla 2 contiene la información de los nodos de decisión y de azar para el problema del reactor; la tabla 3, la del nodo de valor.

D_1
nt
t

D_2
c
a

	C
ce	.98

	A
ae	.660
al	.244

	t			nt		
T	ae	al	am	ae	al	am
m	0	.288	.313	0	0	0
b	.182	.565	.437	0	0	0
nr	0	0	0	1	1	1

Tabla 2. Tablas de los nodos de decisión y azar del problema del reactor

Nótese que se ha añadido a T el estado nr que indica 'no hay resultados' y que permite construir $\pi(T|D_1, A)$, ya que sólo se observan resultados si se lleva a cabo el test. De esta forma *simetrizamos* el problema.

D_1	D_2	A	C	v
nt	c	ae	ce	8
			cf	-4
		al	ce	8
			cf	-4
		am	ce	8
			cf	-4
	a	ae	ce	12
			cf	12
		al	ce	-6
			cf	-6
		am	ce	-10
			cf	-10

D_1	D_2	A	C	v
t	c	ae	ce	7
			cf	-5
		al	ce	7
			cf	-5
		am	ce	7
			cf	-5
	a	ae	ce	11
			cf	11
		al	ce	-7
			cf	-7
		am	ce	-11
			cf	-11

Tabla 3. Tabla del nodo de valor del problema del reactor

3 Evaluación de diagramas de influencia

Describimos en esta sección cómo resolver un problema de decisión modelizado mediante un DI. En los árboles de decisión, la idea básica es que supuesto que hemos tomado ciertas decisiones y se han observado ciertos estados hemos llegado a un nodo que podrá ser: 1) terminal, y le asignamos la utilidad de la consecuencia; 2) de azar, y le asignamos la utilidad esperada máxima a partir de ese nodo; 3) de decisión, y le asignamos la utilidad esperada de la decisión de máxima utilidad esperada a partir de ese nodo. Este procedimiento se aplica recursivamente hacia atrás (en sentido contrario al temporal), utilizando programación dinámica.

En los DI, la idea es esencialmente la misma, pero aprovechando la estructura gráfica del diagrama para obtener ventajas computacionales. Gráficamente, el diagrama experimenta una serie de transformaciones que no modifican la política óptima ni la máxima utilidad esperada. Numéricamente, estas transformaciones redefinen las aplicaciones asociadas a cada nodo, calculando en los de decisión, las soluciones óptimas del problema. Las transformaciones son esencialmente aplicaciones del principio de la programación dinámica y de la fórmula de Bayes. Veamos

primero unas definiciones que necesitaremos posteriormente, y después las transformaciones y el algoritmo.
Suponemos que el DI es regular y orientado.

Definición 5. Un DI es *orientado* si tiene exactamente un nodo de valor. □

Definición 6. Un DI es *regular* si

1. es acíclico
2. el nodo de valor no tiene sucesores
3. existe un camino dirigido que contiene todos los nodos de decisión.

□

La propiedad 3 de la definición anterior requiere un orden total de las decisiones. Como consecuencia, cualquier información disponible en el momento de tomar una decisión debe también estarlo en decisiones posteriores.

Proposición 1. Si el nodo de decisión i precede al nodo de decisión j en un DI regular, entonces $\{i\} \cup I(i) \subset I(j)$. □

Esta propiedad requerirá normalmente la adición de arcos, llamados *de memoria*, que hagan explícito ese orden total de las decisiones. Obsérvese que mientras en el árbol de decisión esto estaba implícito, en el DI del ejemplo 2 (ver figura 6) se ha añadido el arco de memoria (D_1, D_2).

Definición 7. (Nodo sumidero)
Un nodo es un *sumidero* si no tiene sucesores y es de azar o de decisión. □

En general, cualquier nodo que no sea predecesor indirecto del nodo de valor puede considerarse un sumidero. Obviamente pueden eliminarse los sumideros de un DI regular y orientado, lo que constituye la primera transformación. A continuación se exponen las otras tres.

- *Eliminación de nodo de azar.* Si el nodo de azar i precede únicamente al nodo de valor v, puede eliminarse por esperanza condicionada, heredando v los predecesores de i.
- *Eliminación de nodo de decisión.* Si no hay sumideros, el nodo de decisión $i \in C(v)$, verificando $C(v) \backslash \{i\} \subset I(i)$, puede eliminarse maximizando la utilidad esperada (condicionada), registrándose la mejor decisión. v no hereda los predecesores de i, pudiendo aparecer, por tanto, nuevos sumideros.
- *Inversión de arcos.* Dado el arco (i, j) entre los nodos de azar i y j, si no existe otro camino dirigido entre i y j, puede sustituirse por el arco (j, i) mediante la aplicación del teorema de Bayes, con herencia mutua de predecesores.

El siguiente teorema justifica el paso más importante del algoritmo de evaluación:

Teorema 1. (Existencia de nodo de azar eliminable)
Si v tiene predecesores pero no puede eliminarse ningún nodo de decisión, existe un nodo de azar que es predecesor condicional de v pero no es predecesor informativo de ningún nodo de decisión, y puede eliminarse (tal vez tras inversión de arcos).
Demostración: Puede verse en [25]. □

Aplicando de forma reiterada estas transformaciones se tiene un procedimiento que eliminará todos los nodos hasta que sólo quede el de valor. En ese momento se han calculado las decisiones óptimas (acumuladas en cada nodo de decisión) y la máxima utilidad esperada (acumulada en v).

Si $S(i)$ designa al conjunto de sucesores directos del nodo i, el algoritmo de evaluación de DIs, debido a Shachter [25], es:

1. Verificar que el DI es regular, orientado, y añadir arcos de memoria
2. Eliminar sumideros
3. Mientras $C(v) \neq \emptyset$,
   Si $\exists i \in C \cap C(v) : S(i) = \{v\}$, eliminar nodo de azar i
   si no, si $\exists i \in D \cap C(v) : C(v) \backslash \{i\} \subset I(i)$,
   eliminar nodo de decisión i
   eliminar sumideros creados
   si no, encontrar $i \in C \cap C(v) : D \cap S(i) = \emptyset$
   mientras $C \cap S(i) \neq \emptyset$
   encontrar $j \in C \cap S(i)$: $\nexists$ otro camino de i a j
   invertir (i, j)
   eliminar nodo de azar i

Utilizamos el algoritmo de Shachter para resolver el ejemplo 2 del reactor. En el primer paso se puede eliminar tanto A como C. Todas las posibles secuencias de borrado conducen a la solución final pero involucran un esfuerzo computacional diferente. Existen heurísticas para encontrar una buena secuencia [17], ya que encontrar la óptima es un problema NP-completo. Escogemos C como primer nodo a eliminar. El diagrama resultante sería análogo al inicial pero sin el nodo C y sin su arco hacia v. La tabla almacenada en v queda modificada como indica la tabla 4.

El paso siguiente es eliminar A, invirtiendo antes el arco (A, T). La figura 7 muestra los diagramas resultantes, donde se observa que v hereda el nodo T al eliminar A. La tabla 5 indica las operaciones realizadas. Los cálculos de la inversión del arco (A, T) se obtienen de las fórmulas:

$$\pi(T|D_1) = \sum_A \pi(T|D_1, A)\pi(A) \quad \text{y} \quad \pi(A|D_1, T) = \frac{\pi(T|D_1, A)\pi(A)}{\pi(T|D_1)}.$$

D_1	D_2	A	v
nt	c	ae	7.76
		al	7.76
		am	7.76
	a	ae	12
		al	-6
		am	-10
t	c	ae	6.76
		al	6.76
		am	6.76
	a	ae	11
		al	-7
		am	-11

Tabla 4. Tabla del nodo de valor después de eliminar C

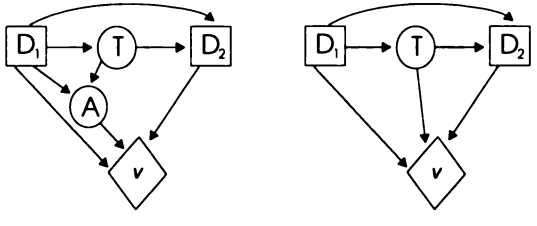


Figura 7. Eliminación del nodo A

Procedemos ahora a eliminar D_2, expresando en cada situación las decisiones óptimas a tomar en D_2, mediante la función Ψ_{D_2} (ver tabla 6). Después se elimina T y finalmente D_1. En la figura 8 se observan las transformaciones del diagrama: en *(a)* eliminamos D_2; en *(b)* T, y, en *(c)*, D_1.

En la tabla 6 encontramos la solución para D_1 y la máxima utilidad esperada del problema, que es 8.128. Por tanto, se aconseja realizar el test, y si los resultados son excelentes construir el reactor de tipo avanzado; en caso contrario, el

	D_1	
T	t	nt
m	.1	0
b	.3	0
nr	0	1

	t				nt			
A	m	b	e	nr	m	b	e	nr
ae	0	.40	.90	x	x	x	x	.660
al	.7	.46	.06	x	x	x	x	.244

D_1	T	D_2	v
nt	m	c	7.76
		a	-4x
	b	c	7.76
		a	-4x
	e	c	7.76
		a	-4x
	nr	c	7.76
		a	5.496

D_1	T	D_2	v
t	m	c	6.76
		a	-8.2
	b	c	6.76
		a	-.36
	e	c	6.76
		a	9.04
	nr	c	6.76
		a	-7x

Tabla 5. Tablas de inversión y eliminación de A

D_1	T	v	Ψ_{D_2}
nt	m	7.76	c
	b	7.76	c
	e	7.76	c
	nr	7.76	c
t	m	6.76	c
	b	6.76	c
	e	9.04	a
	nr	6.76	c

D_1	v
nt	7.76
t	8.128

v	Ψ_{D_1}
8.128	t

Tabla 6. Tablas de eliminación de D_2, T y D_1

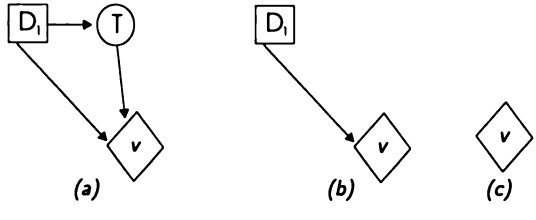


Figura 8. Eliminación de nodos *(a)* D_2, *(b)* T, *(c)* D_1

convencional. Observemos que el nodo de valor actúa como el nodo terminal en un árbol, acumulando la máxima utilidad esperada en cada paso.

En la práctica, los problemas de decisión reales son complejos y es necesario el uso de software que realice los cálculos, ver sección 4. Pero a menudo, incluso estos métodos exactos son incapaces de resolverlos y sólo podemos obtener aproximaciones a soluciones óptimas. Los métodos gráficos hasta ahora vistos tienen problemas computacionales al manejar, por ejemplo, variables continuas de azar y/o de decisión: el teorema de Bayes y los cálculos de esperanzas con variables continuas requieren típicamente integración numérica, y maximizar la utilidad esperada sobre una variable de decisión continua requiere usualmente una búsqueda iterativa. Estos dos problemas incrementan la carga computacional en problemas de gran dimensión.

Una posibilidad es utilizar simulación. En otros capítulos se ha visto que existen numerosos métodos de simulación para Redes Bayesianas, por ejemplo [22]. Por contra, sólo conocemos algún esbozo de un método de simulación en AD, en [16] y [6], si bien tales métodos resultan intratables en presencia de espacios de decisión continuos.

En [1] se propone un método de Montecarlo para resolver problemas de AD. En él, se considera una distribución artificial aumentada sobre el espacio producto de decisiones y estados, de forma que su marginal en el espacio de decisiones es proporcional a la utilidad esperada de la decisión y, en consecuencia, la solución óptima coincide con la moda de la marginal.

Específicamente, si consideramos el DI genérico de la figura 9, el problema de resolución del DI se formula

$$\max_d V(d) = \max_d \int u(d, x, \theta, y)\, dp_d(\theta, y|x)$$

donde

$$p_d(\theta, y|x) \propto p(\theta)p(x|\theta)p_d(y|\theta). \tag{1}$$

Aumentamos ahora la medida de probabilidad dada en (1) a un modelo de probabilidad para (θ, y, d) definiendo una función de densidad conjunta

$$h(\theta, y, d) \propto u(d, x, \theta, y) \cdot p_d(\theta, y|x),$$

suponiendo que u es positiva y acotada. La moda de la distribución marginal $h(d) \propto \int\int u(d, x, \theta, y)p_d(\theta, y|x)\, d\theta\, dy = V(d)$ corresponde a la decisión óptima d^*.

Se sugiere entonces la siguiente aproximación:

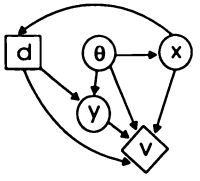


Figura 9. DI genérico

1. Tomar una muestra $(\theta^1, y^1, d^1),...,(\theta^n, y^n, d^n)$ de la distribución aumentada $h(\theta, y, d)$.
2. Marginalizar a una muestra $(d^1, ..., d^n)$ de $h(d)$.
3. Hallar la moda de esta muestra.

Los pasos clave son 1 y 3. Para 3, acudimos principalmente a las herramientas del análisis exploratorio de datos para identificar aproximadamente d^*. Para 1, ya que no será posible en general muestrear directamente de la distribución artificial, introducimos varios métodos de simulación de Montecarlo con cadenas de Markov (MCCM), ver por ejemplo [29], que proporcionan una muestra aproximada. Los métodos MCMC construyen una cadena de Markov de la que es directo simular y cuya distribución de equilibrio es la distribución deseada, en nuestro caso $h(\theta, y, d)$. Entonces, si simulamos la cadena durante un periodo largo de tiempo, eliminando los valores transitorios de la fase inicial, podemos usar los valores simulados de la cadena como muestra aproximada de la distribución deseada.

El método es aplicable a DIs con estructura no secuencial, es decir, los nodos de decisión no pueden tener como predecesores a nodos de azar que tengan a su vez distribuciones que dependen de otros nodos de decisión, aunque en [1] se da alguna idea sobre cómo extender el método a DI secuenciales. No hay más requisitos, pudiendo ser continuas y no conjugadas las distribuciones de probabilidad, los espacios de decisiones continuos y la función de utilidad arbitraria.

El ejemplo 2 del reactor nos permite ilustrar el procedimiento. Con la notación del DI genérico, el problema incluye dos nodos de decisión $d = (D_1, D_2)$, donde $D_1 \in \{nt, t\}$ y $D_2 \in \{c, a\}$ y tres nodos de azar: $y_1 = T \in \{m, b, e\}$, $y_2 = C \in \{ce, cf\}$ e $y_3 = A \in \{ae, al, am\}$, correspondiendo al resultado del test, los accidentes del reactor convencional y los accidentes del reactor avanzado, respectivamente. No hay parámetros desconocidos θ. El problema es secuencial ya que la decisión D_2 puede depender del valor de y_1 y de la primera decisión D_1. Para adaptarlo a los requisitos de nuestro algoritmo de simulación, reemplazamos la segunda decisión por una lista de nodos de decisión con un nuevo nodo separado, correspondiente a cada escenario posible de los nodos predecesores. Por

tanto, reemplazamos D_2 por el par (D_2^{nb}, D_2^e), donde D_2^e es el tipo de reactor a escoger si $D_1 = t, y_1 = e$; y D_2^{nb} es el tipo a escoger en otro caso. Nótese que la decisión bajo $(D_1 = t, y_1 = m)$ está fijada por la compañía como $D_2 = c$. El nodo de decisión D_2^{nb} se podría partir más, en D_2^n, D_2^b, correspondiendo a las decisiones a tomar si $(D_1 = nt)$ y $(D_1 = t, y_1 = b)$, respectivamente. Sin embargo, esto no es necesario, pues D_1 ya separa estos dos escenarios en el sentido siguiente: $(D_1 = t, D_2^{nb})$ es la decisión que corresponde a realizar el test y obtener un buen resultado; $(D_1 = nt, D_2^{nb})$ es la que corresponde a no realizar el test. La figura 10 resume el problema.

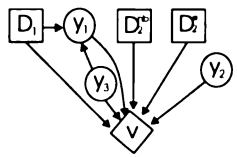


Figura 10. DI modificado para el problema del reactor

El algoritmo propuesto es el siguiente:

1. Comenzar con valores (d^0, y^0) arbitrarios. Hacer $i = 1$.

   Hasta que se juzgue convergencia

2. Evaluar $u^1 = u(d^{i-1}, y^{i-1})$.
3. *Actualizar* (d, y)
   (a) Generar

$$(\tilde{d}, \tilde{y}) \sim g(\tilde{d}|d)p_{\tilde{d}}(\tilde{y}) = g(\tilde{d}|d)p_{\tilde{d}}(\tilde{y}_1|y_3)p(\tilde{y}_2)p(\tilde{y}_3) \tag{2}$$

   (b) Evaluar $\tilde{u} = u(\tilde{d}, \tilde{y})$.
   (c) Calcular

$$a = \min\left\{1, \frac{h(\tilde{d}, \tilde{y})}{h(d^{i-1}, y^{i-1})} \frac{p_{d^{i-1}}(y^{i-1})}{p_{\tilde{d}}(\tilde{y})}\right\} = \min\left\{1, \frac{\tilde{u}}{u^1}\right\}.$$

   (d) Hacer

$$(d^i, y^i) = \begin{cases} (\tilde{d}, \tilde{y}) & \text{con probabilidad } a, \\ (d^{i-1}, y^{i-1}) & \text{en otro caso.} \end{cases}$$

4. Hacer $i = i + 1$.

Los pasos 3(a) y 3(b) implementan una cadena de independencia, usando $p_{\tilde{d}}(\tilde{y})$ como distribución de prueba. Sólo necesitamos una distribución de prueba g, la función de utilidad para la evaluación y algoritmos para generar de $p_d(y)$, lo que será, en general, factible, ya que estas distribuciones se definen explícitamente en el DI, ver (2).

Para la distribución de prueba, dada la naturaleza discreta de d, tomamos aleatoriamente, con probabilidad 1/6, uno de los seis elementos de d. La tabla 7 muestra las probabilidades estimadas $h(d)$, después de 100000 iteraciones del algoritmo.

D_1	D_2^{nb}	D_2^{e}	$h(d)$
nt	*c*		0.178
nt	*a*		0.158
t	*c*	*c*	0.17
t	*c*	*a*	0.182
t	*a*	*c*	0.15
t	*a*	*a*	0.162

Tabla 7. Probabilidades marginales estimadas $h(d) = s \cdot (V(d) + \delta)$, con una traslación común $\delta = +11$ y escala s (desconocida)

Como $h(d) \propto V(d)$, vemos en la tabla 7 que la decisión óptima es la misma que ya obtuvimos. Este problema es muy sencillo pero el método de simulación propuesto permite la adaptación directa a estructuras mucho más complejas. Por ejemplo, el modelo de probabilidad podría extenderse a distribuciones a priori más complicadas para los parámetros de $p(y_1)$, $p(y_2)$ y $p(y_3)$, sin incrementar la complejidad del algoritmo de simulación; también, la función del beneficio podría venir dada como un modelo más complicado de predicción.

4 Aplicaciones y software

Las técnicas vistas en este capítulo para estructurar y resolver problemas de AD requieren ser implementadas para su utilización en problemas reales que, en general, son de gran tamaño y complejidad. Estas dificultades no deben conducirnos a que el modelo se ajuste a la técnica escogida, más que a las necesidades del decisor. La implantación en ordenadores conduce a *sistemas de ayuda a la decisión* [4], con módulos que abarcan todas las fases del ciclo del AD, y a *sistemas de decisión inteligentes* [13], conjugándose entonces con los sistemas expertos. En

esta sección describimos dos aplicaciones reales que hemos desarrollado y recomendamos software representativo de utilización actual para el AD.

4.1 Software

Del software existente en el mercado destacamos los siguientes programas: *Logical Decision* [19], se utiliza para la asignación de utilidades. Para modelizar problemas utilizando árboles de decisión es recomendable *Supertree* [20], escrito en APL, siendo el primer paquete de AD completo comercializado, mientras que para DI destacamos *InDia* [15], escrito en Pascal. Sin embargo, resultan mucho mejores los programas que utilizan técnicas mixtas, aprovechando las ventajas de cada uno de los dos métodos, como hacen *DPL* [11], y *DATA* [10], escritos en C y C++, respectivamente, y ejecutándose en entornos WINDOWS.

4.2 Aplicaciones

Describimos brevemente dos aplicaciones desarrolladas en dominios específicos: gestión de la ictericia neonatal y gestión de embalses.

Ejemplo 3. (Ictericia neonatal)
La ictericia ocurre cuando la bilirrubina aumenta en el sistema sanguíneo en lugar de ser excretada desde el hígado hasta el intestino y fuera del cuerpo. Caracterizada por un aspecto amarillento de la piel, la ictericia es muy frecuente en los recién nacidos porque el hígado está aún inmaduro y no funciona normalmente. Se debe distinguir la ictericia fisiológica de su versión más grave, la ictericia patológica, con la que la hiperbilirrubinemia puede dañar el cerebro y el sistema nervioso central si no se trata, pudiendo producir incluso la muerte. No existe consenso respecto a cuándo es mejor comenzar el tratamiento, es decir, en qué situación el nivel de bilirrubina es lo suficientemente alto como para requerir tratamiento. En [24] se describe en detalle la modelización del problema mediante un DI.

La figura 11 muestra el DI. Observemos el gran tamaño del grafo al tratarse de un problema muy complejo. Los nodos sombreados indican que aparecen dos veces en el diagrama, para no tornarlo más engorroso. La primera decisión a tomar es si se ingresa o no al niño, conociendo en ese momento ciertos aspectos suyos y de la madre (edad, peso, tipo de parto, grupos sanguíneos, factores Rh, concentraciones de bilirrubina y hemoglobina), así como resultados de algunos test que dan indicios de enfermedades relacionadas con el problema, como la asfixia perinatal y la isoinmunización. En caso de ser ingresado, se trata al paciente en varias etapas hasta que mejora. Las terapias posibles son la observación, la fototerapia (exposición a luz que mitiga el exceso de bilirrubina), o la exanguinotransfusión (cambio completo de la sangre), la cual entraña un riesgo alto de

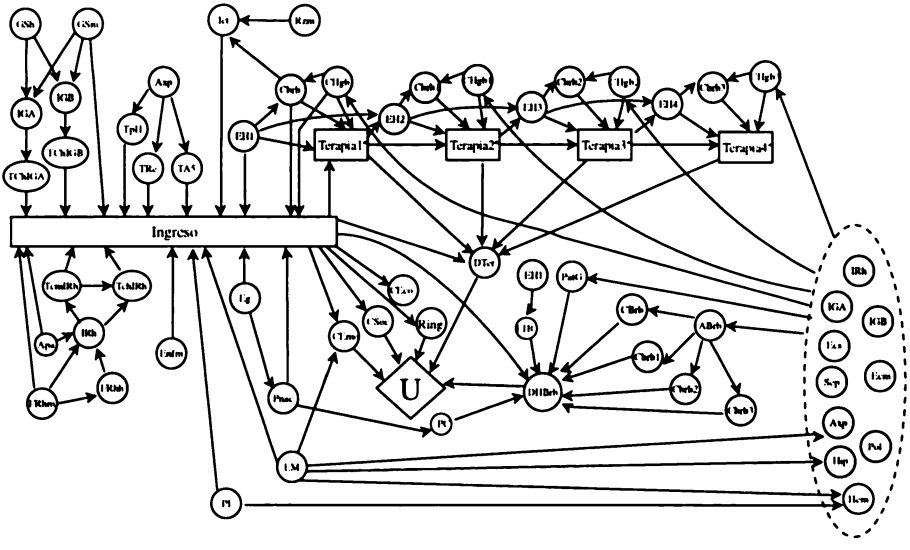


Figura 11. Diagrama de influencia del problema de la ictericia

mortalidad, entre otros riesgos detectados. Las enfermedades que se consideran aparecen encerradas a la derecha del grafo, representando por simplicidad sólo una vez, los arcos comunes a todas, que emergen de cada una. Las consecuencias se valoran en el nodo U, que depende de aspectos tales como coste económico, social, emocional (experimentado por los padres), riesgos derivados del ingreso, daños debidos al tratamiento, y debidos a la hiperbilirrubinemia. Para todas, salvo el coste económico, se definieron escalas construidas para cuantificarlas.

Se escogió el DI para su estructuración pues si se representara mediante un árbol, tendría del orden de 10^{18} nodos terminales (si se tienen en cuenta todos los caminos incluyendo los imposibles), haciendo del árbol un método gráfico ineficaz e inmanejable. El sistema creado para este problema es IctNeo, que gestiona la representación del problema y su evaluación, y a través de un interfaz de usuario muestra los resultados al médico, incorporando explicaciones. □

El otro sistema desarrollado es BayRes [23], para resolver problemas de gestión de embalses. Consta de tres módulos: el primero es para predecir las entradas al embalse, utilizando modelos dinámicos lineales; el segundo cuantifica las preferencias del gestor mediante una función de utilidad multiatributo; el tercero resuelve el problema buscando las decisiones (e.g., cantidad de agua a soltar por aliviaderos y por turbinas) de máxima utilidad esperada. La búsqueda exacta de

éstas es infactible para problemas con un horizonte amplio de planificación, por lo que adoptamos una estrategia que busca decisiones buenas, en el sentido de no separarse demasiado de una trayectoria de referencia marcada por el gestor y guiada por el sistema.

5 Temas avanzados y cuestiones abiertas

En las secciones anteriores hemos hecho una breve introducción sobre algunos aspectos básicos de los modelos gráficos para toma de decisiones, con énfasis en los DIs. Existen otras cuestiones y temas abiertos que en estas breves líneas sólo se pueden mencionar puntualmente.

En primer lugar, existen otros muchos modelos gráficos interesantes en el AD. Mencionemos las redes de evaluación [28] y los diagramas de decisión secuenciales [9]. Un problema que hemos vislumbrado en los ejemplos es el de la asimetría. Numerosos problemas de decisión reales son asimétricos, en el sentido de que, supuesto lo representamos mediante un árbol, no todos los caminos de la raíz a un nodo terminal, siguen la misma secuencia de nodos. Bielza y Shenoy [3] dan una descripción completa del problema de la asimetría con los distintos formalismos gráficos.

Hemos descrito un método muy potente de detección de la alternativa óptima en un diagrama, basada en simulación de un modelo aumentado de probabilidad. Una alternativa es convertir el diagrama en una red bayesiana, utilizando el método de Cooper [8] y aplicar alguno de los métodos descritos en el resto de capítulos. Sin embargo, tal método requiere, esencialmente, la evaluación de la utilidad esperada para cada alternativa, con lo que nuestro método resulta más eficiente. Otra ventaja de este método es su aplicabilidad a problemas continuos, con modelos de probabilidad no conjugados, funciones de utilidad arbitrarias,... El análisis de tales problemas continuos es, en general, muy complejo, salvo en modelos gaussianos, ver [27]. Una posibilidad en este caso es utilizar algún tipo de heurística, como la miope modificada por una trayectoria de referencia como en [23].

La dificultad esencial surge de las dificultades de la programación dinámica para afrontar problemas de decisión secuencial estocásticos. De hecho, sería deseable la extensión de nuestro método de simulación a problemas secuenciales. Algunas posibilidades se apuntan en [1]. Otras posibilidades residen en el empleo de esquemas inteligentes de discretización, o resultados de métodos recientes en programación dinámica, como la programación dinámica neural o el método de alcanzabilidad.

Hemos indicado que un uso fundamental de estos métodos es el desarrollo de sistemas de decisión inteligentes, para ayudar en la toma de decisiones en situa-

ciones que se repitan. Una vez construido tal sistema puede ocurrir que debamos tratar casos parcialmente observados, bien en nodos de azar, bien en nodos de decisión. Para estas situaciones deben desarrollarse métodos similares a los de actualización de redes probabilísticas, ver [18].

También puede ocurrir que los diagramas estén parcialmente especificados, en el sentido de que se dispongan de restricciones sobre las utilidades y las probabilidades. El decisor se siente más cómodo dando, por ejemplo, un rango para las utilidades/probabilidades al tener una capacidad de discriminación finita. En tal caso, deberían proporcionarse esquemas de cálculo de políticas no dominadas, como en [2]. Tal método proporciona un primer paso hacia el desarrollo de una aproximación al *análisis de sensibilidad* en DIs, un tema en el que aún falta mucho por hacer. Con el análisis de sensibilidad, se acometen tareas de revisión y reasignación de las utilidades y probabilidades estudiando qué implicaciones tienen sobre las decisiones óptimas.

Agradecimientos Este trabajo ha sido financiado por la CICYT, TIC 95-0028, y por la Consejería de Educación y Cultura de la CAM.

Referencias

1. Bielza, C., Müller, P., Ríos Insua, D. Monte Carlo methods for Decision Analysis with applications to influence diagrams. Techn. Rep., DP 96-07, Duke University, *ISDS Paper*, 1996.
2. Bielza, C., Ríos Insua, D., Ríos Insua, S. Influence diagrams under partial information, en J.M. Bernardo, J.O. Berger, A.P. Dawid y A.F.M. Smith (eds.). *Bayesian Statistics 5*, pp. 491–497, Oxford U.P., 1996.
3. Bielza, C., Shenoy, P.P. A comparison of graphical techniques for asymmetric decision problems, *WP-271*, School of Business, Univ. of Kansas, 1996. (Aparecerá en *Management Science*, en 1998).
4. Bonczek, R.H., Holsapple, C.W., Whinston, A.B. *Foundations of Decision Support Systems*. Academic Press, New York, 1981.
5. Call, H.J., Miller, W.A. A comparison of approaches and implementations for automating Decision Analysis. *Reliability engineering and system safety*, 30, pp. 115–162, 1990.
6. Charnes, J., Shenoy, P. A forward Monte Carlo method for solving influence diagrams using local computation. *WP-273*, School of Business, Univ. of Kansas, 1996.
7. Clemen, R.T. *Making hard decisions: an introduction to Decision Analysis*. PWS-Kent, Boston, 1997.
8. Cooper, G. A method for using belief networks as influence diagrams. *Fourth Workshop on Uncertainty in Artificial Intelligence*, pp. 55-63, 1988.
9. Covaliu, Z., Oliver, R.M. Representation and solution of decision problems using sequential decision diagrams. *Management Science*, 41, 12, pp. 1860–1881, 1995.

10. *DATA 3.0 User's manual.* Treeage Software, Inc., Williamstown, MA, 1996.
11. *DPL, advanced version user guide.* ADA Decision Systems, Duxbury, Belmont, CA, 1995.
12. French, S., Ríos Insua, D. *Statistical Decision Theory.* Arnold, 1998.
13. Holtzman, S. *Intelligent decision systems.* Addison-Wesley, Mass., 1989.
14. Howard, R.A., Matheson, J.E. Influence diagrams, en R.A. Howard and J.E. Matheson (eds.). *The principles and applications of Decision Analysis*, 2, pp. 719–762, Strategic Decisions Group, Menlo Park, CA., 1981.
15. *InDia, User's Guide*, version 2.0. Decision Focus, PWS-Kent, Boston, MA, 1991.
16. Jenzarli, A. Solving influence diagrams using Gibbs sampling. Tech. Rep., College of Business, University of Tampa, FL, 1995.
17. Kong, A. *Multivariate Belief Functions and Graphical Models*, Tesis Doctoral, Dpt. of Statistics, Harvard University, MA, 1986.
18. Lauritzen, S.L., Spiegelhalter, D.J. Local computations with probabilities on graphical structures and their applications to expert systems. *Jour. Roy. Stat. Soc.* B, 50, 2, pp. 157–224, 1988, (con discusión).
19. *Logical Decision*, Multimeasure Decision Analysis Software V. 4.106, Golden, CO, 1996.
20. McNamee, P., Celona, J. *Decision Analysis with Supertree.* Scientific Press, San Francisco, CA, 1990.
21. Merkhofer, M.W. Quantyfying judgmental uncertainty: methodology, experiences, and insights. *IEEE Trans. on Syst., Man, and Cyber.*, 17, 5, pp. 741–752, 1987.
22. Ríos Insua, D., Ríos Insua, S., Martín, J. *Simulación.* RA-MA, Madrid, 1997.
23. Ríos Insua, D., Bielza, C., Martín, J., Salewicz, K. BayRes: a system for stochastic multiobjective reservoir operations. *Advances in Multiobjective and Goal Programming*, Springer, 1997.
24. Ríos Insua, S., Bielza, C., Gómez, M., Fernández del Pozo, J.A., Sánchez Luna, M., Caballero, S. An intelligent decision system for jaundice management in newborn babies, en F.J. Girón (ed). *Case Studies in Decision Analysis, Lectures Notes in Economics and Mathematical Systems*, Springer, aparecerá en 1998.
25. Shachter, R.D. Evaluating influence diagrams. *Operations Research*, 34, 6, pp. 871–882, 1986.
26. Shachter, R.D. Probabilistic inference and influence diagrams. *Operations Research*, 36, pp. 589–604, 1988.
27. Shachter, R.D., Kenley, C.R. Gaussian influence diagrams, *Management Science*, 35, 5, pp. 527–550, 1989.
28. Shenoy, P.P. Valuation-based systems for Bayesian decision analysis. *Operations Research*, 40, 3, pp. 463–484, 1992.
29. Tierney, L. Markov chains for exploring posterior distributions. *Ann. Statist.*, 22, pp. 1701-1762, 1994, (con discusión).

Modelos Gráficos Dinámicos

José M. Puerta[1]

Dpto. Informática
Universidad de Castilla-La Mancha
Albacete. 02071
correo-e: jpuerta@info-ab.uclm.es

Resumen

En este trabajo abordaremos el estudio de los modelos gráficos dinámicos para representar sistemas estocásticos temporales. En primer lugar justificaremos la utilización de las redes de creencia dinámicas como modelo gráfico para representar y manejar los sistemas dinámicos. Identificaremos los problemas clásicos que se plantean en los sistemas dinámicos y plantearemos su solución mediante la utilización de redes de creencia dinámicas. Por último aplicaremos todo lo estudiado anteriormente a los problemas de planificación y control, para ello definiremos el problema y plantearemos su solución mediante los modelos de redes estudiados y finalizaremos con una aplicación concreta como ilustración a todo lo anterior.

1 Introducción

La mayoría de las investigaciones en razonamiento probabilístico se han centrado en la construcción y uso de modelos fundamentalmente estáticos, en los cuales las relaciones temporales entre las variables del modelo son fijas e invariantes en el tiempo. Las predicciones o cálculos de las probabilidades a posteriori dado un conjunto de observaciones no varían con el tiempo. En estos modelos estáticos se tiene solo en cuenta las observaciones actuales para predecir el estado del sistema sin posibilidad de tener en cuenta la historia de la evolución temporal de las observaciones en el sistema.

Aunque algunos problemas dinámicos se pueden resolver con modelos estáticos parece más razonable considerar en el modelo resultante la evolución temporal del sistema, con los medios necesarios para poder actualizar las relaciones que dependan del tiempo. En comparación con los modelos estáticos, una consideración temporal en el modelo enriquicería el mismo con la información de la tendencia temporal del sistema así como con métodos para poder actualizar el modelo en respuesta a las observaciones de la historia del proceso evolutivo del sistema.

Ejemplo 1. Vamos a suponer que tenemos la red de creencia descrita en la figura 1 para monitorizar el proceso de siembra y recolección del trigo.

En dicha red tenemos la información necesaria para poder obtener mediante algoritmos de inferencia, por ejemplo el método Hugin, de manera eficiente $P(CPS|T, ES, SNV)$ o $P(F|T, ES, SNV)$, etc. Hemos de notar, sin embargo que estas distribuciones de probabilidad a posteriori son válidas solo para un periodo de tiempo determinado, por ejemplo una semana, ya que claramente existen variables en el sistema que dependen del tiempo. Por otra parte, en un proceso dinámico han de tenerse en cuenta no solo las evidencias que tenemos en este instante de tiempo sino también la llegada de nueva evidencia, además de la actual, para un proceso de razonamiento.

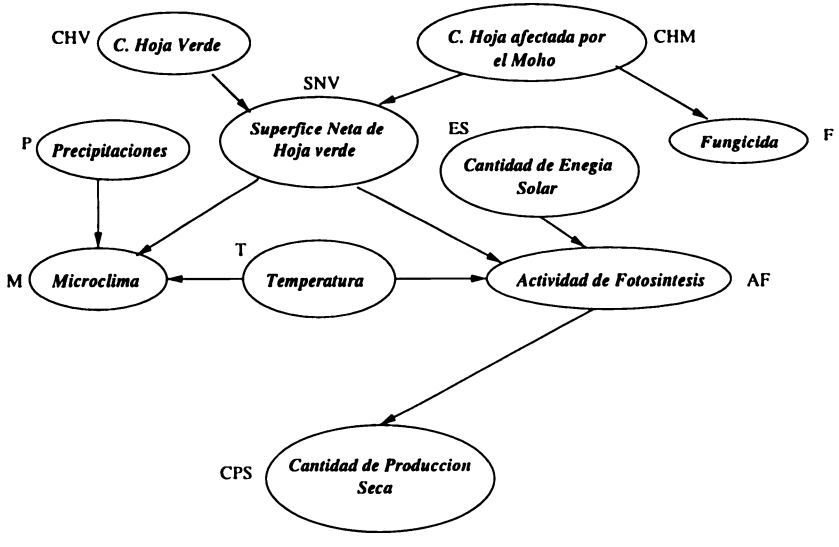


Figura 1. Red de creencia para un sistema de producción de trigo.

En primer lugar, como hemos comentado, parece razonable que un modelo gráfico dinámico pueda tener en cuenta las observaciones históricas, sin embargo existe otra razón por la que es recomendable ampliar el modelo de la red de creencia. Si observamos de nuevo la estructura de la red de la figura 1, parece lógico establecer la siguiente relación: La cantidad de hoja afectada por el moho es causa directa para la superficie neta de hoja verde y ésta a su vez influye directamente en el microclima, pero de nuevo éste influiría de manera directa en

la cantidad de hoja afectada por el moho. Este tipo de relaciones no se pueden representar directamente en una red de creencia ya que se establecería un bucle dirigido no permitido.

Por estas razones ha de plantearse la utilización de un modelo gráfico que permita representar sin ningun problema las situaciones que hemos planteado y realizar procesos de razonamiento válidos.

Recientemente se ha desarrollado una extensión de las redes de creencia, las redes de creencia dinámicas, que caracterizan la evolución temporal del sistema mediante un modelo de evolución que establece las dependencias temporales entre las variables del sistema en cuestión. Para nuestro ejemplo, una modelización temporal del sistema de producción podría parecerse al de la figura 2. Con este modelo desde el instante de tiempo t_0 hasta t_{n-1}, es decir, desde el inicio del proceso dinámico hasta el final del mismo, nos permitiría solventar las cuestiones que planteábamos en un principio, estas son, tener en cuenta la evolución histórica de las observaciones y por otra parte establecer relaciones temporales entre las variables de una manera explícita.

□

Se han desarrollado modelos dinámicos para el razonamiento temporal probabilístico, tales modelos pueden ser aplicados a un gran campo de aplicaciones como la predicción, control, planificación, problemas de simulación, etc. En este trabajo nos centraremos en el estudio de las técnicas de inferencia con modelos gráficos dinámicos probabilísticos.

Los investigadores en el campo de la estadística han desarrollado numerosos métodos para razonar sobre las relaciones temporales entre las variables que describen un modelo. Este campo, generalmente conocido como ***análasis de series temporales***, es una colección de muestras de un proceso evolutivo estocástico consistente en un conjunto de observaciones que se realizan secuencialmente conforme evoluciona el tiempo. Se han obtenido buenos métodos para resolver este tipo de problemas cuando las relaciones temporales que se describen no son complejas y son lineales. Solo recientemente se han unido de alguna forma este último campo y el del estudio de la representación del conocimiento incierto mediante redes de creencia, dando lugar al modelo de red de creencia dinámica.

En general, en un modelo dinámico, consideraremos un conjunto de variables aleatorias $X(t_k)$, que describen el estado del mundo en el instante de tiempo, discreto, t_k, como por ejemplo la variable temperatura del ejemplo anterior. En estos modelos nos interesan conocer las creencias relacionadas con un mundo cambiante. Si tenemos la evolución histórica de una determinada observación desde $t_0, \ldots, t_k$ incluido éste, tendremos una serie de observaciones $O(t_0), \ldots, O(t_k)$ y el

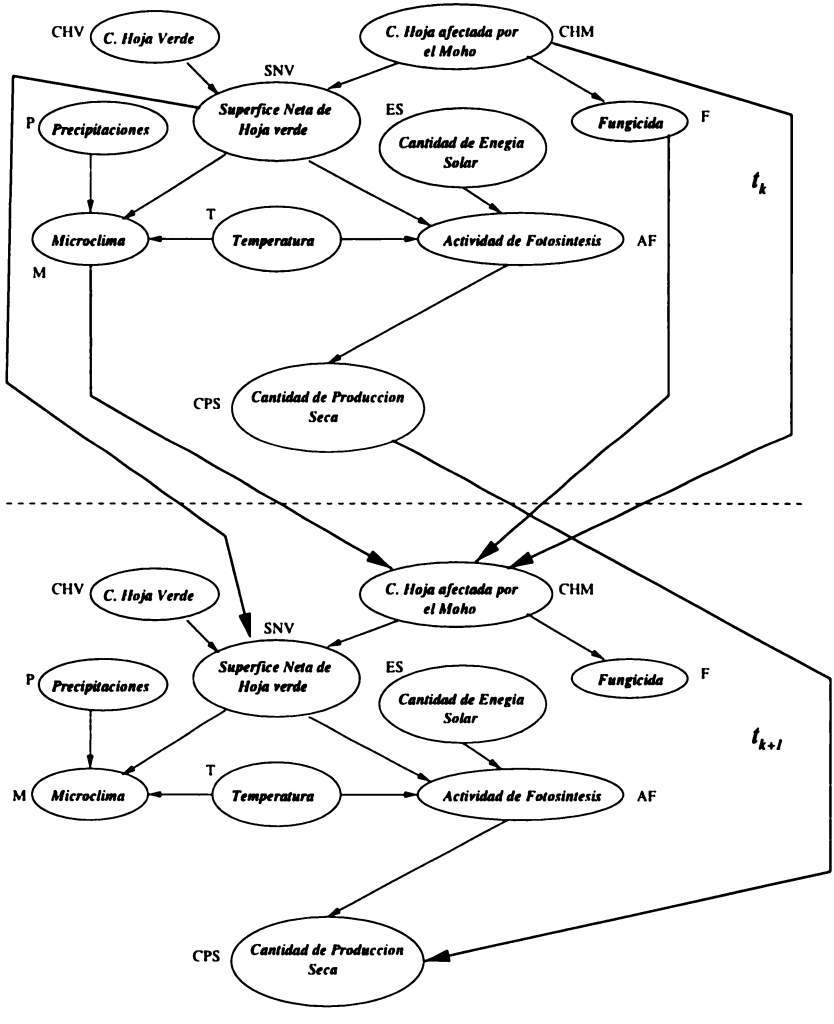


Figura 2. Red de creencia dinámica para un sistema de producción de trigo.

primer problema que se nos puede plantear solucionar será el de calcular la creencia del estado del sistema en este instante de tiempo t_k, en base a la evidencia acumulada hasta t_k. En términos de probabilidad será calcular la expresión:

$$P(X(t_k)|O(t_0),\ldots,O(t_k))$$

Calcular la expresión anterior de manera directa puede ser bastante complejo, así que podemos simplicar bastante su cálculo si consideramos que el problema es de tipo markoviano, esto es, la distribución del estado actual depende exclusivamente del estado anterior. En términos de probabilidad esto quiere decir que:

$$P(X(t_k)|X(t_0),\ldots,X(t_{k-1}),O(t_0),\ldots,O(t_k)) = P(X(t_k)|X(t_{k-1}),O(t_k))$$

A este tipo de modelos dinámicos markovianos se les denomina en la literatura estadística *Modelos Dinámicos Markovianos Parcialmente Observables* MDMPO, modelos que se caracterizan por tener un conjunto de observaciones en cada instante de tiempo. En estos modelos el conjunto de observaciones en t_k solo depende del estado actual del sistema, es decir de $X(t_k)$, en términos de probabilidad:

$$P(O(t_k)|X(t_0),\ldots,X(t_k),O(t_0),\ldots,O(t_{k-1})) = P(O(t_k)|X(t_k))$$

Fijémonos en que parece razonable el pensar que el conjunto de observaciones en t_k nos ayude a estimar el estado actual del proceso dinámico junto con el estado previo del sistema. Las expresiones que hemos visto nos llevan de una manera natural a definir una modelo gráfico que establezaca las relaciones comentadas.

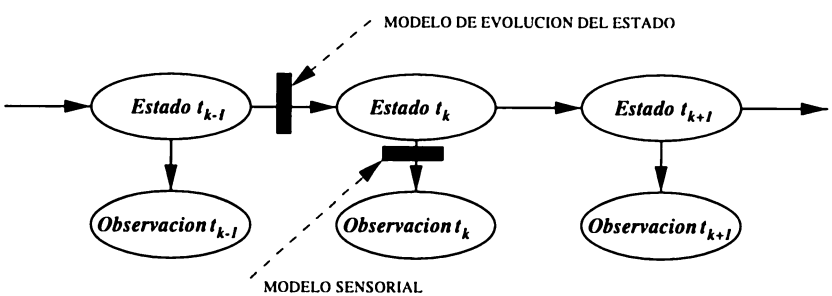


Figura 3. Modelo Gráfico para un MDMPO.

En la figura 3, el modelo de evolución del estado corresponde al modelo de transición entre estados del sistema, en términos de probabilidad se corresponde

con la distribución $P(X(t_k)|X(t_{k-1}))$ y el modelo sensorial se corresponderá a la distribución $P(O(t_k)|X(t_k))$.

El conjunto de expresiones que hemos visto hasta ahora nos permite simplificar de manera significativa el cálculo correspondiente a la estimación del estado actual del sistema $P(X(t_k))$. El cálculo se puede realizar en dos fases: (a) Fase de predicción y (b) Fase de estimación. Estas dos fases son una generalización de las técnicas bien conocidas en el análisis de series temporales con el nombre de *filtración de Kalman* (Kalman filters), estas técnicas se aplica universalmente en problemas de monitorización y control de todo tipo de sistemas dinámicos, desde plantas químicas hasta proyectiles dirigidos.

El cálculo de $P(X(t_k))$ se podrá realizar de la siguiente forma:

- Fase de Predicción: Primero, se predice la distribución de probabilidad en aquellos estados que habríamos esperado, con base al conocimiento que disponemos acerca del estado anterior:

$$P(X(t_k)) = \sum_{X(t_{k-1})} P(X(t_k)|X(t_{k-1}) = x(t_{k-1}))P(X(t_{k-1}) = x(t_{k-1}))$$

- Fase de Estimación: Tenemos ahora una distribución que se extiende a través de las variables de estado actuales, basada en todo menos en las observaciones recientes. La fase de estimación actualiza lo anterior a través de la observación en el instante t_k:

$$P(X(t_k)|O(t_k)) = \alpha P(O(t_k)|X(t_k))P(X(t_k))$$

y en donde α es una constante de normalización.

El trabajo presentado se estructura de la siguiente forma: En la siguiente sección estudiaremos de una manera más formal la definición de las redes de creencia dinámicas, identificaremos los problemas clásicos que se pretende resolver para posteriormente plantear metodologías generales para sus soluciones. Finalizaremos mediante el estudio de la aplicación del formalismo estudiado a los problemas de control y planificación.

2 Redes de creencia dinámicas

En este punto del trabajo vamos a estudiar y definir de una manera formal cómo se puede representar la evolución temporal del estado de un sistema dinámico mediante redes de creencia.

Como hemos descrito anteriormente la evolución temporal del estado de un sistema se representa en general mediante distribuciones temporales del tipo $P(X(t_k)|X(t_{k-1}))$ que describen de qué manera depende el estado actual del estado inmediantemente anterior. Vamos por tanto a suponer que se cumple la propiedad de Markov. Además consideraremos que estamos en el supuesto de condición estacionaria, es decir, que dichas distribuciones no cambian con el paso del tiempo, son las mismas para cada instante de tiempo t_k.

Por otra parte, para representar el estado en un sistema dinámico vamos a tener un conjunto de variables de estado X_i, además deberemos tener una serie de distribuciones de probabilidad que indiquen las dependencias entre las variables de estado en un mismo instante de tiempo, es decir, distribuciones del tipo $P(X_i|Pad(X_i))$. Estas distribuciones no dependen de la evolución temporal y son las mismas que describen la estructura de una red de creencia estática. Vamos a suponer que estas distribuciones se mantienen igual durante todo el proceso evolutivo del sistema. Por tanto este tipo de dependencias dan lugar a una red de creencia estática que definen las relaciones entre las variables del sistema dentro de un mismo intervalo de tiempo.

Si consideramos la evolución temporal del sistema junto con lo que hemos descrito en el párrafo anterior, nos dará como resultado la definición de una red de creencia dinámica. Así pues, en una red de creencia dinámica tendremos distribuciones de probabilidad en donde se especifiquen tanto las dependencias dentro del mismo intervalo de tiempo y las dependencias del intervalo de tiempo inmediatemente anterior, dando lugar a distribuciones de probabilidad del tipo: $P(X_i(t_k)|Pad_{t_k}(X_i(t_k)) \cup Pad_{t_{k-1}}(X_i(t_k)))$, siendo $Pad_{t_k}(X_i(t_k))$ el conjunto de padres de la variable $X_i(t_k)$ en el mismo intervalo de tiempo, y por tanto $Pad_{t_{k-1}}(X_i(t_k))$, el conjunto de padres en el intervalo de tiempo inmediantamente anterior.

Formalmente una Red de Creencia Dinámica (RCD) cubre un número de periodos de tiempo n. Sea $G = (V, E)$ un grafo dirigido acíclico que describe la estructura del modelo dinámico. Si t_0 es el primer modelo de la red, entonces V lo forman los subconjuntos disjuntos, $V(t_0), ..., V(t_{n-1})$. Es decir,

$$V = V(t_0, n) = \bigcup_{t=t_0}^{t_{n-1}} V(t)$$

Al conjunto de arcos dirigidos

$$E^{tmp}(t_k) = \{(v, u) \in E | v \in V(t_{k-1}), u \in V(t_k)\}, \quad t_0 \leq t_k \leq t_{n-1},$$

se le denomina *arcos temporales* o *relaciones temporales*, de un periodo de tiempo t y define cómo la distribución de las variables del periodo de tiempo t son dadas condicionalmente sobre la distribución de las variables del periodo de tiempo $t-1$.(Figura 4)

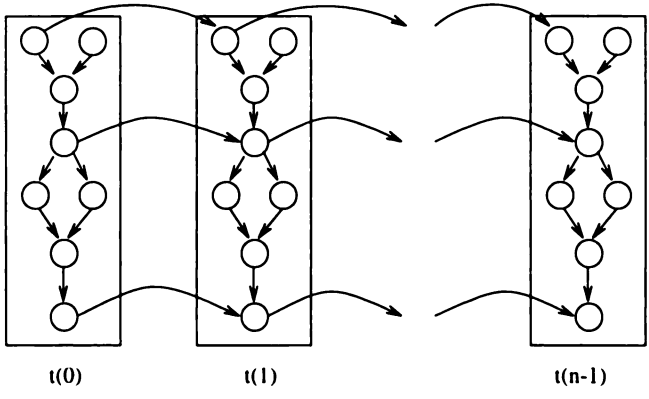


Figura 4. Red de Creencia Dinámica. Los arcos temporales aparecen como curvas.

El conjunto E de arcos de G puede describirse como sigue.

$$E = E(t_0, n) = E(t_0) \cup \bigcup_{t=t_0}^{t_{n-1}} E^*(t),$$

donde $E(t_k) \subseteq V(t_k) \otimes V(t_k)$ y $E^*(t_k) = E(t_k) \cup E^{tmp}(t_k)$.

Como hemos visto, asumiremos que las estructuras de los periodos de tiempo en una RCD cumplen la propiedad de Markov, es decir, el futuro es condicionalmente independiente del pasado dado el presente. Formalmente, y en términos de condiciones de independencia lo podemos notar como:

$$I(V(t_0), ..., V(t_{k-1}) | V(t_k) | V(t_{k+1}), ..., V(t_{n-1}))$$

para $k > 0$ y $n > 0$. Dando lugar a las distribuciones de probabilidad expresadas, es decir $P(X_i(t_k) | Pad_{t_k}(X_i(t_k)) \cup Pad_{t_{k-1}}(X_i(t_k)))$.

En definitiva, una RCD se puede considerar como un grafo dirigido acíclico, en donde el conjunto de variables se repiten en el tiempo y el conjunto de arcos está compuesto por los arcos temporales que interconectan periodos de tiempo

consecutivos y arcos no temporales que interconectan las variables en el mismo periodo de tiempo.

Bien, una vez representado un sistema dinámico mediante una estructura bien conocida como son las redes de creencia, podremos pensar que las tareas que involucran los procesos de razonamiento en este tipo de sistemas se podrán resolver mediante algoritmos de inferencia tratados en trabajos anteriores. Por ejemplo si estamos en el paso temporal actual t_k, entonces tendremos evidencias para diferentes nodos de nuestra red hasta el momento t_k, incluido éste, entonces será posible actualizar nuestras creencias para todos los nodos pertenecientes a nuestra red de creencia dinámica mediante algoritmos de propagación, bien exactos por ejemplo Hugin , o bien mediante algoritmos aproximados.

Si bien esto último es cierto, existe una razón fundamental para pensar detenidamente en realizar los procesos de actualización de nuestro conocimiento de una forma más razonable, ya que una red de creencia dinámica puede resultar extremadamente grande como para que los algoritmos de inferencia estudiados hasta ahora resulten ineficientes.

En primer lugar hemos de definir los problemas que clásicamente se nos plantea resolver en sistemas evolutivos para después estudiar las maneras más eficientes de tratarlos mediante las redes de creencia dinámicas.

Existen fundamentalmente dos problemas clásicos que se quieren resolver en cualquier sistema que evoluciona con el tiempo. El primero es determinar el estado actual del sistema dada la evidencia acumulada hasta el momento y el segundo será determinar el estado del sistema en un futuro dada la evidencia disponible hasta el instante en cuestión. El primer problema recibe el nombre de ***monitorización*** del sistema y el segundo se le denomina ***predicción*** del comportamiento del sistema.

Como hemos comentado, el primer problema que se pretende abordar en un sistema dinámico es el de la monitorización del sistema, esto es, si el paso de tiempo actual es t_k, determinar la creencia en t_k de las variables de estado del sistema teniendo en cuenta la evidencia observada hasta el instante t_k, incluido éste. En términos de distribuciones de probabilidad, estamos interesados en calcular:

$$P(X_i(t_k)|O(t_0),\ldots,O(t_k))$$

para $i = 1,\ldots,m$, siemdo m el número de variables de estado del sistema.

El problema de la predicción en un sistema dinámico pretende conocer el estado del sistema en instantes posteriores o futuros al periodo de tiempo actual o presente t_k. Hemos de notar que en este tipo de problemas parece lógico suponer que no tenemos ninguna evidencia u observación en los instantes de tiempo

que corresponden al futuro, es decir, para variables del sistema que pertenecen a periodos posteriores a t_k. En términos de probabilidad, al igual que en el caso anterior, estamos considerando calcular la siguiente expresión:

$$P(X_i(t_{k+j})|O(t_0),\ldots,O(t_k))$$

para $i = 1,\ldots,m$ y $j = 1,\ldots,(n-k-1)$, donde m, de nuevo, es el número de variables de estado del sistema y n es el número de periodos de tiempo total del sistema dinámico.

Si bien estos dos problemas anteriores son los que habitualmente se pretenden resolver, existen situaciones en las que podemos estar interesados en considerar las observaciones producidas en el pasado, presente y futuro, o lo que es lo mismo, actualizar todo el modelo a la luz de toda la evidencia disponible hasta un periodo de tiempo determinado. Hemos de notar que entonces t_k no se refiere tanto al paso actual de tiempo en el proceso, sino más bien un índice de referencia dentro de la red de creencia dinámica. A este problema se le suele conocer como ***suavizado*** del sistema en la teoría de control y se utiliza para eliminar incertidumbre en el sistema, ya que se tienen en cuenta toda la información disponible en el proceso evolutivo del mismo. En términos de probabilidad lo que se pretende calcular es la siguiente expresión:

$$P(X_i(t_k)|O(t_0),\ldots,O(t_j))$$

donde $k = 0,\ldots,j$.

Vamos por tanto a pasar a estudiar como se pueden resolver estos tres tipos de problemas que hemos descrito teniendo en cuenta las especificaciones del modelo de red de creencia dinámica que hemos definido.

2.1 Monitorización de un sistema dinámico

Como hemos estudiado en los apartados anteriores, un problema típico en el análisis de series temporales es el de monitorización, que como hemos descrito será calcular la siguiente expresión en términos de probabilidad:

$$P(X_i(t_k)|O(t_0),\ldots,O(t_k))$$

para $i = 1,\ldots,m$, siemdo m el número de variables de estado del sistema.

Ahora, fijándonos en la figura 3, observamos que en tales modelos existen dos tipos de nodos, (a) los nodos correspondientes a la descripción del modelo y (b) los nodos que corresponden a las observaciones en cada intervalo de tiempo,

por consiguiente, podremos establecer las siguientes sentencias de independencia condicional:

$$I(X_i(t_{k-1}), X_i(t_{k+1})|X_i(t_k)) \Longrightarrow$$
$$P(X_i(t_{k+1})|X_i(t_k), X_i(t_{k-1})) = P(X_i(t_{k+1})|X_i(t_k))$$

$$I(O(t_{k-1}), O(t_k)|X_i(t_k)) \Longrightarrow$$
$$P(O(t_k)|O(t_{k-1}), X_i(t_k)) = P(O(t_k)|X_i(t_k))$$

Partiendo de las expresiones anteriores se puede reescribir la expresión

$$P(X_i(t_k)|O(t_0), \ldots, O(t_k)) = \alpha P(O(t_k)|X_i(t_k)) \times$$
$$\sum_{X(t_{k-1})} P(X_i(t_k)|X_i(t_{k-1}))P(X_i(t_{k-1})|O(t_0), \ldots, O(t_{k-1}))$$

donde α es una constante de normalización.

Si denominamos a $\mathbf{F}_{t_k} = P(X_i(t_k)|O(t_0), \ldots, O(t_k))$, y volvemos a la expresión anterior, entonces tendremos:

$$\mathbf{F}_{t_k} = \alpha P(O(t_k)|X_i(t_k) \times \sum_{X(t_{k-1})} P(X_i(t_k)|X_i(t_{k-1}))\mathbf{F}_{t_{k-1}}$$

Por consiguiente, tendremos una expresión recursiva que depende del instante de tiempo anterior.Por tanto, partiremos de los instantes de tiempo iniciales y a partir de ellos iremos calculando la expresión anterior.

Mediante el estudio realizado se puede establecer un método general para realizar el proceso de monitorización de un sistema dinámico. El ciclo del proceso consta de los tres pasos siguientes:

1. Fase de Predicción: Partiremos de la red correspondiente a dos intervalos de tiempo consecutivos, es decir t_{k-1} y t_k. Hemos de notar que previamente tenemos calculado $P(X_i(t_{k-1}))$, en la que hemos incorporado toda la evidencia acumulada hasta el momento y que incluye $O(t_{k-1})$. También hemos de ver que la porción de red t_{k-1} no tiene relación con porciones anteriores en el tiempo. Las variables de estado de t_{k-1} sí tienen relación con probabilidades anteriores. Calcularemos ahora el vector de probabilidad $P(X_i(t_k))$, para lo que se puede realizar un proceso de actualización de creencia estándar de una red de creencia aplicado a la evidencia $O(t_{k-1})$.

2. Fase de Eliminación: Ahora eliminamos la porción de red que corresponde al instante de tiempo t_{k-1}. Para ello, hay que añadir la tabla de probabilidad anterior de las variables de estado que correspondan al instante de tiempo t_k. Esta probabilidad es justamente la que hemos calculado anteriormente, es decir: $P(X_i(t_k))$, dada la evidencia acumulada hasta el momento, incluido $O(t_{k-1})$.
3. Fase de Estimación: Añadimos ahora una nueva observación $O(t_k)$, aplicando la actualización estándar de una red de creencia para el cálculo de $P(X_i(t_k))$, que es la distribución de probabilidad en el estado actual. Luego añadiremos la porción correspondiente para t_{k+1}. La red quedaría lista para el siguiente ciclo.

2.2 Predicción de estados futuros de un sistema dinámico

En el an'alisis de series temporales una aplicación típica es la de realizar predicciones en los procesos estocásticos considerados, esto es, calcular estimaciones de las distribuciones de las variables futuras desde las observaciones del pasado y el presente. En términos de probabilidad tedremos que calcular la expresión:

$$P(X_i(t_{k+j})|O(t_0),\ldots,O(t_k))$$

para $i = 1,\ldots,m$ y $j = 1,\ldots,(n-k-1)$, donde m es el número de variables de estado del sistema y n es el número de periodos de tiempo total del sistema dinámico.

Dentro del modelo computacional presentado, la predicción es una extensión directa del proceso de monitorización. Para ello tomaremos la red después del paso 3 del método anterior, añadiremos porciones de la red de creencia dinámica, es decir $t_{k+1},\ldots,t_{k+j}$, y aplicaremos un algoritmo de inferencia en la red de creencia resultante para calcular las distribuciones de estados futuros dada la evidencia acumulada hasta el instante de tiempo actual.

De todas formas este método consume mucho tiempo en las operaciones realizadas ya que la red resultante puede resultar demasiado grande para que los algoritmos de inferencia sean eficientes. Asimismo también implica un número alto de operaciones innecesarias ya que el cálculo que se realiza entra en una especie de contradicción entre la exactitud deseada y la confiabilidad, en general, que puede ser alcanzada en los procesos de predicción, además hemos de tener en cuenta que habitualmente las predicciones se realizan en un pequeño número de variables de estado.

Si embargo si nos fijamos en la red que resulta para el cálculo de las distribuciones de predicción, nos daremos cuenta que posee una característica especial

y es que no se posee ningun tipo de evidencia para los instantes de tiempo que corresponden al futuro. Por consiguiente, se pueden proponer métodos alternativos utilizando métodos de Monte-Carlo de muestreo hacia adelante para realizar estimaciones en las variables deseadas. Esto es posible gracias a que las muestras generadas por dichos métodos serán todas congruentes con la evidencia por ser ésta nula. La complejidad de estos métodos por lo general está en función lineal del número de nodos en la red y del grado de precisión deseado.

2.3 Suavizado o propagación hacia atrás en un sistema dinámico

En algunos problemas podemos desear tener en cuenta evidencias "futuras", así como las presentes y las del pasado. Notemos que entonces un periodo de tiempo actual t_k más que índice para indicar el paso del tiempo actual del proceso dinámico es un índice para hacer referencia a intervalos de tiempo de la red de creencia en su conjunto.

Las observaciones que poseemos posteriores a t_k nos pueden ayudar a eliminar incertidumbre en el estado t_k del sistema. Por ejemplo, podemos deducir quién estaba dentro de una casa donde se ha cometido un asesinato a través de observaciones consecutivas sobre qué personas abandonaron la casa en instantes posteriores al asesinato además de las personas que estaban en el interior de la casa antes del asesinato.

A este proceso se le denomina suavizado en teoría de control y se utiliza para corregir trayectorias calculadas mediante el proceso de monitorización. Así como el proceso de monitorización se puede conocer como propagación hacia adelante, el proceso de suavizado se puede ver como una propagación hacia atrás. En términos de probabilidad, estamos interesados en conocer:

$$P(X_i(t_k)|O(t_0),\ldots,O(t_j)) \text{ donde } k \in [0..j]$$

El método directo para calcular esta expresión es realizar un proceso de monitorización en ambos sentidos, esto es, hacia adelante y hacia atrás y posteriormente combinar la información en cada paso de tiempo, esto lo podemos ver en las siguientes expresiones:

$$P(X_i(t_k)|O(t_0),\ldots,O(t_j)) = \alpha \times P(X_i(t_k)|O(t_0),\ldots,O(t_k)) \times \\ \times P(O(t_{k+1}),\ldots,O(t_j))|X(t_k))$$

donde tenemos que $P(X_i(t_k)|O(t_0),\ldots,O(t_k))$ es la expresión del cálculo en el proceso de monitorización, llamemos $\mathbf{B}_{t_k}$ al resto de la expresión, es decir:

$$\mathbf{B}_{t_k} = P(O(t_{k+1}), \ldots, O(t_j))|X(t_k))$$

Por tanto nos faltaría calcular la anterior expresión para tener resuelto el problema. Ahora bien

$$\mathbf{B}_{t_k} = \sum_{X(t_{k+1})} P(O(t_{k+1})|X(t_{k+1})) \times$$
$$\times P(X(t_{k+1})|X(t_k)) \times P(O(t_{k+2}), \ldots, O(t_j)|X(t_{k+1}))$$

donde $\mathbf{B}_{t_{j-1}} = P(O(t_j)|X(t_{j-1}))$. De nuevo vemos que tenemos una expresión de carácter recursivo, por lo que el método general para el proceso descrito ser'a muy parecido al de monitorización, ya que si antes se partía de los instantes de tiempo iniciales y a partir de ellos vamos progresando a medida que el tiempo lo hace, en este caso tendremos que partir del periodo de tiempo actual y a partir del mismo y del inmediatemente anterior ir progresando, aunque esta vez hacia atrás.

3 Aplicaciones a problemas de planificación y control

En primer lugar vamos a estudiar de una forma teórica y abstracta los componentes de un sistema dinámico desde el punto de vista de la toría del control, pero esta formulación vale igualmente para tener un enfoque al problema de planificación.

Empezaremos por dar una descripción general de un controlador y los objetivos que deben cumplir, posteriormente pasaremos a ver su posible representación y manejo aplicando el formalismo de las redes de creencia dinámicas.

Un controlador es una caja negra que toma señales de entrada y como salida ofrece algunas acciones. El entorno se puede ver como otra caja negra que toma como entrada las acciones del controlador y genera como salida la siguientes señales de entrada del controlador.(fig 5).

Un sistema dinámico se puede ver de la siguiente forma: Tendremos un conjunto de puntos o instantes de tiempo $\mathcal{T}$, tendremos un conjunto de estados posibles del sistema $\mathcal{X}$, tendremos otro conjunto para las señales de entrada para el controlador $\mathcal{Y}$ y por último tendremos otro conjunto de acciones posibles que puede tomar el controlador $\mathcal{U}$. Para definir el comportamiento del sistema y del controlador tendremos un conjunto de variables ordenadas en el tiempo: $X(t_k) \in \mathcal{X}$, $Y(t_k) \in \mathcal{Y}$ y $U(t_k) \in \mathcal{U}$, con $t_k \in \mathcal{T}$.

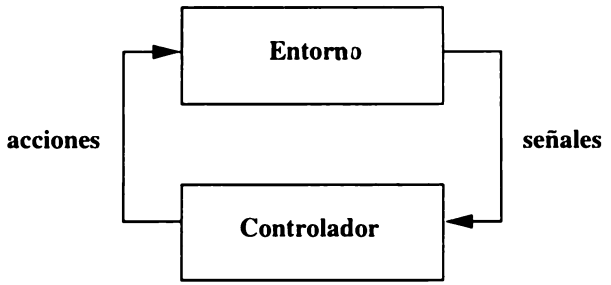


Figura 5. Componentes de un sistema dinámanico.

Los modos en que evolucionan con el tiempo estas variables reciben el nombre de historias, lineas de tiempo o en el caso del control, trayectorias. El conjunto de todas las posibles trayectorias del estado se definen como:

$$H_{\mathcal{X}} \triangleq \{h_{\mathcal{X}} : \mathcal{T} \longrightarrow \mathcal{X}\}$$

El conjunto de posibles entradas u observaciones para el controlador evoluciona en el tiempo conforme:

$$H_{\mathcal{Y}} \triangleq \{h_{\mathcal{Y}} : \mathcal{T} \longrightarrow \mathcal{Y}\}$$

Generalmente restringiremos el posible conjunto de estados del sistema imponiendo que debe evolucionar conforme a un conjunto de leyes, dichas leyes se suelen denominar ecuaciones del estado del sistema y tienen la siguiente forma:

$$X(t_{k+1}) = f(X(t_k), U(t_k))$$

También se restringe el conjunto de posibles entradas al controlador u observaciones para éste, de la misma forma que el estado del sistema:

$$Y(t_k) = g(X(t_k))$$

Hasta ahora hemos definido el comportamiento del entorno, pasaremos por tanto a definir el comportamiento del controlador de forma análoga a la anterior. Para el posible conjunto de salidas del controlador dependientes del tiempo, es decir, para la evolución temporal de las salidas tendremos:

$$H_{\mathcal{U}} \triangleq \{h_{\mathcal{U}} : \mathcal{T} \longrightarrow \mathcal{U}\}$$

conjunto de acciones tomadas por el controlador conforme el tiempo progresa. Restringiremos la evolución temporal de las entradas al controlador de la siguiente forma:

$$\mathcal{P} \triangleq \{\pi : H_{\mathcal{Y}} \longrightarrow \mathcal{U}\}$$

A este último tipo de funciones se le denomina leyes de control o políticas. En el caso más simple, la función de salida g, será la función identidad y solo el último estado del sistema será relevante para la decisión que ha de tomar el controlador en función de lo que observa. En este caso, el conjunto de políticas puede ser restringido a

$$\mathcal{P} = \{\pi : \mathcal{X} \longrightarrow \mathcal{U}\}$$

A continuación necesitamos alguna manera de espeficar lo que un controlador tiene que hacer. Empezaremos definiendo la tarea de un controlador como una relación entre pares entradas/salidas.

$$\mathcal{K} \subset \mathcal{Y} \otimes \mathcal{U} \quad \text{ó} \quad \mathcal{K} \subset \mathcal{X} \otimes \mathcal{U}$$

Lo normal, sin embargo, es definir la tarea que debe realizar el controlador en términos de mejor acción que puede realizar para un estado dado o para unas señales de entrada dadas. Definir una tarea es un método directo de especificar el comportamiento deseado de un controlador.

El primer método que podremos tener para especificar la tarea del controlador es mediante un objetivo en términos de estado preferido, sin especificar la manera de llegar a estos estados. Podremos definir un objetivo como $G \subset H_{\mathcal{X}}$. Partiendo del objetivo G la tarea es determinar una política $\pi \in \mathcal{P}$ que restringa el comportamiento del controlador para alcanzar a G, cuando ocurre esto último se suele decir que hemos encontrado una solución satisfactoria.

Otra aproximación es definir una función de valoración

$$V : H_{\mathcal{X}} \longrightarrow \mathbb{R}$$

que asocia una medida de deseabilidad a cada trayectoria del estado. En este caso, deseamos buscar una política $\pi \in \mathcal{P}$ que fuerce al estado del sistema dinámico a evolucionar de acuerdo a una trayectoria que es maximal con respecto a V y la solución se le denomina solución óptima. El problema de buscar una política que alcance un objetivo G o que maximiza una función se le denomina ***problema de control.***

Hemos de notar que si tenemos un modelo computacional para poder determinar el conjunto de acciones que debe tomar el controlador en cada instante de tiempo, estaremos especificando la forma de realizar un plan, por lo tanto todo lo que hemos visto y veremos a continuación servirá también para resolver el problema de la planificación de tareas en un entorno dinámico.

Como en la mayoría de los problemas complejos descompondremos nuestro problema es subproblemas más pequeños. En lo que al problema del control se refiere, éste se descompondrá habitualmente en dos subtareas: (a) problema de estimación del estado del sistema, a la luz de las observaciones actuales y (b) problema de la regulación de las entradas, es decir, tomar una decisión sobre qué acción o acciones tomar por parte del controlador.

El problema de la estimación se puede ver en el caso más simple como una función:

$$E \triangleq \{e : \mathcal{Y} \longrightarrow \mathcal{X}\}$$
$$e(Y(t_k)) = \hat{X}(t_k)$$

De forma similar, el problema de regulación se puede ver como un función:

$$R \triangleq \{r : \mathcal{X} \longrightarrow \mathcal{U}\}$$
$$r(X(t_k)) = U(t_k)$$

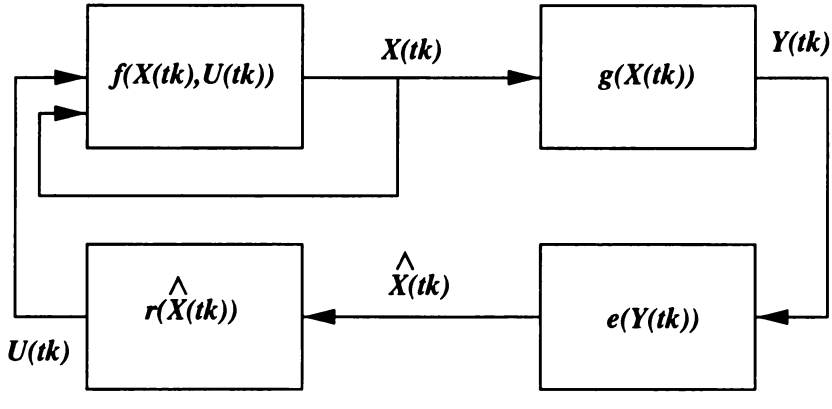


Figura 6. Un sistema dinámico.

En la figura 6 podemos observar un diagrama de bloques con los distintos componentes comentados para una controlador de un sistema dinámico.

Si nos fijamos detenidamente en el diagrama de bloques de la figura 6, podremos extraer las siguientes conclusiones:

- Tendremos que hacer una estimación del estado actual del sistema a partir de las observaciones de que disponemos en cada instante de tiempo; esto es precisamente lo que se hacía en la definición del modelo sensorial de un modelo dinámico markoviano parcialmente observado.
- Por otra parte tendremos un modelo para decidir en cada intervalo de tiempo qué acción o acciones tomar a partir de la estimación efectuada del estado del sistema.
- El estado del sistema en el instante de tiempo siguiente viene dado en función de la acción o accciones tomadas en un instante dado y del estado del sistema en dicho instante de tiempo.
- Y por último, hemos de establecer una función de utilidad o valoración para conseguir restringir el comportamiento del controlador. Podemos suponer que esta función es separable en el tiempo, esto quiere decir que tenemos una función de utilidad para cada periodo de tiempo y que se puede obtener una función de utilidad total en función de las anteriores.

Partiendo de estas conclusiones, la forma de representar un sistema dinámico mediante redes de creencia dinámicas es inmediata. Su posible representación aparece en la figura 7. En este tipo de redes, al ser redes enfocadas a la toma de desiciones, aparecen varios tipos de nodos: nodos de utilidad, nodos de decisión, nodos de observación y nodos de estado.

3.1 Aplicación: localización y seguimiento de un robot móvil

En este punto ilustraremos el uso de las redes de creencia dinámicas en el problema de planificación, problema este que requiere razonamiento temporal bajo incertidumbre. Utilizaremos para ello un problema concreto bien conocido en la literatura relacionada con el control y la planificación como es el de la navegación de un robot móvil. La aplicación involucra un robot móvil que navega y sigue agentes en movimiento en un entorno limitado. El robot esta equipado con un radar y un sistema de visión un poco rudimentario y los agentes en movimiento pueden ser personas u otros robots móviles.

La tarea del robot consistirá en detectar y seguir objetos en movimiento, registrando sus localizaciones en un sistema de coordenadas del entorno limitado en donde navega. El robot conoce bien el entorno donde permanece y debe evitar los

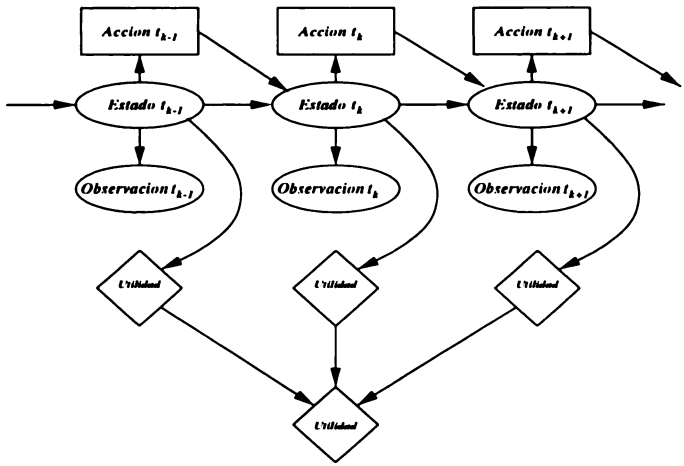


Figura 7. Un sistema dinámico modelizado mediante una RCD.

obstáculos que se le presenten en el camino. Por otra parte lo normal es que exista un error en los movimientos del robot y por tanto debe de estimar continuamente su localización dentro del entorno para no perderse.

Para modelar el sistema vamos a considerar los siguientes elementos: Poseemos un conjunto de posibles localizaciones dentro del entorno donde se mueve el robot y el agente en movimiento, lo denominaremos $\mathcal{L}$. Posteriormente se definen un conjunto de variables para estructurar un modelo de decisión. Sean S_A y S_R dos variables de estado que representan las localizaciones del agente en movimiento y del robot respectivamente. Ambas variables toman sus valores en el conjunto $\mathcal{L}$.

Sea M una variable de decisión cuyos valores serán las posibles acciones de movimiento del robot. Tendremos otra variable de decisión A cuyos posibles valores son la mejor estimación de la actual localización del agente en movimiento. Asi, $\Omega_A = \Omega_{S_T}$ y la decisión del robot se convierte en un problema de estimación de la mejor acción que debe realizar.

En cada intervalo de tiempo, el robot puede disponer de observaciones, tanto de su propia localización dentro del entorno como de la localización del agente en movimiento con respecto al propio robot. Sean O_R y O_A, las variables que representan lo anterior. La figura 8 representa nuestro problema a través de una red de creencia dinámica, esta red tiene nodos de decisión y nodos de evaluación, por tanto dicha red está enfocada a las decisiones.

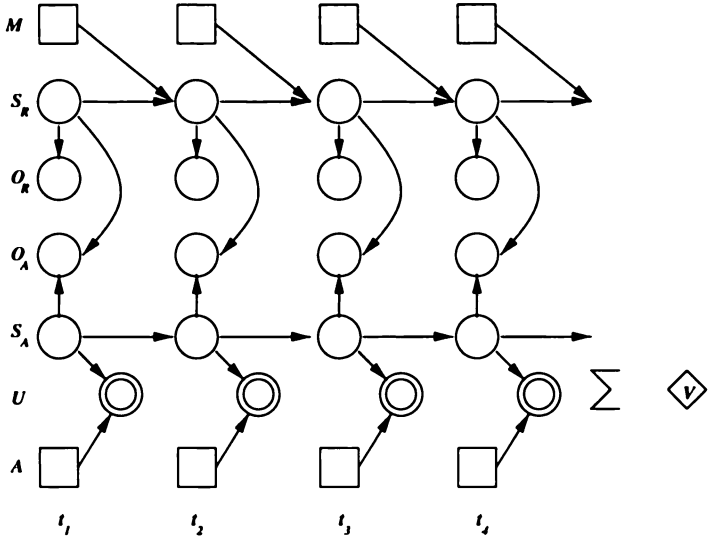


Figura 8. Modelo de decisión para el problema del robot móvil.

Además necesitamos especificar una función de utilidad que represente nuestras preferencias sobre los posibles resultados de las acciones. El valor de un movimiento se determina por cómo de bien posicione al robot para realizar observaciones que se esperan mejoren su estimación de la localización del agente en movimiento. Mediremos la calidad de una estimación $\hat{\delta}$, mediante una distancia euclídea, $||\hat{\delta}, \delta||$, con respecto a la localización actual del agente δ. El modelo de decisión incluye un nodo determinístico U que representa el error estimado en cada intervalo de tiempo.

Si suponemos que la función de utilidad es separable en el tiempo, la función total de utilidad vendrá dada por la suma de las funciones de utilidad en cada intervalo de tiempo, es decir:

$$U(t_k) = -\gamma(t_k)||A(t_k), S_T(t_k)||$$

donde $\gamma : \mathcal{T} \longrightarrow [0, 1]$ es una función decreciente sobre el tiempo que se usa para descontar el impacto de futuras consecuencias.

La función de valoración total será la suma de las funciones de utilidad para cada instante de tiempo t_k:

$$V(A(t_1), \ldots, A(t_n), S_T(t_1), \ldots, S_T(t_n)) = \sum_{k=1}^{n} U(t_k)$$

La estimación de las acciones pueden ser determinadas directamente por las observaciones. Dada la evidencia actual $O(t_k)$, la distancia ponderada de una estimación de $A(t_k)$ es:

$$E[||A(t_k), S_A(t_k)|| \mid O(t_k)] = \sum_{S_A(t_k)} ||A(t_k), S_A(t_k)|| \times P(S_A(t_k)|O(t_k))$$

donde $O(t_k) = \bigcup_{j=1}^{k} \{O_R(t_j), O_A(t_j), M(t_{j-1})\}$

Dado que la función de utilidad es separable en el tiempo, el valor óptimo $a^*(t_k)$ de $A(t_k)$ se puede caracterizar como una función de la evidencia $O(t_k)$

$$a^*(t_k)(O(t_k)) = \arg \max_{A(t_k)} E[||A(t_K), S_A(t_k)|| \mid O(t_k)]$$

Para nuestro problema de planificación, M son las acciones que puede llevar a cabo el robot como por ejemplo, desplazarse hasta el final del pasillo. Como entrada a nuestro problema dispondremos de un conjunto secuencial de acciones del robot que tomaremos como evidencia en cada intervalo de tiempo, y como resultado nuestra red nos ofrecerá la bondad del plan suministrado al robot. Hemos de notar que los movimientos del robot influyen indirectamente en los valores que podremos observar de la localización del agente en movimiento que queremos seguir. Para evaluar nuestra función de valoración tendremos que seguir los siguiente pasos:

1. Instanciar el valor de M correspondiente al paso de tiempo t_k, que junto con las variables de observación $O_R(t_k)$ y $O_A(t_k)$ será nuestra evidencia en esta etapa t_k.
2. Calcular las distribuciones de probabilidad de predicción para las variables de observación O_R y O_A, además de S_A para los tiempos futuros $t_{k+1}, \ldots, t_n$.
3. Usar las distribuciones de probabilidad anteriores para calcular $a^*(t_i)(O(t_i))$ para $k < i \leq n$ de acuerdo con la expresión anterior.
4. Calcular el estimador óptimo $V = \sum_i U(t_i)$.

Por tanto, de esta forma podremos evaluar planes para nuestro robot y quedarnos con el que optimice nuestra función de valoración.

4 Conclusiones

Hemos estudiado la forma de representar un sistema que evoluciona con el tiempo mediante una extensión de las redes de creencia, dando lugar a lo que se conoce como redes de creencia dinámicas. Una vez establecida la representación,

hemos descrito los problemas que clásicamente se han de resolver en un sistema dinámico. Existen algoritmos eficientes para realizar procesos de inferencia o razonamiento en redes de creencia, pero estos algoritmos no son aplicables directamente en las redes de creencia dinámicas ya que esta representación suele ser demasiado grande para que los algoritmos de inferencia, ya sean exactos o aproximados, se vean claramente desbordados, por lo que hemos estudiado la forma de adaptar estos algoritmos para resolver los problemas clásicos de los sistemas dinámicos.

Por último, hemos estudiado la manera de adaptar las redes de creencia dinámicas para resolver los problemas de control y planificaci'on en entornos con incertidumbre. Para ello hemos utilizado un problema concreto como ilustración de esto último.

Referencias

1. C. Berzuini, R. Bellazi y S. Quaglini. Temporal reasoning with probabilities. *Proceeding of the V Workshop on Uncertainty in Artificial Intelligence* ,pp. 14–21, 1989.
2. J. Binder, K. Murphy y S. Russell. Space-efficient inference in dynamic probabilistic networks. *Proceeding of the 15th International Conference on Uncertainty in Artificial Intelligence* ,Nagoya, Japón, 1997.
3. P. Dagum y A. Galper. Forecasting sleep apnea with dynamic network models. *Proceeding of the Ninth Conference on Uncertainty in Artificial Intelligence* ,pp. 64–71, 1993.
4. P. Dagum, A. Galper and E. Horvitz. Dynamic network models for forecasting. *Proceeding of the Eighth Conference on Uncertainty in Artificial Intelligence* ,pp. 41–48, 1992.
5. T. Dean y K. Kanazawa. A model for reasoning about persistence an causation. *Computational Intelligence*,5 ,pp. 41–48, 1992.
6. T. L. Dean y M. P. Wellman. *Planning and Control.* Morgan Kaufmann Publishers, San Mateo, California , 1991.
7. K. Kanazawa. *Reasoning about Time and Probability*, Tesis Doctoral, University of Brown, 1992.
8. K. Kanazawa. A logic and time nets for probabilistic inference. *Proceedings of the Tenth National Conference on Uncertainty in Artificial Intelligence* ,pp. 360–365, 1991.
9. K. Kanazawa, D. Koller, S. Russell. Stochastic simulation algorithms for dynamic probabilistic networks. *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence* ,pp. 346–351, 1995.
10. U. Kjærulff. A computational scheme for dynamic bayesian networks. *Research Report R-93-2018*, Departamento de Matemáticas e Informática, Universidad de Aalborg, Dinamarca, 1993.
11. D. Koller. Approximate probabilistic inference in dynamic processes. *Working Notes of the 1996 AAAI Spring Symposium on Learning Dynamical Systems* , 1996.
12. A. Lekuoma. *Modelización gráfica de sistemas dinámicos markovianos parcialmente observados*, Tesis Doctoral, Departamento de Métodos Estadísticos,Universidad de Zaragoza, 1996.
13. A. Nicholson y M. Brady. Sensor validation using dynamic belief networks. *Proceedings of the Eighth Conference on Uncertainty in Artificial Intelligence*,pp. 207–214. 1992.
14. J. Pearl. *Probabilistic Reasoning in Intelligence Systems: Networks of Plausible Inference.* Morgan Kaufmann Publishers, San Mateo, California , 1988.
15. S. Russell y P. Norvig. *Inteligencia Artificial. Un enfoque moderno.* Prentice Hall Hispanoamericana, 1996.

Modelos Gráficos para Probabilidades Imprecisas

Serafín Moral

Dpto. de Ciencias de la Computación e I.A.
Universidad de Granada
18.071 - Granada
correo-e: smc@decsai.ugr.es

Resumen

En este artículo se introduce el problema del cálculo con probabilidades imprecisas. Se distinguen dos casos bien diferenciados, el cálculo con restricciones y la propogación de probabilidades imprecisas bajo relaciones de independencia. Para el primero, se estudiarán métodos basados en programación lineal y de propagación de restricciones. Los algoritmos de propagación se aplicarán a ambos problemas, a través de su generalización a los sistemas basados en valuaciones.

1 Introducción

Uno de los inconvenientes más importantes de las redes Bayesianas es que necesitan una distribución de probabilidad para cada variable condicionada a sus padres. En muchas ocasiones, no se dispone de todos los valores necesarios para especificar una única distribución de probabilidad, o éstos se conocen sólo de forma parcial.

El razonamiento con probabilidades imprecisas se ha considerado en la literatura desde hace más de una centuria (ver Hailperin [17] para una revisión histórica del tema). Sin embargo, los enfoques empleados para plantear y resolver este problema han sido muy diversos. Muchas veces, bajo nombres distintos se consideran los mismos problemas; y en otras ocasiones, se usa la misma denominación para problemas totalmente distintos.

Moral [30] ha tratado de clasificar y sistematizar las distintas aproximaciones al uso de las probabilidades imprecisas en Inteligencia Artificial. La distinción fundamental se basa en la consideración de relaciones de independencia entre las variables. En el caso de que no se consideren, diremos que tenemos un problema de cálculo con *restricciones probabilísticas*, aunque también se han usado otros nombres como el de *Lógica Probabilística* [32–34] o el de *consistencia probabilística* [19]. Bajo relaciones de indepencia, se trata de generalizar los modelos de propagación de redes Bayesianas para probabilidades [3,5,8,16,12].

Otro aspecto importante relativo a las probabilidades imprecisas es el modelo matemático que se use para representarlas. El caso más general es el de conjuntos

convexos de probabilidades [50–52,42,7]. Sin embargo, existe un modelo más intuitivo como es el uso de intervalos de probabilidad, que se ha usado ampliamente en la literatura [1,13,16,44,43].

Shafer y Shenoy [41,40] han generalizado los algoritmos de propagación probabilísticos expresándolos en términos de valuaciones abstractas que verifican una serie de axiomas. Este esquema es la base que se ha empleado para obtener los algoritmos de propagación en otros modelos para representar la incertidumbre, como la Teor'ia de la Evidencia [26], la Teoría de la Posibilidad, o el caso que nos ocupa: las probabilidades imprecisas [8].

Los problemas con probabilidades parcialmente conocidas tienen, en general, una complejidad mayor que los problemas probabilísticos clásicos. Usualmente, se emplean algoritmos aproximados, muchos de ellos basados en técnicas de optimización combinatoria. Este trabajo no pretende ser un catálogo exhaustivo de todos los que han sido utilizados. Más bien trata de plantear de forma clara y sencilla cada uno de los problemas, difirenciándolo de los demás y haciendo referencia a las técnicas más importantes empleadas en su resolución.

La estructura de este trabajo es cómo sigue: la sección 2 introduce los fundamentos del cálculo con probabilidades imprecisas. La sección 3 considera la estructura axiomática de Shafer y Shenoy, presentando los algoritmos de forma abstracta. La sección 4, estudia el problema del cálculo con restricciones probabilísticas. Se consideran dos métodos alternativos para su solución: la programación lineal con la técnica de generación de columnas [18–20] y los algoritmos de propagación [30,47]. La sección 5 estudia los algoritmos de propagación bajo condiciones de independencia. Se describe la transformación de este problema en un algoritmo de optimización combinatoria y se indican algunas de las técnicas usadas en su resolución. Por último la sección 6 se dedica a las conclusiones.

2 Probabilidades Imprecisas

Supongamos una variable X que toma sus valores en un conjunto U. Existen algunos autores [38,28] que opinan que si tenemos incertidumbre sobre el valor de esta variable en un momento dado, entonces nuestro conocimiento se puede siempre representar mediante una única distribución de probabilidad. Sin embargo, existen ocasiones en las que disponemos de muy poca información sobre X, y determinar una única distribución de probabilidad puede traducirse en un ejercicio de adivinar unos valores a partir de nada. Surge en estas situaciones de forma natural el uso de probabilidades imprecisas. Desde una interpretación objetiva de la probabilidad, éstas representarían los distintos posibles valores de la frecuencia de un suceso. Desde un punto subjetivo [50] las probabilidades imprecisas reflejan un comportamiento muy cauteloso a la hora de tomar decisiones.

Quizás el modelo más natural para las probabilidades imprecisas sean los intervalos de probabilidad [50,13,48]. En este caso, en lugar de asignar un único valor de probabilidad a cada suceso, se le asigna un intervalo de valores. Así un sistema de intervalos para la variable X será un par $(\underline{P}, \overline{P})$ de funciones

$$\underline{P}, \overline{P} : 2^U \rightarrow [0, 1] \tag{1}$$

donde $\underline{P}(A) \leq \overline{P}(A), \forall A \subseteq U$.

En general, esta definición no implica que los intervalos de probabilidad sean imposibles de utilizar en la práctica debido a que necesitan un conjunto exponencial de valores en función del tamaño de U. Inicialmente, nuestra información se concentrará en algunos de los subconjuntos de U a los que les asignamos sus intervalos correspondientes. Para el resto de los sucesos, el intervalo asociado será el $[0, 1]$ que no será necesario representarlo de forma explícita.

Un sistema de intervalos siempre define un conjunto de posibles distribuciones de probabilidad en U. El conjunto de distribuciones de probabilidad asociado a $(\underline{P}, \overline{P})$ viene dado por la siguiente expresión:

$$H = \{p : p \in \mathcal{P}, \underline{P}(A) \leq P(A) \leq \overline{P}(A)\} \tag{2}$$

donde $\mathcal{P}$ es el conjunto de todas las distribuciones de probabilidad posibles en U y P es la medida de probabilidad asociada a la distribución p.

Este conjunto de probabilidades H es convexo. Es decir, si $p_1, p_2 \in H$, y $\alpha \in [0, 1]$, entonces se verifica que $\alpha p_1 + (1 - \alpha) p_2 \in H$.

Ejemplo 1. Supongamos una urna que contiene bolas de cuatro colores: blancas (B), rojas (R), negras (N) y verdes (V). La urna contiene 10 bolas, de las que conocemos lo siguiente: 2 son blancas, 3 son rojas o negras, 3 son verdes, blancas o rojas, y 3 son verdes, blancas o negras.

Esta información se puede representar mediante los siguientes intervalos de probabilidad:

$$\begin{array}{llll}
\underline{P}(\emptyset) = 0 & \overline{P}(\emptyset) = 0 & \underline{P}(\{B\}) = 0.2 & \overline{P}(\{B\}) = 0.7 \\
\underline{P}(\{R\}) = 0 & \overline{P}(\{R\}) = 0.5 & \underline{P}(\{N\}) = 0 & \overline{P}(\{N\}) = 0.6 \\
\underline{P}(\{V\}) = 0 & \overline{P}(\{V\}) = 0.6 & \underline{P}(\{B, R\}) = 0.2 & \overline{P}(\{B, R\}) = 1 \\
\underline{P}(\{B, N\}) = 0.2 & \overline{P}(\{B, N\}) = 1 & \underline{P}(\{B, V\}) = 0.2 & \overline{P}(\{B, V\}) = 0.7 \\
\underline{P}(\{R, N\}) = 0.3 & \overline{P}(\{R, N\}) = 0.8 & \underline{P}(\{R, V\}) = 0 & \overline{P}(\{R, V\}) = 0.8 \\
\underline{P}(\{N, V\}) = 0 & \overline{P}(\{N, V\}) = 0.8 & \underline{P}(\{B, R, N\}) = 0.5 & \overline{P}(\{B, R, N\}) = 1 \\
\underline{P}(\{B, R, V\}) = 0.2 & \overline{P}(\{B, R, V\}) = 1 & \underline{P}(\{B, N, V\}) = 0.2 & \overline{P}(\{B, N, V\}) = 1 \\
\underline{P}(\{R, N, V\}) = 0.3 & \overline{P}(\{R, N, V\}) = 0.8 & \underline{P}(\{B, R, N, V\}) = 1 & \overline{P}(\{B, R, N, V\}) = 1
\end{array} \tag{3}$$

El conjunto de probabilidades asociado contiene, entre otras a las siguientes distribuciones de probabilidad en $\{B, R, N, V\}$

$$\begin{array}{llll} p_1(B) = 0.7, & p_1(R) = 0.3, & p_1(N) = 0, & p_1(V) = 0 \\ p_2(B) = 0.2, & p_2(R) = 0.6, & p_2(N) = 0.3, & p_2(V) = 0 \\ p_3(B) = 0.2, & p_3(R) = 0, & p_3(N) = 0.3, & p_3(V) = 0.5 \end{array} \tag{4}$$

Existen muchas más. De hecho el conjunto es infinito. Sin embargo, al ser U finito, existe un procedimiento para representar el conjunto H: mediante sus puntos extremos. Las distribuciones extremas son aquellas que no se pueden expresar mediante combinación convexa de otras dos distribuciones distintas de H, o de forma menos precisa, pero quizás más ilustrativa: las esquinas del convexo.

Todas las distribuciones anteriores son extremas, pero hay algunas más. Existen algoritmos para calcular las distribuciones extremas a partir de los intervalos de probabilidad. Una revisión de los mismos se puede consultar en [48].

□

Un sistema de intervalos $(\underline{P}, \overline{P})$ se dice que es propio (o envolvente de probabilidad) si y solo si existe un conjunto convexo H tal que

$$\underline{P}(A) = \text{Inf}\ \{P(A) : p \in H\}, \qquad \overline{P}(A) = \text{Sup}\ \{P(A) : P \in H\}, \tag{5}$$

Dados unos intervalos cualesquiera, $(\underline{P}, \overline{P})$ si su conjunto de probabilidades asociado H a través de la expresión (2) es distinto del vacío, podemos transformar $(\underline{P}, \overline{P})$ en una envolvente de probabilidad $(\underline{P}', \overline{P}')$ por medio de la expresión:

$$\underline{P}'(A) = \text{Inf}\ \{P(A) : P \in H\}, \qquad \overline{P}'(A) = \text{Sup}\ \{P(A) : P \in H\}, \tag{6}$$

El nuevo sistema de intervalos es siempre una envolvente de probabilidad y se verifica que $\underline{P} \leq \underline{P}'$ y $\overline{P} \geq \overline{P}'$. Es decir, para todo $A \subseteq U$ se tiene que $[\underline{P}(A), \overline{P}(A)] \subseteq [\underline{P}'(A), \overline{P}'(A)]$. Desde este punto de vista, podemos decir que, para cada suceso, los nuevos intervalos son más informativos que los originales, ya que los intervalos son más precisos.

Si la información de que disponemos originalmente es un sistema de intervalos $(\underline{P}, \overline{P})$, es conveniente transformar este sistema de intervalos en la envolvente de probabilidad asociada $(\underline{P}', \overline{P}')$ ya que no se aãde ninguna información que no esté contenida en el sistema original de intervalos. Lo que se hace es optimizar el intervalo de cada suceso, de acuerdo con los intervalos del resto de los sucesos.

Ejemplo 2. Supongamos un partido de fútbol entre nuestro equipo y un equipo visitante. Los posibles resultados son: ganamos (G), perdemos (P) o empatamos (E). Supongamos que inicialmente disponemos de los siguientes intervalos:

$$\begin{array}{llll} \underline{P}(\emptyset) = 0 & \overline{P}(\emptyset) = 0 & \underline{P}(\{G\}) = 0.3 & \overline{P}(\{G\}) = 0.7 \\ \underline{P}(\{P\}) = 0.1 & \overline{P}(\{P\}) = 0.8 & \underline{P}(\{E\}) = 0.1 & \overline{P}(\{E\}) = 0.7 \\ \underline{P}(\{G,P\}) = 0.3 & \overline{P}(\{G,P\}) = 0.9 & \underline{P}(\{G,E\}) = 0.5 & \overline{P}(\{G,E\}) = 1 \\ \underline{P}(\{P,E\}) = 0.3 & \overline{P}(\{P,E\}) = 0.8 & \underline{P}(\{G,P,E\}) = 1 & \overline{P}(\{G,P,E\}) = 1 \end{array} \tag{7}$$

Estos intervalos definen un conjunto convexo con los siguientes puntos extremos (ver [13] para un procedimiento eficiente para calcular estas distribuciones):

$$\begin{array}{lll} p_1(G) = 0.7, & p_1(P) = 0.2, & p_1(E) = 0.1 \\ p_2(G) = 0.7, & p_2(P) = 0.1, & p_2(E) = 0.2 \\ p_3(G) = 0.4, & p_3(P) = 0.5, & p_3(E) = 0.1 \\ p_4(G) = 0.3, & p_4(P) = 0.5, & p_4(E) = 0.2 \\ p_5(G) = 0.3, & p_5(P) = 0.1, & p_5(E) = 0.6 \end{array} \tag{8}$$

Desde estos puntos extremos podemos calcular los intervalos propios (tomando el ínfimo y el supremo sobre el valor asignado por cada probabilidad extrema):

$$\begin{array}{llll} \underline{P}'(\emptyset) = 0 & \overline{P}'(\emptyset) = 0 & \underline{P}'(\{G\}) = 0.3 & \overline{P}'(\{G\}) = 0.7 \\ \underline{P}'(\{P\}) = 0.1 & \overline{P}'(\{P\}) = 0.5 & \underline{P}'(\{E\}) = 0.1 & \overline{P}'(\{E\}) = 0.6 \\ \underline{P}'(\{G,P\}) = 0.4 & \overline{P}'(\{G,P\}) = 0.9 & \underline{P}'(\{G,E\}) = 0.5 & \overline{P}'(\{G,E\}) = 0.9 \\ \underline{P}'(\{P,E\}) = 0.3 & \overline{P}'(\{P,E\}) = 0.7 & \underline{P}'(\{G,P,E\}) = 1 & \overline{P}'(\{G,P,E\}) = 1 \end{array} \tag{9}$$

□

En muchas ocasiones, los intervalos solo se dan sobre los sucesos elementales, esto es los elementos de U. En ese caso, se puede realizar un tratamiento mucho más eficiente de la información ya que el número máximo de intervalos se reduce de las partes de U a los elementos de U [13]. Un sistema de intervalos elementales será un par de aplicaciones $(\underline{p}, \overline{p})$:

$$\underline{p}, \overline{p} : U \to [0,1] \tag{10}$$

verificando que $\underline{p}(u) \leq \overline{p}(u), \forall u \in U$.

Analogamente se define el conjunto de distribuciones de probabilidad asociado a un sistema de intervalos elementales:

$$L = \{p \in \mathcal{P} : \underline{p}(u) \leq p(u) \leq \overline{p}(u), \forall u \in U\} \tag{11}$$

A partir de L se puede calcular un sistema propio de intervalos tomando supremo e ínfimo:

$$\underline{p}'(u) = \text{Inf } \{p(u) : p \in L\}, \qquad \overline{p}'(u) = \text{Sup } \{p(u) : p \in L\} \tag{12}$$

Un sistema de intervalos $(\underline{P}, \overline{P})$ con conjunto de probabilidades H y un sistema de intervalos elementales $(\underline{p}, \overline{p})$ con un conjunto de probabilidades L se dicen equivalentes si y solo si $L = H$.

Ejemplo 3. Se puede comprobar que en el ejemplo de las urnas, no hay un sistema de intervalos elementales que defina el mismo conjunto de distribuciones posibles.

En el ejemplo del partido de fútbol, el sistema de intervalos es equivalente al sistema de intervalos elementales siguiente:

$$\begin{array}{llll} \underline{p}'(\{G\}) = 0.3 & \overline{p}'(\{G\}) = 0.7 & \underline{p}'(\{P\}) = 0.1 & \overline{p}'(\{P\}) = 0.5 \\ \underline{p}'(\{E\}) = 0.1 & \overline{p}'(\{E\}) = 0.6 & & \end{array} \tag{13}$$

□

Los conjuntos convexos de probabilidades se pueden representar de forma gráfica en conjuntos de tamaño tres en un triángulo equilátero de altura 1. Si hacemos corresponder cada lado del triángulo con uno de los elementos de U, una distribución de probabilidad, p, en U se representa por el punto del triángulo tal que para cada lado u_i la altura desde el punto sobre el lado tiene longitud $p(u_i)$.

Ejemplo 4. En el caso de los intervalos del ejemplo del fútbol, el conjunto convexo asociado es el de la figura 1.

□

Una forma alternativa de representar las probabilidades imprecisas es, por medio, de un convexo de distribuciones de probabilidad directamente. Este procedimiento es más general que el uso de los intervalos de probabilidad. Ya vimos que a cada sistema de intervalos se le puede asociar un conjunto convexo de probabilidades. Sin embargo, el recíproco no es cierto. Existen conjuntos convexos que no se pueden definir a partir de un sistema de intervalos. Dado un conjunto convexo H, siempre se puede definir un sistema de intervalos de probabilidad propio a partir de las ecuaciones (6). Sin embargo, si volvemos a calcular el conjunto de probabilidades, H', asociado a este sistema de intervalos, no siempre se tiene la igualdad $H = H'$. En general, lo que se verifica es que $H \subseteq H'$.

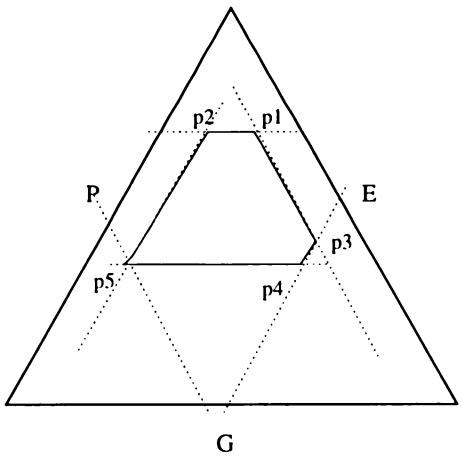


Figura 1. Conjunto convexo de probabilidades

Ejemplo 5. En el ejemplo del partido de fútbol, supongamos que sabemos que $p(G) \geq p(P) \geq p(E)$. Esto da lugar al conjunto convexo de la figura 2. Las distribuciones extremas de este convexo son:

$$\begin{array}{lll} p_1(G) = 1, & p_1(P) = 0, & p_1(E) = 0 \\ p_2(G) = 0.5, & p_2(P) = 0.5, & p_2(E) = 0 \\ p_3(G) = 1/3, & p_3(P) = 1/3, & p_3(E) = 1/3 \end{array} \tag{14}$$

Si calculamos un sitema de intervalos, y volvemos a calcular el covexo asociado, se obtiene H' que es igual a H añadiendo las probabilidades de la zona sombreada clara de la figura 2.

□

En general, las informaciones que se pueden representar mediante conjuntos convexos de probabilidades son aquellas que se pueden transformar en un conjunto de restricciones lineales. Por ejemplo, si $A, B \subseteq U$, entonces una desigualdad $P(A|B) \leq 0.4$, se puede transformar en una restricción lineal:

$$0.6 \sum_{u \in A \cap B} p(u) - 0.4 \sum_{u \in B - A} p(u) \leq 0 \tag{15}$$

La diferencia entre estas restricciones y las asociadas a los intervalos de probabilidad es que en estas los coeficientes pueden ser números reales cualesquiera,

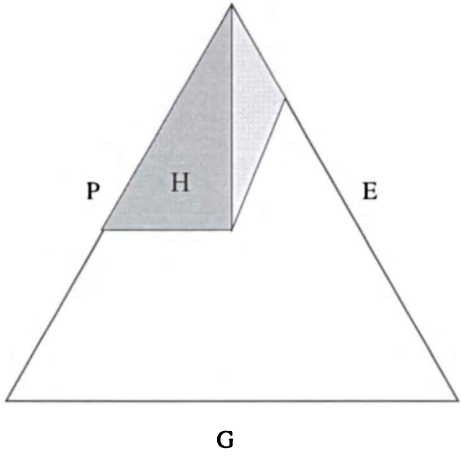


Figura 2. Conjunto convexo e intervalos

y en las asociadas a los intervalos de probabilidad los coeficientes solo pueden ser 0 y 1.

Un conjunto convexo H, podrá venir dado alternativamente por sus puntos extremos $\text{Ext}(H)$ o por un conjunto de restricciones lineales, a ser posible no redundante $\text{Res}(H)$.

Hay algoritmos clásicos de Geometría Computacional que permiten transformar unas representaciones en otras (puntos o restricciones) y minimizar el tamaño de las representaciones. Entre ellos podemos señalar los siguientes:

- *Algoritmos de Cláusula Convexa.*- Estos algoritmos se usan para eliminar todos los puntos no extremos de un conjunto convexo. Al mismo tiempo, calculan un conjunto minimal de restricciones que definen el conjunto convexo. Se pueden encontrar descripciones de estos algoritmos en [15,36].
- *Eliminación de redundancia.*- Estos algoritmos eliminan todas las restricciones redundantes de un conjunto convexo. Un estudio de los distintos algoritmos se encuentra en [24].
- *Algoritmos de enumeración de vértices.*- Estos algoritmos calculan los puntos extremos de un conjunto convexo a partir de un conjunto de restricciones. Una revisión puede encontrarse en [29].

Los conjuntos convexos de probabilidad, no son solo más generales que los intervalos de probabilidad, sino que también permiten una definición sencilla de las operaciones elementales para el cálculo. Vamos a considerar, en primer lugar,

dos operaciones con conjuntos de probabilidades que no tienen una contrapartida en el caso de probabilidades simples.

Supongamos que tenemos una variables n-dimensional $(X_1, \ldots, X_n)$, y que cada X_i toma valores en un conjunto U_i, seguiremos la siguiente notación:

- Para cada $I \subseteq \{1, \ldots, n\}$, X_I representa la variable $(X_i)_{i \in I}$. Esta variable toma valores en $\prod_{i \in I} U_i$ que se notará por U_I. Algunas veces, por simplicidad en el lenguaje, y cuando esté claro por el contexto, identificaremos un conjunto de índices I con la variable X_I.
- Si $u \in U_I$ y $J \subseteq I$, entonces $u^{\downarrow J}$ denotará al elemento de U_J que se obtiene a partir de u borrando las coordenadas en $I - J$.
- Si h es una función de U_I en $\mathbb{R}$, y $J \subseteq I$, entonces la marginal de h a U_J es la función $h^{\downarrow J}$ definida en U_J y dada por, $h^{\downarrow J}(u) = \sum_{v^{\downarrow J} = u} h(v)$.
- Si H es un conjunto convexo de funciones en U_I, con puntos extremos, Ext (H) $= \{h_1, \ldots, h_k\}$, y $J \subseteq I$ entonces la marginalización de H a J es el conjunto convexo dado por,

$$H^{\downarrow J} = H^{-(I-J)} \mathrm{CC}\{h_1^{\downarrow J}, \ldots, h_k^{\downarrow J}\} \tag{16}$$

  donde CC indica la cláusula convexa (el convexo más pequeño que contiene a un conjunto dado).
- $H^{\downarrow J}$ es igual a la marginazación a U_J de todas las funciones h en H.
- $\mathcal{P}_J$ es el conjunto de todas las distribuciones de probabilidad en U_J.
- Supongamos que h es una función de U_I que toma valores en $\mathbb{R}$ y h' una función de U_J en $\mathbb{R}$, entonces la multiplicación de las funciones es una función, $h.h'$, definida en $U_{I \cup J}$ y dada por, $h.h'(u) = h(u^{\downarrow I}).h'(u^{\downarrow J})$.
- Si H es un conjunto convexo de aplicaciones en U_I, y H' es un conjunto convexo en U_J, con $\mathrm{Ext}(H) = \{h_1, \ldots, h_k\}, \mathrm{Ext}(H') = \{h'_1, \ldots, h'_l\}$. Entonces, la combinación de H y H' será el conjunto convexo de aplicaciones en $U_{I \cup J}$, $H \otimes H'$ dado por

$$H \otimes H' = \mathrm{CC}\{h_1.h'_1, .., h_1.h'_l, \ldots, h_k.h'_1, .., h_k.h'_l\} \tag{17}$$

- Si H es un conjunto convexo de aplicaciones en U_I, y H' es un conjunto convexo de aplicaciones en U_J, entonces $H \cap H'$ es el conjunto convexo de aplicaciones h definidas en $U_{I \cup J}$ verificando que $h^{\downarrow I} \in H$ y $h^{\downarrow J} \in H'$.

La primera operación que vamos a considerar para los conjuntos convexos de probabilidades, no tiene una contrapartida directa en probabilidades: la extensión.

Definición 1. Si H es un conjunto convexo de probabilidades sobre las variables X_I y J es un conjunto de índices con $I \subseteq J$, se llama extensión de H a J al conjunto convexo definido en U_J: $H^{\uparrow J} = \{p \in \mathcal{P}_J : p^{\downarrow I} \in H\}$. □

La intersección que definimos a continuación tampoco tiene sentido cuando nuestro modelo admite una única distribución de probabilidad.

Definición 2. Si H_1 es un conjunto convexo de probabilidades en U_I y H_2 es un conjunto convexo de probabilidades en U_J, la intersección de estos dos convexos es el conjunto $H_1 \cap H_2 = \{p \in \mathcal{P}_{I \cup J} : p^{\downarrow I} \in H_1, p^{\downarrow J} \in H_2\}$. □

La intersección de conjuntos convexos es igual a la intersección clásica de conjuntos de la extensión de ambos convexos a un marco común.

En general, la mayoría de las operaciones de probabilidades tienen una extensión directa. Aunque no es del todo preciso, podemos decir que una operación probabilística se generaliza al caso de los conjuntos convexos, repitiendo la operación para cada una de las probabilidades de los convexos, y tomando la cláusula convexa cuando el resultado no sea un conjunto convexo. Esta afirmación, aunque nos puede servir de guía es un poco simplista, y hay casos que no se ajustan a ella totalmente, o en los que esta idea permita iterpretaciones distintas.

Si H es un conjunto convexo de probabilidades sobre X_I y $J \subseteq I$, se llama información marginal sobre X_J al conjunto convexo $H^{\downarrow J}$.

Una información condicional sobre X_I dado X_J será un conjunto covexo de distribuciones de probabilidad sobre X_J condicionadas a X_I.

Si partimos de una información *a priori* sobre X_I, H_1, y una información condicional sobre X_J dado X_I, H_2, entonces la información global inducida sobre $X_{I \cup J}$ al conjunto convexo $H_1 \otimes H_2$.

La definición de información condicionada no es un tema sencillo en el caso de las probabilidades imprecisas. Existen distintos enfoques. Un estudio detallado se puede encontrar en [31,7].

Aquí daremos la que es quizás la definición más sencilla y más extendida. Si H es una información sobre X_I y $O \equiv (X_I \in A)$ es una observación sobre esta variable, la información condicionada es el conjunto convexo $H|A = \{p(.|A) : p \in H\}$ donde $p(.|A)$ es la distribución de probabilidad condicionada.

Observemos como este conjunto convexo puede calcularse facilmente a partir del conjunto $H_A = H \otimes \{l_A\}$ donde $_A$ es la función indicadora (o verosimilitud) de A.

La independencia no es un concepto ni mucho menos obvio en el caso de los conjuntos convexos de probabilidades. De Campos y Moral [14] estudian cinco definiciones distintas de este concepto. Este es un caso en el que la regla general que dimos anteriormente falla. No podemos considerar que existe independencia

bajo un convexo cuando existe indepencia probabilística para cada una de las probabilidades del convexo.

Definición 3. Si H es un convexo de probabilidades sobre $X_{I\cup J\cup K}$ diremos que X_I es fuertemente independiente de X_J dado X_K si y solo si exiten dos conjuntos convexos H_1 definido en $U_{I\cup K}$ y H_2 definido en $U_{I\cup K}$ tales que

$$H = H_1 \otimes H_2$$

□

Esta es la definición más apropiada para desarrollar algoritmos de propagación, ya que implica una descomposición del conjunto convexo.

Dependiendo de las operaciones que se vayan a realizar con un conjunto convexo, una representación puede ser más adecuada que otra. Por ejemplo, para la intersección, la representación por restricciones es la más apropriada: sólo hay que calcular la unión de las restricciones.

Para la combinación, $\otimes$, la representación por puntos extremos es más apropriada, ya que estas operaciones se expresan directamente de esta forma.

3 Algoritmos Basados en Valuaciones

Con el término valuación aludimos de forma general al concepto de representación matemática de una información. Dependiendo del modelo que se utilice una valuación será una distribución de probabilidad, un conjunto convexo de probabilidades o un conjunto de fórmulas lógicas. Supondremos que para cada $I \subseteq \{1,\ldots,n\}$ existe un conjunto $\mathcal{V}_I$ de valuaciones definidas en el producto cartesiano, U_I.

$\mathcal{V}$ será el conjunto de todas las valuaciones $\mathcal{V} = \cup_{I\subseteq\{1,\ldots,n\}}\mathcal{V}_I$. Si V es una valuación perteneciente a $\mathcal{V}_I$ (que informa sobre X_I), entonces diremos que el conjunto de definición de V es I (ó U_I), lo que se notará como $s(V) = I$.

Se supone que existen dos operaciones básicas en el conjunto de las valuaciones (ver Zadeh [53], Shenoy, Shafer [39,41]): marginalización y combinación. La marginalización de una valuación definida en un conjunto U_I consiste en obtener la información inducida por ella (proyección) en un conjunto menos preciso U_J ($J \subseteq I$). Si V es la valuación definida en U_I, su marginalización a U_J se nota por $V^{\downarrow J}$. La combinación resume en una sola valuación la información de dos valuaciones. Si las dos valuaciones combinadas son V_1 y V_2 definidas en U_I y U_J, respectivamente su combinación se notará como $V_1 \otimes V_2$ y estará definida en $U_{I\cup J}$. En resumen tenemos:

- *Marginalización.-* Si $J \subseteq I$ and $V_1 \in \mathcal{V}_I$ entonces la marginalización de V_1 a J es una valuación $V_1^{\downarrow J}$ perteneciente a $\mathcal{V}_J$.
- *Combinación.-* Si $V_1 \in \mathcal{V}_I$ y $V_2 \in \mathcal{V}_J$, entonces su combinación es una valuación $V_1 \otimes V_2$ perteneciente a $\mathcal{V}_{I \cup J}$.

Shenoy y Shafer [39,41], introducen los siguientes axiomas para estas operaciones:

Axioma 1 $V_1 \otimes V_2 = V_2 \otimes V_1$, $(V_1 \otimes V_2) \otimes V_3 = V_1 \otimes (V_2 \otimes V_3)$.
Axioma 2 Si $I \subseteq J \subseteq K$, y $V \in \mathcal{V}_K$, entonces $(V^{\downarrow J})^{\downarrow I} = V^{\downarrow I}$.
Axioma 3 Si $V_1 \in \mathcal{V}_I$, $V_2 \in \mathcal{V}_J$, entonces $(V_1 \otimes V_2)^{\downarrow I} = V_1 \otimes V_2^{\downarrow (J \cap I)}$.

Cano, Delgado y Moral [6] introdujeron dos axiomas adicionales que son útiles en muchas ocasiones:

Axioma 4 *Elemento Neutro.-* Para cada I existe una valuación $V_0^I \in \mathcal{V}_I$ tal que para toda valuación $V \in \mathcal{V}_J$ con $I \subseteq J$ se tiene que ${V_0}^I \otimes V = V$.
Axioma 5 *Contradicción.-* Existe una y sólo una valuación, V_c, definida en $U_1 \times \ldots \times U_n$, tal que $\forall V \in \mathcal{V}$, $V_c \otimes V = V_c$.

Los primeros tres axiomas contienen las propiedades necesarias para deducir los algoritmos de propagación. El tercer axioma es de particular importancia para el desarrollo de los mismos, ya que nos permite calcular $(V_1 \otimes V_2)^{\downarrow I}$ sin necesidad de calcular de forma explícita $(V_1 \otimes V_2)$, definida en $U_{I \cup J}$. Lo mismo puede hacerse calculando $V_2^{\downarrow (J \cap I)}$ y combinando el resultado con V_1. En este último caso, sólo necesitamos trabajar con valuaciones definidas en U_J, $U_{I \cap J}$ y U_I, lo que es mucho más eficiente.

A continuación ilustramos estas ideas en el caso particular de la Teoría de la Probabilidad.

Ejemplo 6. Desde el punto de vista de la Teoría de la Probabilidad una valuación es la representación de una información probabilística sobre algunas de las variables, $X_I, I \subseteq \{1, \ldots, n\}$. Más concretamente, si tenemos tres variables (X_1, X_2, X_3) que toman valores en $U_1 \times U_2, \times U_3$, donde $U_i = \{u_{i1}, u_{i2}\}, i = 1, 2, 3$, entonces una valuación puede ser una distribución de probabilidad sobre X_1,

$$p(u_{11}) = 0.8$$
$$p(u_{12}) = 0.2$$

Puede ser también una distribución de probabilidad sobre X_3 dada X_2,

$$p(u_{31}|u_{21}) = 0.9 \; p(u_{32}|u_{21}) = 0.1$$
$$p(u_{31}|u_{22}) = 0.6 \; p(u_{32}|u_{22}) = 0.4$$

Desde un punto de vista general una valuación sobre X_I es una aplicación no negativa,

$$p : U_I \longrightarrow \mathbb{R}_0^+$$

donde $\mathbb{R}_0^+$ es el conjunto de los reales no negativos.

Las valuaciones probabilísticas se denominan también potenciales.

La combinación se define mediante la multiplicación punto a punto. Si p_1 y p_2 son funciones no negativas definidas en U_I y U_J respectivamente, entonce $p_1 \otimes p_2$ es una aplicación definida en $U_{I \cup J}$ y que toma valores en $\mathbb{R}_0^+$, de acuerdo con la siguiente expresión,

$$p_1 \otimes p_2(u) = p_1(u^{\downarrow I}).p_2(u^{\downarrow J}), \forall u \in U_{I \cup J}$$

Esta operación se usa en la Teoría de la Probabilidad para combinar una información marginal con una una probabilidad condicional. También, si obviamos el factor de normalización, se puede usar para calcular la información condicional *a posteriori*: condicionar al conjunto A puede verse como la multiplicación de la probabilidad *a priori* con la verosimilitud asociada a A (su función característica: $l_A(u) = 1$, si $u \in A$; $l_A(u) = 0$, en otro caso). La marginalización se define de la forma usual: Si p es una valuación definida en U_I y $J \subseteq I$, entonces

$$p^{\downarrow J}(v) = \sum_{u^{\downarrow J} = v} p(u), \qquad \forall v \in U_J$$

En el caso de la Teoría de la Probabilidad, el elemento neutro es la valuación:

$$p_0(u) = 1, \forall u \in U_I$$

Y la contradicción es la función identicamente igual a 0,

$$p_c(u) = 0, \qquad \forall u \in U_1 \times \ldots \times U_n$$

□

En general, el problema que abordan los conocidos como algoritmos de propagación es el siguiente: tenemos un conjunto finito de valuaciones $R = \{V_1, \ldots, V_m\}$, donde cada V_i se define en un referencial I_i. Estamos interesados en calcular la proyección o marginalización en una variable de interés X_j de la combinación de todas las valuaciones en R. Es decir en calcular [39]:

$$R_j = \left(\bigotimes R\right)^{\downarrow\{j\}} = (V_1 \otimes \ldots \otimes V_m)^{\downarrow\{j\}} \tag{18}$$

para un valor $j \in \{1, \ldots, n\}$.

El conjunto R representa toda la información de que disponemos. En el contexto de las probabilidades, en R suele haber dos tipos de informaciones: informaciones genéricas que determinan una distribución de probabilidad global y observaciones sobre algunas de estas variables para un caso particular. Normalmente es muy difícil poder especificar la distribución global de forma directa. Si partimos de una grafo dirigido acíclico que expresa las independencias del problema mediante el criterio de d-separación [49,35], entonces una distribución global puede obtenerse combinando una distribución de probabilidad para cada nodo condionada a los valores de sus padres en el grafo. Todas ellas formarán parte de R. Para las observaciones, si tenemos que $X_j = u_j$, entonces se añade una valuación que es una función definida en U_j que toma el valor 1 en u_j y 0 en el resto.

El cálculo se realiza mediante algoritmos de propagación [39,41,35,27,6]. Estos algoritmos se pueden aplicar a cualquier modelo de representación de la incertidumbre que sea un sistema de valuaciones que verifique los axiomas anteriores. Esencialmente, el cálculo de R_j se lleva a cabo trasformando el conjunto $R = \{V_1, \ldots, V_m\}$ de acuerdo con el siguiente paso básico (*Borrado de k*):

- Sea k un índice, $k \neq j$. Consideremos $K = \{V_i \in R : k \in s(V_i)\}$ y $L = s(\bigotimes K) - \{k\}$. Entonces R se transforma en el conjunto

$$R - K \cup \{(\otimes K)^{\downarrow L}\} = R - K \cup \{(\otimes K)^{-k}\} \tag{19}$$

Este paso se repite (borrando todos los índices k distintos de j) hasta que todas las valuaciones estén definidas en el referencial $\{j\}$. La valuación que buscamos, R_j, es la combinación de todas las valuaciones que quedan en R.

Este procedimiento es, en general, más eficiente que combinar todas las valuaciones y marginalizar después. Pensemos que el tamaño de una representación es en la mayoría de los casos, al menos, proporcional al tamaño del referencial U_I, y este tamaño es el producto de los elementos de cada conjunto U_i. Este es el caso, por ejemplo, de una distribución de probabilidad. Si combinamos todas las valuaciones en R obtenemos una valuación en $U_1 \times \ldots \times U_n$ lo que, para valores moderados de n, ya no se puede representar. Sin embargo, en el algoritmo anterior las valuaciones están definidas en referenciales más pequeños (involucran menos variables): $s(\bigotimes K)$.

A veces, por ejemplo cuando se quiere calcular la información marginal R_j para varias variables X_j, es conveniente organizar los cálculos en lo que se llama un árbol de grupos. Este árbol de grupos, T_G, es un árbol no dirigido en el que los nodos son grupos de variables, G, y en los que se cumplen las dos condiciones siguientes [27]:

1. Para toda valuación $V \in R$, existe un nodo G del árbol tal que $s(V) \subseteq G$.

2. Si G_1 y G_2 son dos nodos tales que $G_1 \cap G_2 \neq \emptyset$, entonces para todo nodo G en el camino que une G_1 y G_2 se tiene que $(G_1 \cap G_2) \subseteq G$.

Este árbol de grupos se puede obtener por un proceso de triangulación [25,4] a partir de un grafo no dirigido que tiene un nodo por cada variable $X_1, \ldots, X_n$ y tal que X_i y X_j están unidos si y solo si existe una valuación $V \in R$ para la que $\{i, j\} \subseteq s(V)$.

En general, todos los algoritmos comienzan asignando cada valuación $V \in R$ a un nodo del árbol de grupos G tal que $s(V) \subseteq G$. Sea $R(G)$ el conjunto de valuaciones asignadas a G. A continuación se calcula, para cada nodo G la valuación

$$V_G = \otimes_{V \in R(G)} V \tag{20}$$

Los algoritmos trabajan entonces mandando mensajes entre los nodos adyacentes del árbol de grupos. Hay dos modelos esenciales: el de Shafer y Shenoy [39] y el conocido como HUGIN [23]. Ambos serán brevemente descritos a continuación. Más detalle puede encontrarse en los libros de Jensen [22] y de Castillo, Hadi y Gutiérrez [9].

En general, puede decirse que el algoritmo HUGIN es el más eficiente (si la división se tiene la misma complejidad que la combinación o la marginalización). Sin embargo, requiere unas condiciones de aplicación un poco más fuertes que el algoritmo de Shafer y Shenoy. Los algoritmos para conjuntos convexos se basarán en la arquitectura de Shafer y Shenoy, que es la que describiremos a continuación.

3.1 Arquitectura de Shafer y Shenoy

Se supone que existen dos mensajes para cada par de nodos adyacentes G_1 y G_2, uno de G_1 a G_2: V_{G_1,G_2} y otro de G_2 a G_1: V_{G_2,G_1}. La operación fundamental de mandar mensaje entre dos nodos G_1 y G_2 consiste en realizar el siguiente cálculo:

$$V_{G_1,G_2} = \left(V_{G_1} \otimes \left(\bigotimes_{G \in Ady(G_1,G_2)} V_{G,G_1} \right) \right) \tag{21}$$

donde $Ady(G_1, G_2)$ es el conjunto de todos los grupos adayacentes a G_1 excepto G_2. En lo que sigue $Ady(G_1)$ notará todos los nodos adyacentes a G_1.

El algoritmo de propagación asociado a esta arquitectura consiste en el cálculo de todos los mensajes lo que se hace mediante dos recorridos del árbol de grupos. Para ello se elige un nodo G como raíz o pivote y en una primera fase se mandan

mensajes de las hojas al nodo raíz, y en la segunda se distribuye la información desde el nodo raíz a las hojas. De manera más concreta la primera parte puede representarse como:

Primera(G)
- **Para todos los nodos** $G' \in Ady(G)$
  - **Pide**(G', G)

donde $Pide(G, G')$ es como sigue:

Pide(G', G)
- **Para todo nodo** $G'' \in Ady(G', G)$
  - **Pide**(G'', G')
- **Mandar mensaje de** G' **a** G**.**

El segundo recorrido en el grafo distribuye la información a partir del nodo raíz G. El algoritmo se puede expresar de la siguiente forma.

Segunda(G)
- **Para todo nodo** $G' \in Ady(G)$
  - **Distribuye**(G', G)

El procedimiento *Distribuye* es como sigue,

Distribuye(G', G)
- **Mandar un mensaje de** G **a** G'
- **Para cada** $G'' \in Ady(G', G)$
  - **Distribuye**(G'', G')

Despues de aplicar este algoritmo, para calcular R_j solo tenemos que elegir un grupo de variables G al que pertenezca X_j y calcular,

$$R_j = \left(V_G \otimes \left(\bigotimes_{G' \in Ady(G)} V_{G',G} \right) \right)^{\downarrow j} \tag{22}$$

4 Cálculo con Restricciones Probabilísticas

El problema lo podemos enunciar de la siguiente forma. Tenemos un conjunto de restricciones lineales, cada una de ellas, sobre un conjunto de variables distinto: $\mathcal{R} = \{r_i : r_i \text{ restricción lineal sobre } X_{I_i}\}$. Tenemos un suceso $A \subseteq U_J$ y queremos calcular el máximo y el mínimo que puede tomar la probabilidad de A sujeto a que se verifican las restricciones lineales en $\mathcal{R}$.

Ejemplo 7. Supongamos que sabemos que

$$P(\text{Soltero}|\text{Estudiante}) \geq 0.9, P(\text{Paro}|\text{Estudiante}) \geq 0.95, P(\text{Paro}|\text{Soltero}) \geq 0.7$$

y que queremos calcular el máximo y el mínimo valor que puede tomar la probabilidad de ser soltero una vez que se está parado. □

Un método de solución obvio que funciona bien para problemas de tamaño pequeño es el uso de la programación lineal, ya que tenemos un problema de optimización con restricciones lineales.

Una referencia fundamental para estos tipos de problemas es el trabajo de Hansen y Jaumard [19]. Para simplificar supongamos que tenemos variables X_i bivaluadas que toman los valores x_i y $\overline{x_i}$. Supongamos que tenemos una serie de restricciones como las siguientes,

$$\begin{array}{l} 0.3p(x_1) - 2p(\overline{x_2}) \geq 0 \\ 0.4p(x_2, \overline{x_3}) - p(x_2) \geq 0.1 \\ p(\overline{x_1}, x_2, x_3) = 0 \end{array} \tag{23}$$

y queremos conocer cuales son los límites de la probabilidad de $p(x_2, \overline{x_3})$. El principal problema es expresar las restricciones en un marco común, es decir en términos de valores $p(x_i, x_j, x_k)$. Eso implica que, por ejemplo, $p(x_2, \overline{x_3})$ se transforme en $p(x_1, x_2, \overline{x_3}) + p(\overline{x_1}, x_2, \overline{x_3})$. El término $p(x_1)$ se tendrá que expander cómo

$$p(x_1, x_2, x_3) + p(x_1, x_2, \overline{x_3}) + p(x_1, \overline{x_2}, x_3) + p(x_1, \overline{x_2}, \overline{x_3})$$

Así de esta forma este problema se expresaría como un problema de programación lineal sobre 8 dimensiones: una por cada valor conjunto de las variables X_1, X_2, y X_3. La solucion es muy sencilla por métodos de programación lineal.

Las dificultades aparecen cuando el número de variables no es tan reducido como en este caso. Si tenemos n variables con dos valores cada una, la dimensión del problema de programación lineal que debemos de resolver es de 2^n, lo cual es

a todas luces excesivo. Sin embargo, esto no quiere decir que el problema sea totalmente intratable. Existen dos métodos exactos que permiten una resolución de problemas que involucran un número alto de variables: los métodos de generación de columnas [19,21] y los basados en algoritmos de propagación [48,30]. Existen también una gran cantidad de métodos aproximados basados en el uso de reglas locales que permiten la obtención de cotas para sucesos de interés a partir de las cotas conocidas.

Antes de pasar a describir de forma somera estos métodos, vamos a indicar que si nuestro objetivo es obtener cotas sobre una probabilidad condicional, entonces lo que tenemos es un problema de programación fraccional. El método de Charnes y Cooper [10] transforma un problema de programación fraccional en un problema de programación lineal mediante la adición de un parámetro adicional. Por lo que su resolución es totalmente análoga al caso anterior.

Ejemplo 8. El problema de programación fraccional:

$$\begin{array}{ll} \text{Min/Max} & \frac{p(x_1,x_2)}{p(x_1,x_2)+p(x_1,\overline{x_2})+p(\overline{x_1},x_2)} \\ \text{Sujeto a:} & p(x_1,x_2)+p(x_1,\overline{x_2})=0.8 \\ & p(x_1,x_2)+p(\overline{x_1},x_2)=0.5 \\ & p(x_1,x_2)+p(x_1,\overline{x_2})+p(\overline{x_1},x_2)+p(\overline{x_1},\overline{x_2})=1 \\ & p(x_1,x_2),p(x_1,\overline{x_2}),p(\overline{x_1},x_2),p(\overline{x_1},\overline{x_2})\geq 0 \end{array} \tag{24}$$

se transforma en el problema de programación lineal:

$$\begin{array}{ll} \text{Min/Max} & p(x_1,x_2) \\ \text{Sujeto a:} & p(x_1,x_2)+p(x_1,\overline{x_2})=t.0.8 \\ & p(x_1,x_2)+p(\overline{x_1},x_2)=t.0.5 \\ & p(x_1,x_2)+p(x_1,\overline{x_2})+p(\overline{x_1},x_2)+p(\overline{x_1},\overline{x_2})=t \\ & p(x_1,x_2)+p(x_1,\overline{x_2})+p(\overline{x_1},x_2)=1 \\ & p(x_1,x_2),p(x_1,\overline{x_2}),p(\overline{x_1},x_2),p(\overline{x_1},\overline{x_2})\geq 0 \end{array} \tag{25}$$

□

4.1 Algoritmos para Restricciones basados en Generación de Columnas

El nombre de *generación de columnas* proviene de la expresión matricial de un problema de programación lineal. En nuestro caso el número de columnas de la matriz de coeficientes es exponencial en función del número de variables.

Aquí solo expondremos la idea principal de este algoritmo ya que su descripción técnica es demasiado compleja. Esta consiste en mantener un número pequeño de columnas representadas de forma explícita. El resto de ellas se obtendrían a partir de las ecuaciones tal y cómo venían especificadas al principio (con un número pequeño de variables). En cada paso, la columna entrante al problema del símplex se obtiene resolviendo un subproblema auxiliar.

Supongamos un problema de programación lineal:

$$\begin{array}{rl} \text{Min} & z = cx \\ \text{Sujeto a :} & Ax = b, \\ & x \geq 0 \end{array} \tag{26}$$

Una base es un conjunto de columnas (o variables) igual al número de restricciones. Sea B la matriz de variables básicas. Si c_B es el subvector del vector de costos correspondiente a las variables básicas y c_k el coeficiente de la componente x_k, entonces calcular la variable que entra en la base equivale a calcular la columna A^j para la que el valor $c_j - c_B B^{-1} A^j$ es mínimo. Una vez calculada la variable entrante en la base, se calcula la variable saliente por el procedimiento usual del símplex, y se prosigue realizando iteraciones hasta que se alcance la condición de optimalidad.

En nuestro problema, cada columna se identifica con una variable $p(z_1, \ldots, z_n)$ donde cada $z_i = x_i$ ó $z_i = \overline{x_i}$. El valor $c_j - c_B B^{-1} A^j$ se expresa como una función de n variables $z_1, \ldots, z_n$ cada una de las cuales puede tomar dos valores: $1(x_i), 0(\overline{x_i})$. Lo que depende de estos valores $(z_i = 0, z_i = 1)$ es la columna A^j. Supongamos que la ecuación i-ésima original tiene un coeficiente $0.4p(x_2, \overline{x_3})$ entonces para todos los $(z_1, \ldots, z_n)$ que tengan un $z_2 = 1, z_3 = 1$ se tiene que el elemento i de la columna A_j tiene un valor 0.4. Esto mismo se obtiene expresando que este elemento es $0.4z_2(1 - z_3)$. Una vez expresados todos los elementos de A^j como productos de variables z_i y $(1 - z_j)$, nos queda un problema de optimización booleana. Para este problema se pueden aplicar técnicas exactas o aproximadas como el enfriamiento estocástico, o los algoritmos genéticos.

Estos métodos han permitido obtener la solución de problemas en tiempo razonable que involucran a miles de variables, lo que implicaría un número de columnas realmente intratable.

4.2 Algoritmos de Propagación para Restricciones Probabilísticas

Otro procedimiento alternativo para tratar problemas con un número elevado de variables es el uso de algoritmos de propagación. La idea es muy sencilla: cada restricción lineal se puede transformar en un conjunto convexo. Ahora bien, los

conjuntos convexos con las operaciones de intersección y marginalización verifican la axiomática de Shafer y Shenoy (una demostración puede encontrarse en Verdegay [48]). Y por tanto, se pueden aplicar los algoritmos de propagación que hemos descrito en la sección 3.

El elemento neutro de este conjunto de valuaciones en U_I es $\mathcal{P}_I$: el conjunto de todas las dsitribuciones de probabilidad en U_I. La contradicción es el conjunto vacío.

El problema fundamental de estos algoritmos de propagación es que las operaciones de combinación y marginalización necesitan dos representaciones distintas para poder llevarse a cabo de manera eficiente: para la marginalización la representación por puntos extremos es más apropriada y para la combinación es más apropriada la representación por restricciones lineales. Se puede pensar en usar algoritmos que transforman una representación en otra, pero recientemente Verdegay [48] ha presentado algoritmos que realizan la marginalización directamente con restricciones lineales con más eficiencia que el cambio de representación.

4.3 Métodos Basados en el Uso de Reglas Locales

Existen numerosos métodos basados en el uso de reglas locales: Amarger, Dubois y Prade [1], Thöne [45,46], Salo [37].

Su uso es más eficiente que las técnicas anteriores, sin embargo, en general, la variedad de restricciones que se pueden usar está bastante limitada y en la mayoría de los casos no se obtienen cotas óptimas. Son reglas que nos permiten obtener nuevas cotas a partir de las conocidas. Una aplicación sistemática de las mismas nos permite obtener cotas para los sucesos de interés. No tenemos espacio en este trabajo para hacer una revisión de todos los tipos de reglas. Sin embargo, y a modo de ilustración, daremos dos de ellas.

La primera es la regla de concatenación de Thöne. Thöne [45] considera variables proposicionales $\{A_1, \ldots, N_n\}$, y reglas intervalares del tipo $A \overset{x_1,x_2}{\longrightarrow} B$, con el significado de $P(A) > 0$ y $0 \leq x_1 \leq P(B|A) \leq x_2 \leq 1$.

Se puede trabajar también con reglas bidireccionales $A \underset{y_1,y_2}{\overset{x_1,x_2}{\longleftrightarrow}} B$ donde ($x_2 = 0 \Leftrightarrow y_2 = 0$) con el significado de $A \overset{x_1,x_2}{\longrightarrow} B$ y $B \overset{y_1,y_2}{\longrightarrow} A$.

Regla de concatenación.- Si $A \underset{v_1,v_2}{\overset{w_1,w_2}{\longleftrightarrow}} B$ y $B \underset{y_1,y_2}{\overset{x_1,x_2}{\longleftrightarrow}} C$, deducir $A \underset{r_1,r_2}{\overset{z_1,z_2}{\longleftrightarrow}} C$ donde

$$z_1 = f_1(w_1, v_1, x_1) = \begin{cases} \frac{w_1}{v_1}\text{Max}\ \{0, w_1 + x_1 - 1\} & \text{si } v_1 > 0 \\ w_1 & \text{si } v_1 \text{ y } x_1 = 1 \\ 0 & \text{en otro caso} \end{cases} \qquad (27)$$

$$z_2 = f_2(w_1, w_2, v_1, x_2, y_1) =$$

$$\begin{cases} \text{Min } \{1, w_2 + \tau(1-y_1), 1-w_1+\tau y_1, \tau\} & \text{con } \tau = \frac{w_2 x_2}{v_1 y_1} \\ & \text{si } v_1 > 0 \text{ y } y_1 > 0 \\ \text{Min } \left\{1, 1-w_1+\frac{w_2 x_2}{v_1}\right\} & \text{si } v_1 > 0 \text{ y } y_1 = 0 \\ 1-w_1 & \text{si } v_1 = 0 \text{ y } x_2 = 0 \\ 1 & \text{en otro caso} \end{cases} \quad (28)$$

La segunda regla es el Teorema de Bayes generalizado [1].

Teorema de Bayes Generalizado.- Dados k conjuntos $A_1, A_2, \ldots, A_k$ con $k > 2$ y las reglas, $A_i \overset{x_1^i, x_2^i}{\underset{y_1^i, y_2^i}{\longleftrightarrow}} A_{i+1}$, $A_k \overset{w_1, w_2}{\longrightarrow} A_1$, entonces deducir $A_1 \overset{z_1, z_2}{\longrightarrow} A_k$, con

$$\begin{aligned} z_1 &= w_1 \prod_{i=1}^{k-1} \frac{y_1^i}{x_2^i} \\ z_2 &= w_2 \prod_{i=1}^{k-1} \frac{y_2^i}{x_1^i} \end{aligned} \quad (29)$$

5 Algoritmos para Probabilidades Imprecisas basados en Relaciones de Independencia

El problema se puede plantear de la siguiente forma. Supongamos que tenemos un cojunto convexo global, que debido a las relaciones de independencia de las variables de un problema se puede descomponer en combinación de un producto de conjuntos convexos definidos para conjuntos de variables reducidos:

$$H = H_1 \otimes \cdots \otimes H_m \quad (30)$$

Supongamos que tenemos una serie de observaciones para algunas de las variables: $e = \{X_{l_1} = u_{l_1}, \ldots, X_{l_k} = u_{l_k}\}$. Nuestro objetivo es calcular el conjunto convexo condicional $H|e$ marginalizado en una o varias variables de interés, X_j: $(H|e)^{\downarrow j}$.

Como ya indicamos en la sección 2, para calcular $H|e$ es suficiente calcular

$$H \otimes \{l_{i_1}\} \otimes \cdots \otimes \{l_{i_k}\} \quad (31)$$

donde l_{i_1} es la verosimilitud asociada a la observación: $X_{i_1} = u_{i_1}$, es decir la función definida en U_{i_1} que toma el valor 1 en u_{i_1} y 0 en el resto.

Teniendo en cuenta la descomposición de H, nuestro objetivo es calcular:

$$(H_1 \otimes \cdots \otimes H_m \otimes \{l_{i_1}\} \otimes \cdots \otimes \{l_{i_k}\})^{\downarrow j} \tag{32}$$

De nuevo a este problema se le puede aplicar la axiomática de Shafer y Shenoy [41]. La operación de combinación es distinta que en el caso de restricciones probabilísticas y la de marginalización es la misma. Los detalles de la aplicación de la axiomática a este caso se pueden consultar en Cano, Moral y Verdegay López [8]. Dos conjuntos convexos se consideran equivalentes si son proporcionales. El elemento neutro en U_I es un cojunto con una única distribución de probabilidad: la distribución uniforme.

Ahora la representación por puntos extremos es la más apropriada para ambas operaciones, por lo que ésta es la que usualmente se ha empleado en la literatura. Existen algunos estudios que usan restricciones, o más concretamente intervalos de probabilidad, pero estos utilizan reglas de propagación que siendo óptimas en el cálculo local, no obtienen los intervalos correctos, sino intervalos mucho más amplios. Este es el caso de los enfoques de Breese y Fertig [2] y Tessem [44].

Desde el punto de vista del cálculo, el principal problema es que si combinamos dos convexos H_1 y H_2. El número de puntos extremos de la combinación puede llegar a ser el número de extremos de H_1 por el número de puntos extremos de H_2. Además detectar qué multiplicación de un extremo de H_1 por uno de H_2 no es extremo en el producto implica la aplicación de un algoritmo de cláusula convexa, que llevan un importante coste computacional asociado.

5.1 Transformación en un Problema de Optimización Combinatoria

Cano y Moral [3,5] han propuesto la transformación del problema de marginalización en un problema de optimización combinatoria, al que se puedan aplicar técnicas como las de enfriamiento estocástico o algoritmos genéticos.

La idea es añadir una nueva variable T_i por cada convexo H_i, y que se llamará variable transparente asociada a H_i. T_i tendrá tantos casos como puntos extremos tenga el conjunto convexo H_i.

Cada uno de los convexos H_i también se transforman en otro convexo H_i', que está definido para todas las variables de H_i más T_i. Supongamos que I_i es el conjunto de variables de H_i, entonces a cada distribución extrema p en H_i se le asigna un valor t_p entre los valores posibles de T_i, mediante una aplicación biyectiva. El convexo H_i' se calcula considerando para cada distribución p extrema en H_i una distribución asociada p' en H_i' y que viene dada por

$$p'(u_i, t) = \begin{cases} p(u_i) & \text{si } t = t_p \\ 0 & \text{en otro caso} \end{cases} \tag{33}$$

Esta transformación no afecta a la solución final del problema y permite una parametrización del mismo. Para cada configuración de valores de las variables transparentes $(t_1, \ldots, t_m)$, se determina una único elemento para cada p_i en cada convexo H_i (esta distribución es aquella que verifica, $t_{p_i} = t_i$). El producto de las distribuciones $p_1.p_2.\ldots.p_m$ es una distribución global para todas las variables que denotaremos por $p_{t_1,\ldots,t_m}$. Además todos los puntos extremos de H se pueden obtener de esta forma.

Como consecuencia, si tenemos una variable de interés X_j y queremos calcular el Max (Min) de $p(X_j = u_j|e)$ donde $p \in H$, el problema se puede plantear como,

$$\text{Max } \{p_{t_1,\ldots,t_m}(X_j = u_j|e) : t_i \text{ es un valor de } T_i, \quad i =, \ldots, m\} \qquad (34)$$

Este problema es muy similar al de la abducción parcial. Ya que si consideramos el problema extendido con las variables artificiales tenemos que maximizar en estas variables y sumar en el resto: las variables originales X_j. Existen algunas diferencias como son que no hay distribución *a priori* para estas variables y que $p_{t_1,\ldots,t_m}(e)$ no es constante ya que cambia al variar los valores de las variables transparentes. Sin embargo, el valor del objetivo para cada configuración $(t_1, \ldots, t_m)$ también se puede calcular mediante propagaciones puramente probabilísticas: dos en este caso. En una de ellas con las observaciones de e que permite calcular $p_{t_1,\ldots,t_m}(e)$ y otra en las que se añade la observación $X_j = u_j$ que permite calcular $p_{t_1,\ldots,t_m}(e, x_j = u_j)$. El valor del objetivo es el cociente de estas dos cantidades. En algunos casos, la segunda cantidad se puede calcular directamente, a partir de la primera si toda la información se encuentra propagada a un nodo que contenga X_j.

En definitiva, este planteamiento del problema ha permitido aplicar algoritmos de optimización combinatoria, entre los que podemos destacar algoritmos genéticos [5], de enfriamiento estocástico [3] o de gradiente ascendiente [11,12].

6 Conclusiones

En este trabajo hemos realizado una aproximación al problema del cálculo con probabilidades imprecisas. La primera observación es que existe una mayor variedad de planteamientos posibles y procedimientos de resolución.

Otra observación es que la complejidad de los algoritmos es mucho mayor que en el caso probablístico. Pensemos que en algunos casos tenemos un problema de optimización en el que para calcular el objetivo necesitamos realizar una propagación probabilística.

Sin embargo, a pesar de estos inconvenientes opinamos que se ha avanzado bastante en la solución de estos problemas. Existen implementaciones que permiten trabajar con probabilidades imprecisas en problemas en los que el número de variables no sea excesivamente grande o en situaciones en las que la imprecisión solo se encuentra en un número reducido de variables. También esperamos que en el futuro el uso de algoritmos más eficientes de Geometría Computacional y el desarrollo de algoritmos de optimización mejor adaptados al problema permita resolver problemas de tamaño cada vez mayor.

Existen varias implementaciones de algoritmos de propagación con probabilidades imprecisas. En el Departamento de Ciencias de la Computación e Inteligencia Artificial tenemos el sistema *Entorno* que implementa algoritmos exactos y aproximados. Como sistema más destacable podemos señalar el sistema *Java Bayes* de Fabio Cozman que es de libre disposición y que se puede obtener en la dirección URL: http://www.cs.cmu.edu/ javabayes/Home/

Referencias

1. S. Amarger, D. Dubois y H. Prade. Constraint propagation with imprecise conditional probabilities. En: B.D. Ambrosio, Ph. Smets y P.P. Bonissone (eds.), *Proceedings of the 7th Conference on Uncertainty in Artificial Intelligence*, págs. 26–34. Morgan & Kaufmann, 1991.
2. J.S. Breese y K.W. Fertig. Decision making with interval influence diagrams. En: L.N. Kanal P.P. Bonissone, M. Henrion (ed.), *Uncertainty in Artificial Intelligence, 6*, págs. 467–478. Elsevier, 1991.
3. A. Cano, J.E. Cano y S. Moral. Convex sets of probabilities propagation by simulated annealing. En: *Proceedings of the Fith International Conference IPMU'94*, págs. 4–8, Paris, 1994.
4. A. Cano y S. Moral. Heuristic algorithms for the triangulation of graphs. En: B. Bouchon-Meunier, R.R. Yager y L.A. Zadeh (eds.), *Advances in Intelligent Computing*, págs. 98–107. Springer Verlag, 1995.
5. A. Cano y S. Moral. A genetic algorithm to approximate convex sets of probabilities. En: *Proceedings of Information Processing and Management of Uncertainty in Knowledge-Based Systems Conference (IPMU' 96) Vol. 2*, págs. 859–864, 1996.
6. J.E. Cano, M. Delgado y S. Moral. An axiomatic system for the propagation of uncertainty in directed acyclic networks. *International Journal of Approximate Reasoning*, **8**:253–280, 1993.
7. J.E. Cano, S. Moral y J.F. Verdegay-López. Combination of upper and lower probabilities. En: B.D. Ambrosio, Ph. Smets y P.P. Bonissone (eds.), *Proceedings of the 7th Conference on Uncertainty in Artificial Intelligence*, págs. 61–68. Morgan & Kaufmann, 1991.
8. J.E. Cano, S. Moral y J.F. Verdegay-López. Propagation of convex sets of probabilities in directed acyclic networks. En: B. Bouchon-Meunier et al. (eds.), *Uncertainty in Intelligent Systems*, págs. 15–26. Elsevier, 1993.

9. E. Castillo, J.M. Gutiérrez y A.S. Hadi. *Expert Systems and Probabilistic Network Models*. Springer Verlag, New-York, 1997.
10. A. Charnes y W.W. Cooper. Programming with linear fractional functionals. *Naval Research Logistics Quaterly*, **9**:181–186, 1962.
11. F. Cozman. Robustness analysis of bayesian networks with global neighborhoods. Technical Report CMU-RI-TR96-42, Carnegie Mellon University, 1996.
12. F. Cozman. Robustness analysis of bayesian networks with local convex sets of distributions. En: *Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence*. Morgan & Kaufmann, San Mateo, 1997.
13. L.M. de Campos, J.F. Huete y S. Moral. Probability intervals: a tool for uncertain reasoning. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, **2**:167–196, 1994.
14. L.M. de Campos y S. Moral. Independence concepts for convex sets of probabilities. En: Ph. Besnard y S. Hanks (eds.), *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*, págs. 108–115. Morgan & Kaufmann, 1995.
15. H. Edelsbrunner. *Algorithms in Combinatorial Geometry*. Springer Verlag, Berlin, 1987.
16. K.W. Fertig y J.S. Breese. Interval influence diagrams. En: M. Henrion, R.D. Shacter, L.N. Kanal y J.F. Lemmer (eds.), *Uncertainty in Artificial Intelligence, 5*, págs. 149–161. North-Holland, Amsterdam, 1990.
17. T. Hailperin. *Boole's Logic and Probability*. Studies in Logic and the Foundations of Mathematics 85. North-Holland, Amsterdam, 1976.
18. P. Hansen y B. Jaumard. Algorithms for the maximum satisfiability problem. *Computing*, **44**:279–303, 1990.
19. P. Hansen y B. Jaumard. Probabilistic satisfiability. Por aparecer en J. Kohlas y S. Moral, eds., *Algorithms for Uncertain and Defeasible Reasoning*, 1998.
20. P. Hansen, B. Jaumard, G.-B. Douanya Nguetse y M. Poggi de Aragão. Models and algorithms for probabilistic and Bayesian logic. En: *Proceedings of the 14th IJCAI Conference (IJCAI' 95) Vol. 2*, págs. 1862–1868, 1995.
21. B. Jaumard, P. Hansen y M. Poggi de Aragão. Column generation methods for probabilistic logic. *ORSA Journal of Computing*, **3**:135–147, 1991.
22. F.V. Jensen. *An Introduction to Bayesian Networks*. University College London Press, London, 1996.
23. F.V. Jensen, S.L. Lauritzen y K.G. Olesen. Bayesian updating in causal probabilistic networks by local computation. *Computational Statistics Quaterly*, **4**:269–282, 1990.
24. H.H. Karwan, V. Lofti, J. Telgen y S. Zionts. *Redundancy in Mathematical Programming: a State-of-the-Art Survey*. Lecture Notes in Economics and Mathematical Systems N. 206. Springer Verlag, 1983.
25. U. Kjærulff. Optimal decomposition of probabilistic networks by simulated annealing. *Statistics and Computing*, **2**:7–17, 1992.
26. S.L. Lauritzen y F.V. Jensen. Local computation with valuations from a commutative semigroup. *Annals of Mathematics and Artificial Intelligence*, **21**:51–69, 1997.
27. S.L. Lauritzen y D.J. Spiegelhalter. Local computation with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society, Ser. B*, **50**:157–224, 1988.

28. D.V. Lindley. Scoring rules and the inevitability of probability (with discussion). *International Statistical Review*, **50**:1–26, 1982.
29. T.H. Matheiss y D.S. Rubin. A survey and comparison of methods for finding all vertices of convex polyedral sets. *Mathematics of Operational Research*, **5**:167–185, 1980.
30. S. Moral. Algoritmos for imprecise probabilities. Por aparecer en: J. Kohlas y S. Moral, eds., *Handbook on Algorithms for Uncertain and Defeasible Reasoning*, 1998.
31. S. Moral y L.M. de Campos. Updating uncertain information. En: B. Bouchon-Meunier, R.R. Yager y L.A. Zadeh (eds.), *Uncertainty in Knowledge Bases*, págs. 58–67. Springer Verlag, Berlin, 1991.
32. N.J. Nilsson. Probabilistic logic. *Artificial Intelligence*, **28**:71–87, 1986.
33. N.J. Nilsson. Probabilistic logic revisited. *Artificial Intelligence*, **59**:39–42, 1993.
34. G. Paass. Probabilistic logic. En: Ph. Smets, A. Mamdani, D. Dubois y H. Prade (eds.), *Non-Standard Logics for Automated Reasoning*, págs. 213–251. Academic Press, London, 1988.
35. J. Pearl. *Probabilistic Reasoning with Intelligent Systems*. Morgan & Kaufman, San Mateo, 1988.
36. F.P. Preparata y M.I. Shamos. *Computational Geometry. An Introduction*. Springer Verlag, New York, 1985.
37. A.A. Salo. Tighter estimates for the posteriors of imprecise prior and conditional probabilities. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, **26**:820–825, 1996.
38. L.J. Savage. *The Foundations of Statistics*. Dover, 1972.
39. G. Shafer y P.P. Shenoy. Local computation in hypertrees. Working Paper N. 201, School of Business, University of Kansas, Lawrence, 1988.
40. P.P. Shenoy. A valuation-based language for expert systems. *International Journal of Approximate Reasoning*, **3**:383–411, 1989.
41. P.P. Shenoy y G. Shafer. Axioms for probability and belief-function propagation. En: Shachter et al. (eds.), *Uncertainty in Artificial Intelligence, 4*, págs. 169–198. Elsevier, 1990.
42. W. Stirling y D. Morrel. Convex bayes decision theory. *IEEE Transactions on Systems, Man and Cybernetics*, **21**:163–183, 1991.
43. B. Tessem. Interval probability propagation. *International Journal of Approximate Reasoning*, **7**:95–120, 1992.
44. B. Tessen. *Interval Representation of Uncertainty in Artificial Intelligence*. Tesis doctoral, Department of Informatics, University of Bergen, Norway, 1989.
45. H. Thöne. *Precise Conclusions under Uncertainty and Incompleteness in Deductive Database Systems*. Tesis doctoral, Universität Tübingen, Germany, 1994.
46. H. Thöne, U. Güntzer y W. Kießling. Towards precision of probabilistic bounds propagation. En: *Proceedings of the 8th Conference on Uncertainty in Artificial Intelligence*, págs. 315–322, 1992.
47. L.C. van der Gaag. Computing probability intervals under independence constraints. En: P.P. Bonissone, M. Henrion, L.N. Kanal y J.F. Lemmer (eds.), *Uncertainty in Artificial Intelligence, 6*, págs. 457–466. North-Holland, Amsterdam, 1991.

48. J.F. Verdegay-López. *Representaicón y Combinación de la Información con Incertidumbre mediante Convexos de Probabilidades.* Tesis doctoral, Universidad de Granada, 1997.
49. G. Verma y J. Pearl. Indentifying independence in bayesian networks. *Networks*, **20**:507–534, 1990.
50. P. Walley. *Statistical Reasoning with Imprecise Probabilities.* Chapman and Hall, London, 1991.
51. P. Walley. Measures of uncertainty in expert systems. *Artificial Intelligence*, **83**:1–58, 1996.
52. N. Wilson y S. Moral. A logical view of probability. En: A. Cohn (ed.), *Proceedings of the Eleventh European Conference on Artificial Intelligence (ECAI'94)*, págs. 386–390, London, 1994. Wiley.
53. L.A. Zadeh. A theory of approximate reasoning. En: J.E. Hayes y D. Mikulich (eds.), *Machine Intelligence, 9*, págs. 149–194. Elsevier, Amsterdam, 1979.

Aplicaciones de los Modelos Gráficos Probabilistas en Medicina

Francisco Javier Díez Vegas

Dpto. Inteligencia Artificial. Facultad de Ciencias
Universidad Nacional de Educación a Distancia
Avda. Senda del Rey, s/n. 28040 Madrid
correo-e: fjdiez@dia.uned.es

Resumen

La medicina tiene dos propiedades que hacen que los modelos gráficos probabilistas (MGP) encajen en ella como anillo al dedo: el conocimiento causal, correspondiente a los mecanismos patofisiológicos, y las numerosas fuentes de incertidumbre. Por ello, no es de extrañar que la mayor parte de los MGP, desde el principio hasta la actualidad, se hayan desarrollado en el campo de la medicina. En este artículo revisamos algunos de ellos y abordamos después aspectos generales, como la construcción de MGP en medicina y la explicación del razonamiento.

1 Introducción

1.1 Sistemas expertos en medicina: perspectiva histórica

El desarrollo de programas de diagnóstico basados en técnicas bayesianas comenzó en los años 60. Entre los sistemas de esa década destacan el de Warner y colaboradores [43] para el diagnóstico de cardiopatías congénitas, los de Gorry y Barnett [14,15] y el de de Dombal [8] y colaboradores para el diagnóstico del dolor abdominal agudo. Estos sistemas aplicaban el **método probabilístico clásico**, que consiste en seleccionar una variable D, que representa los n diagnósticos posibles d_i, y m variables H_j —binarias en general— correspondientes a los posibles hallazgos, que en medicina suelen ser los síntomas y signos; para que el problema sea tratable se introducen dos hipótesis: la primera, que los diagnósticos son *exclusivos y exhaustivos* y, la segunda, la *independencia condicional*, es decir, que los hallazgos son independientes entre sí para cada diagnóstico:

$$P(h_1, \ldots, h_m|d_i) = P(h_1|d_i) \cdot \ldots \cdot P(h_m|d_i), \qquad \forall d_i \tag{1}$$

Con estas hipótesis, el teorema de Bayes se reduce a la siguente expresión:

$$P(d_i|h_1,\ldots,h_m) = \frac{P(h_1|d_i)\cdot\ldots\cdot P(h_m|d_i)\cdot P(d_i)}{\sum_j P(h_1|d_j)\cdot\ldots\cdot P(h_m|d_j)\cdot P(d_j)} \qquad (2)$$

A partir de ella, resulta muy sencillo comparar la probabilidad de dos diagnósticos, con la posibilidad de incorporar secuencialmente nuevos hallazgos,

$$\frac{P(d_i|h_1,\ldots,h_m)}{P(d_j|h_1,\ldots,h_m)} = \frac{P(h_1|d_i)}{P(h_1|d_j)}\cdot\ldots\cdot\frac{P(h_n|d_i)}{P(h_n|d_j)}\cdot\frac{P(d_i)}{P(d_j)} \qquad (3)$$

Aunque este método sirvió de base a los sistemas de diagnóstico ya citados, con resultados satisfactorios para pequeños problemas, presenta serias deficiencias, pues ni en medicina los diagnósticos suelen ser exclusivos ni se da en general la independencia condicional, sino que los los hallazgos correspondientes a cada diagnóstico suelen estar correlacionados.

Como consecuencia de las dificultades que presentaba el método probabilístico clásico, los creadores del programa **MYCIN**[1] diseñaron en los años 70 un modelo que, en vez de buscar un fundamento matemático sólido, trataba de reproducir la forma en que el ser humano combina intuitivamente distintas fuentes de información. La idea básica consistía en asignar a cada regla "Si E entonces H" un **factor de certeza**, $CF(H,E)$. Aunque estos factores se definieron a partir de las probabilidades $P(H)$ y $P(H|E)$, en la práctica se obtenían directamente a partir de estimaciones de expertos humanos y se combinaban según reglas *ad hoc*, sin tener en cuenta los principios de la teoría de la probabilidad. A pesar del éxito que obtuvo MYCIN, cuyo índice de aciertos era comparable al de los mejores expertos humanos, pronto se comprobó —mediante razonamientos matemáticos— que contenía graves incosistencias, por lo que fue duramente criticado (cf. [27, sec. 1.2] y [10, sec. 2.4]).

Examinando los sistemas expertos de la década de los 80, observamos que la mayor parte de ellos se basaron en la **lógica difusa** y, en menor medida, en la teoría de la posibilidad, lo cual no es de extrañar, teniendo en cuenta que una parte considerable de los conceptos médicos son difusos: presión alta, dolor agudo, fatiga leve, tumor grande, síntoma evidente, prueba muy sensible, diagnóstico complejo, pronóstico grave, terapia arriesgada, cirujía mínimamente invasiva, alta mortalidad, etc., etc. Sin embargo, al contrario de lo que ocurre otras metodologías

[1] El proyecto MYCIN, desarrollado en la Universidad de Stanford, tenía como objetivo construir un sistema experto para el tratamiento de enfermedades infecciosas. MYCIN está considerado en la actualidad como el primer sistema experto y el "padre" de todos los sistemas basados en reglas.

de razonamiento aproximado, las distintas aplicaciones de la lógica difusa difieren notablemente entre sí, pues esta teoría carece de un fundamento normativo que indique cómo se debe aplicar en cada caso.

Pero fue también en la década de los 80 cuando se desarrollaron las **redes bayesianas** y los **diagramas de influencia**, desde su definición axiomática hasta el diseño de algoritmos eficientes para la computación de la evidencia, y pronto se vio que venían "como anillo al dedo" para el tratamiento de la incertidumbre en medicina. De hecho, como veremos en la sección 2, los primeros y la mayor parte de los sistemas de diagnóstico probabilístico se han construido en este campo, con un crecimiento exponencial en los últimos años.

Por otra parte, cada vez son más los argumentos tanto teóricos como empíricos[2] a favor del formalismo bayesiano, hasta el punto de que los propios creadores del programa MYCIN afirmaron en 1993 [7]:

> En la última década, la investigación sobre tratamiento de la incertidumbre en Inteligencia Artificial ha avanzado notablemente. Muchas de las restricciones que limitaban las opciones para tratar la incertidumbre en MYCIN (en particular, los argumentos en contra de adoptar un método bayesiano estadístico clásico) ya no son válidos. Por ejemplo, las redes bayesianas proporcionan ahora un método viable para construir grandes sistemas de diagnóstico sin utilizar las hipótesis burdas e inherentemente defectuosas de MYCIN sobre la independencia condicional y la modularidad del conocimiento.

1.2 Fuentes de incertidumbre en medicina

En prácticamente todas las aplicaciones de la inteligencia artificial surgen la incertidumbre y la imprecisión, fundamentalmente por tres motivos: deficiencias de la información, indeterminismo del mundo real y deficiencias de los modelos; los distintos métodos de razonamiento incierto han tratado de abordar al menos uno de estos tres tipos de incertidumbre. En medicina se pueden identificar fácilmente los siguientes:

- *Información incompleta.* En muchos casos la historia clínica completa no está disponible, y el paciente es incapaz de recordar todos los síntomas que ha experimentado y cómo se ha desarrollado la enfermedad. Además, en otras ocasiones, las limitaciones prácticas impiden contar con todos los medios que deberían estar disponibles, por lo que el médico debe realizar su diagnóstico con la información que posee, aunque sea muy limitada.

[2] Pueden encontrarse las referencias en [10].

- *Información errónea.* En cuanto a la información suministrada por el paciente, puede que éste describa incorrectamente sus síntomas e incluso que trate de mentir deliberadamente al médico. También es posible que el diagnóstico anterior, contenido en la historia clínica, haya sido erróneo. Y tampoco es extraño que las pruebas de laboratorio den falsos positivos y falsos negativos. Por estas razones, el médico debe mantener siempre una duda razonable frente toda la información disponible.
- *Información imprecisa.* Hay muchos datos en medicina que son difícilmente cuantificables. Tal es el caso, por ejemplo, de síntomas como el dolor o la fatiga. Incluso en un método tan técnico como la ecocardiografía, por ejemplo, hay muchas observaciones que en la práctica deben ser cuantificadas subjetivamente, como son el prolapso valvular o la aquinesia ventricular.
- *Mundo real no determinista.* A diferencia de las máquinas mecánicas o eléctricas, cuyo funcionamiento se rige por leyes deterministas, los profesionales de la medicina comprueban a diario que cada ser humano es un mundo diferente, en que las leyes generales no siempre resultan aplicables. Muchas veces las mismas causas producen efectos diferentes en distintas personas, sin que haya ninguna explicación aparente. Por ello, el diagnóstico médico debe tener siempre en cuenta la probabilidad y las excepciones.
- *Modelo incompleto.* Por un lado, hay muchos fenómenos médicos cuya causa aún se desconoce. Por otro, es frecuente la falta de acuerdo entre los expertos de un mismo campo. Finalmente, aunque toda esta información estuviera disponible, sería imposible, por motivos prácticos, incluirla en un sistema experto.
- *Modelo inexacto.* Por último, todo modelo que trate de cuantificar la incertidumbre, por cualquiera de los métodos que existen, necesita incluir un elevado número de parámetros; por ejemplo, en el caso de las redes bayesianas, necesitamos especificar todas las probabilidades a priori y condicionales. Sin embargo, rara vez está disponible toda esta información, por lo que debe ser estimada de forma subjetiva. Es deseable, por tanto, que nuestro modelo de razonamiento pueda tener en cuenta sus propias inexactitudes; por ejemplo, mediante la asignación de intervalos o de distribuciones de probabilidad para las probabilidades condicionales en el caso de los MGP.

De aquí se deducen dos razones recíprocas que explican por qué todos los modelos de razonamiento incierto se han centrado en alguna rama de la medicina: por un lado, la necesidad de abordar un problema médico concreto ha llevado en ocasiones a desarrollar un nuevo método, que luego se ha generalizado a distintos tipos de problemas y, por otro lado, la medicina constituye un excelente campo de pruebas para observar las cualidades y limitaciones de cualquier nuevo método que se proponga.

En los MGP se cumple claramente este principio: los primeros sistemas expertos basados en redes bayesianas tuvieron como objeto algún problema médico concreto y, de hecho, hoy en día es la medicina el campo donde se han desarrollado la mayor parte de los sistemas basados en MGP. Describimos los más importantes en la sección siguiente.

2 Ejemplos de MGP en medicina

2.1 Aplicaciones desarrolladas fuera de España

En esta sección nos vamos a centrar en los modelos normativos, es decir, en los que se ajustan a los principios de la teoría de la probabilidad y de la decisión, que, en la práctica, son aquéllos cuya base de conocimientos viene dada por una red bayesiana o por un diagrama de influencia. No vamos a describir aquí los sistemas expertos, como CASNET o el *Heart Disease Program*, de W. Long, que utilizan modelos probabilísticos aproximados. Tampoco vamos a hablar de los basados en el método probabilístico clásico, porque ya lo hicimos en la sección 1.1.

Siguiendo el orden de complejidad creciente —que no coincide con el cronológico— debemos mencionar el modelo de Schwartz, Baron y Clarke [33] para el diagnóstico de la apendicitis. Frente al método clásico tenía la ventaja de que, al introducir causas intermedias, salvaguardaba la independencia condicional de ciertos hallazgos correlacionados respecto del diagnóstico principal. Frente a las redes bayesianas, presentaba la limitación de que cada nodo sólo podía tener un padre y, en consecuencia, tampoco admitía bucles.

La primera red bayesiana médica fue construida por Cooper [5,6] en la Universidad de Stanford, como ejemplo para la aplicación del programa NESTOR. Entre los aspectos más avanzados de este sistema destacaban la posibilidad de definir las probabilidades condicionales mediante intervalos, y la capacidad de explicación, de la que hablaremos en la sección 4.3. La red que utilizó como ejemplo ilustrativo contenía cinco nodos: cáncer metastásico, elevación del calcio sérico, tumor cerebral, coma y jaquecas.

Otro de los primeros ejemplos de redes bayesianas médicas es la de Lauritzen y Spiegelhalter [23]; tiene 8 nodos y realiza el diagnóstico diferencial entre tuberculosis, bronquitis y cáncer de pulmón. Éstas son, con diferencia, las dos redes bayesianas más famosas, por haber sido utilizadas repetidamente para ilustrar muchos de los algoritmos que se han desarrollado desde entonces; obviamente, dos redes tan minúsculas no intentan resolver problemas reales, sino que sólo son útiles con fines ilustrativos.

En cuanto a las redes bayesianas con un conocimiento extenso, destinadas al diagnóstico clínico real, la primera y una de las más conocidas fue desarrollada

por un equipo de investigadores de la Universidad de Aalborg (Dinamarca) [2,26]; recibió el nombre de MUNIN y estaba destinada al diagnóstico de enfermedades musculares mediante electromiografía.[3] Los nodos estaban agrupados en tres niveles: enfermedades, estados patofisiológicos y hallazgos. También en este grupo se desarrolló un sistema que permitía modelar el metabolismo de los carbohidratos con el fin de ajustar la dosis de insulina en pacientes diabéticos [1].

Volviendo a la Universidad de Stanford, destaca por su importancia el sistema experto PATHFINDER, de David Heckerman [18], destinado al diagnóstico de enfermedades de los ganglios linfáticos. La principal aportación del trabajo de Heckerman es la creación de las *redes de semejanza* (*"similarity networks"*), que se caracterizan por la existencia de un nodo principal, que representa los posibles diagnósticos. La limitación principal de este modelo es suponer que el paciente padece una sola enfermedad, lo cual es una hipótesis razonable en el caso de los ganglios linfáticos, pero resultaría inverosímil en otros dominios, como la cardiología, en que las enfermedades suelen estar relacionadas; a cambio, facilita la construcción del modelo (porque se centra en el diagnóstico diferencial de cada par de enfermedades), hace más eficiente la computación de la probabilidad y permite explicar el razonamiento (cf. sec. 4.3). La empresa Intellipath, que comercializa actualmente el sistema PATHFINDER, ha vendido cientos de copias, que se usan a diario en numerosos hospitales.

En la misma universidad se construyó el sistema QMR-DT [35], que es una reformulación en forma de red bayesiana del sistema experto QMR, el cual a su vez, era la versión comercial de INTERNIST-1. Las dos limitaciones principales de QMR-DT son la representación todos los diagnósticos y hallazgos mediante variables binarias y la disposición de los nodos en dos niveles, sin permitir variables intermedias; estas dos simplificaciones impiden representar correctamente las relaciones de independencia, como explican muy bien sus propios creadores. Igualmente, el programa Iliad, un tutor para medicina interna construido inicialmente mediante un modelo probabilista aproximado, ha sido reformulado posteriormente en forma de red bayesiana, con lo que se ha logrado mejorar su rendimiento [25].

En Europa, el grupo más importante dedicado a las redes bayesianas médicas —junto con el de Aalborg— es el de la Universidad de Pavía, en Italia, donde se han desarrollado redes bayesianas y diagramas de influencia para optimización de terapia en anemia urémica, monitorización, leucemia infantil, hemodiálisis, diabetes, SIDA, tratamiento de niños con transplantes de médula ósea, nefritis,

[3] Dentro de este mismo proyecto se desarrolló HUGIN, una herramienta destinada a la construcción de redes bayesianas, que es comercializada actualmente por la empresa del mismo nombre.

linfoma gástrico primario, trombosis idiopática en venas profundas, esplenotomía, etc.[4]

El mexicano Luis Enrique Súcar [39] ha desarrollado un sistema de visión artificial para colonoscopia; además de ser —según nuestros conocimientos— la única red bayesiana para visión que resuelve un problema real, tiene el mérito de que el sistema es capaz de obtener las probabilidades condicionales e incluso refinar la estructura de la red a partir de los datos disponibles.

Entre las redes bayesianas más utilizadas se encuentra el programa *Microsoft Pregnancy and Child Care*, que ofrece sus consejos en la red de Microsoft;[5] la base de conocimientos fue desarrollada y comprobada por Knowledge Industries, empresa que también ha construido redes bayesianas para dermatología, alteraciones del sueño, cuidado de traumatismos, chequeo de mano y muñeca y atención sanitaria a domicilio.[6]

Otros sistemas basados en MGP son: ALARM [3], para la monitorización de pacientes durante la anestesia; VP-net [32], para monitorización e interpretación de datos en la UCI; THOMAS [24], para interpretar los resultados de experimentos clínicos aleatorizados publicados; ABDO [29], para el diagnóstico del dolor abdominal agudo; el de Haddawy y colaboradores [17], para la vesícula biliar; CPCS-BN [28], para enfermedades heptobiliares; MammoNet [21], para enfermedades de mama; etc.[7]

Una mención a parte merece el programa BANTER, de Haddawy, Jacobson y Kahn [16], cuyo objetivo no es el diagnóstico ni la toma de decisiones, sino la enseñanza de la medicina a partir de cualquier red bayesiana o diagrama de influencia cuyos nodos puedan clasificarse en hipótesis, observaciones y métodos diagnósticos. Además de calcular la probabilidad a posteriori de cada hipótesis, BANTER es capaz de seleccionar el mejor método diagnóstico para confirmar o descartar cualquier hipótesis, de examinar al usuario sobre la selección de métodos diagnósticos, y de explicar su razonamiento (utilizando el método INSITE, de Suermondt, del que hablaremos en la sec. 4.3).

---

[4] Las referencias pueden encontrarse en `http://ipvaimed9.unipv.it/lab/publications.html`.

[5] Microsoft está desarrollando también una red bayesiana para cardiología (comunicación personal de Eric Horvitz y Jack Breese), aunque aún no conocemos referencias escritas.

[6] Las referencias pueden encontrarse a partir de `http://www.auai.org/auai-companies.html`.

[7] El código completo de algunas de las redes mencionadas en esta sección, como PATHFINDER, MUNIN y ALARM, puede encontrarse en `http://www-nt.cs.berkeley.edu/home/nir/public_html/Repository/`.

2.2 Aplicaciones desarrolladas en España

DIAVAL [10,12] es un sistema experto para el diagnóstico de enfermedades cardíacas, que considera principalmente la información ecocardiográfica, aunque teniendo en cuenta también otras fuentes de información: datos personales, síntomas y signos, hallazgos electrocardiográficos, etc. Fue desarrollado por Francisco J. Díez, de la Universidad Nacional de Educación a Distancia, en colaboración con el Hospital de la Princesa, de Madrid. El nombre se debe a que inicialmente estaba orientado al DIAgnóstico de VALvulopatías. En la sección 4 hablaremos de sus aportaciones en cuanto al paso de una red bayesiana a un sistema experto completo.

DIABNET es un sistema de planificación de terapias en diabetes gestacional, construido por Elena Hernando [19] de la Facultad de Telecomunicación de la Universidad Politécnica de Madrid, en colaboración con el Servicio de Endocrinología y Nutrición del Hospital San Pau de Barcelona. Su base de conocimiento está constituida por una red bayesiana que modeliza cualitativamente el metabolismo de la glucosa. Dado que está orientado a la monitorización y al seguimiento de una enfermedad, el empleo de redes dinámicas desempeña un papel esencial.

IctNeo [31] es un sistema destinado al tratamiento de la ictericia neonatal, que está siendo desarrollado por varios investigadores del Departamento de Inteligencia Artificial de la Universidad Politécnica de Madrid, en colaboración con el Hospital Gregorio Marañon de Madrid. Además de las dificultades inherentes a la construcción del diagrama de influencia (estructura, probabilidades condicionales y funciones de utilidad), el tamaño relativamente elevado de la red —59 nodos en la última versión, con numerosos bucles— dificulta el cálculo que llevará a determinar la política terapéutica.

En la Universidad del País Vasco, Basilio Sierra y Pedro Larrañaga [36] han desarrollado un método para la construcción de redes bayesianas a partir de bases de datos mediante algoritmos genéticos, y lo han aplicado al pronóstico en dermatología, concretamente a la predicción de supervivencia de pacientes con melanoma maligno (véase el capítulo de P. Larrañaga en este mismo libro).

Por último, mencionamos dos proyectos en curso: el de Carmen Lacave y Juan Giralt, de la Universidad de Castilla-La Mancha, para el diagnóstico diferencial de enfermedades infecciosas en pediatría, y el de Enrique Nell, para el diagnóstico de enfermedades del miocardio. Ambos se encuentran aún en sus comienzos.

3 Construcción de MGP en medicina

Hay básicamente dos métodos para la construcción de redes bayesianas:

- *A partir de una base de datos*, aplicando alguno de los métodos de aprendizaje de redes descritos en capítulos anteriores de este libro.
- *Con la ayuda de expertos humanos* (médicos de la especidad, en nuestro caso), mediante una serie de sesiones en que el constructor del modelo interroga a los expertos y, con el conocimiento obtenido, va añadiendo nodos-variables, enlaces y probabilidades condicionales a la red.

Describimos cada uno de ellos en las dos secciones siguientes.

3.1 Construcción a partir de bases de datos

La forma más rápida de construir red bayesiana para medicina consiste en tomar una base de datos que contenga un número suficientemente grande de casos (de pacientes, generalmente) y aplicar algún algoritmo de aprendizaje. Como estos algoritmos ya se han descrito en capítulos anteriores de este libro, nos vamos a limitar a discutir aquí su aplicación en medicina.

En primer lugar, comprobamos que la mayoría de estos algoritmos suponen que tenemos una base de datos en que *el valor de cada variable está determinado con certeza* para cada caso. Sin embargo, la mayor parte de las bases de datos médicas sólo recogen unos pocos de los datos observados, junto con el diagnóstico final. En cambio, la construcción de una red bayesiana requiere especificar numerosas variables intermedias, para que tengan validez las hipótesis de independencia.

Es cierto que existen algoritmos capaces de encontrar variables ocultas examinando las correlaciones entre los datos. Aun así, sigue habiendo dos inconvenientes. El primero es que hace falta una cantidad muy grande de datos para que los resultados sean fiables; el problema se agrava cuando aumenta la proporción de variables ocultas frente a variables registradas. Y el segundo inconveniente es que puede ocurrir que las variables "descubiertas" no correspondan a ningún concepto médico, con lo que la validez del modelo resultaría más que cuestionable.

Esto explica por qué, a pesar de los numerosos trabajos sobre aprendizaje automático de redes bayesianas, ninguna de las aplicaciones mencionadas en la sec. 2 —salvo las construidas con fines académicos— se hayan construido mediante estos algoritmos.

Finalmente, señalemos que tales algoritmos podrían ser útiles, a lo sumo, para construir redes bayesianas, pero no para generar diagramas de influencia, pues son incapaces de extraer de las bases de datos nodos-decisión y nodos-utilidad.

3.2 Construcción con la ayuda de expertos humanos

La construcción de un MGP puede dividirse en dos fases. La primera de ellas consiste en recopilar la *información cualitativa*, es decir, en identificar las anomalías y los datos relevantes, y formar una red causal con las relaciones entre

ellos. La segunda fase se ocupa de recoger la *información cuantitativa*: las probabilidades a priori y las probabilidades condicionales. Veamos cada una de ellas por separado.

Obtención de la información cualitativa. Todo MGP implica un número —generalmente elevado— de relaciones de independencia condicional, que, en principio, habría que justicar mediante análisis estadísticos; sin embargo, la falta de datos empíricos impide casi siempre realizar tal comprobación (los trabajos de Luis Enrique Súcar [40,39] constituyen una notable excepción). La alternativa más utilizada consiste en aplicar conocimiento sobre los **mecanismos causales**, aunque rara vez los constructores de los modelos se cuestionan las hipótesis que están introduciendo (aquí, las excepciones son el trabajo de Shwe y colaboradores [35] y el de Díez [11], que resumimos a continuación). Por ello, debemos interrogar a los expertos sobre los mecanismos causales que, a su juicio, intervienen en nuestro problema, y a partir de ellos intentar justificar las propiedades de independencia mediante la aplicación de las reglas siguientes (véase la fig. 1):

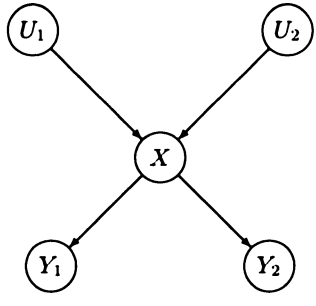


Figura 1. Independencia condicional para un nodo X con dos padres y dos hijos.

Independencia a priori. Cuando hay dos variables U_1 y U_2 tales que *(1)* no existe correlación conocida entre una y otra, *(2)* no hay ningún mecanismo causal por el que U_1 pueda producir U_2, ni viceversa, y *(3)* no hay ninguna causa común de ambas, entonces podemos suponer que son a priori independientes. Por ejemplo, entre el sexo y el país de origen podemos suponer que hay independencia a priori. Cuando la correlación es pequeña (por ejemplo, entre sexo y edad), podemos considerar la posibilidad de despreciarla y tratar

las variables como independientes a priori, con el fin de no complicar excesivamente la propagación de evidencia.

Independencia condicional entre varios efectos de una causa. Cuando *(1)* X es una causa común de Y_1 e Y_2, *(2)* el mecanismo causal por el que X produce Y_1 no interactúa con el mecanismo $X \rightarrow Y_2$, *(3)* no hay ninguna relación causal conocida $X \rightarrow Y_1$ ni $X \rightarrow Y_2$, y *(4)* no hay ninguna otra causa común de Y_1 e Y_2, entonces podemos suponer que ambas son condicionalmente independientes dado X. Por ejemplo, entre un síntoma Y_1 y una prueba de laboratorio Y_2 indicativas de una misma enfermedad X, podemos suponer casi siempre que hay independencia condicional.

Independencia condicional entre un efecto y sus "abuelos". Cuando *(1)* las causas de X son $U_1, \ldots, U_n$, *(2)* el mecanismo $X \rightarrow Y$ es independiente de cómo se ha producido X, y *(3)* no hay ningún otro mecanismo conocido $U_i \rightarrow Y$, entonces podemos suponer que U_i e Y son condicionalmente independientes dado X. Por ejemplo, la zona de origen (U_1) y el grupo sanguíneo (U_1) son dos factores de riesgo para el paludismo (X); en la práctica, podemos suponer que la probabilidad de que el test de la gota gruesa (Y) —la prueba más habitual para detectar el paludismo— dé positivo es independiente de la zona de origen y del grupo sanguíneo una vez que conozcamos con certeza si una persona padece la enfermedad o no.

Desgraciadamente, hay muchos casos en que no se conocen los mecanismos causales que llevan a una determinada alteración. Por ejemplo, un libro de cardiología puede afirmar que los principales factores riesgo del infarto agudo miocardio (IAM) son la edad, el ser varón, el ser de raza blanca, los antecedentes familiares, el tabaquismo, la obesidad, el estrés, la ingesta elevada de sodio, la hipercolesterolemia, la diabetes y la hipertensión arterial. Obviamente, estos ocho factores no son independientes entre sí, pero resulta imposible conocer en qué medida cada uno de ellos afecta a los demás, pues, que nosotros sepamos, ninguno de los numerosos estudios epidemilógicos que se han llevado a cabo sobre el IAM ha intentado estudiar la dependencia e independencia condicional entre sus factores de riesgo.

Aplicación de la puerta OR. Otro de los puntos importantes, posterior a la determinación de la estructura de la red y previo a la obtención de la información cuantitativa, consiste en decidir cuáles de las familias de la red pueden ser modeladas mediante la puerta OR. En efecto, la puerta OR requiere *muchos menos parámetros* que el modelo general, lo cual conlleva ventajas en cuanto al almacenamiento de la información, en cuanto a la *propagación de la evidencia* y, sobre todo, en cuanto a la *obtención del conocimiento*, no sólo porque necesita muchos

menos parámetros, sino porque los parámetros que intervienen son mucho más significativos para un médico y más fáciles de estimar que los elementos de una tabla de probabilidad; por ejemplo, tienen mucho más sentido las preguntas como "¿cuál es la probabilidad de que U_i produzca X?" que "¿cuál es la probabilidad de $+x$ cuando $+u_1$, $\neg u_2$, $+u_3$ y $+u_4$?", con la complicación adicional de que nuestro especialista probablemente nunca ha visto un enfermo que padeciera a la vez U_1, U_3 y U_4, con lo que le resultaría absolutamente imposible estimar dicha probabilidad.

Por último, la puerta OR presenta ventajas en cuanto a la *explicación del razonamiento*: concretamente, en presencia de un síntoma o signo S, la confirmación de una enfermedad causante de S resta credibilidad a las otras causas de S (este fenómeno se denomina en inglés "*explaining away*"); y viceversa, la exclusión de todas las causas de S excepto una, lleva a considerar ésta como el diagnóstico más probable. Este tipo de razonamiento, que en medicina se denomina *diagnóstico diferencial*, es específico de la puerta OR, y no se da en caso general.

Por tanto, es muy deseable aplicar la puerta OR siempre que sea posible, pero para ello han de darse ciertas condiciones:

1. tanto el nodo hijo como sus padres han de ser variables que indiquen el grado grado de presencia de una anomalía; es decir, el rango de valores debe ser "ausente/presente" o "ausente/leve/moderada/severa" o un conjunto similar [9]; esto impide la aplicación de la puerta OR cuando los padres representan otro tipo de variables, como la edad, el sexo o la raza;
2. cada uno de los padres representa una causa que puede producir el efecto (el nodo hijo) en ausencia de las demás causas;
3. no hay sinergia entre las causas; es decir, el mecanismo por el que U produce X es independiente de los mecanismos de las demás causas de X (obsérvese que estamos hablando nuevamente de **causalidad**).

Por tanto, las condiciones 2 y 3 impiden la aplicación de la puerta OR cuando los padres representan factores de riesgo, tales como el tabaquismo, la obesidad, la hipercolesterolemia, etc., ninguno de los cuales es capaz de *producir* (causar) la enfermedad (el infarto agudo de miocardio, volviendo al ejemplo anterior) en ausencia de los demás.

Obtención de la información cuantitativa. Si ya la adquisición del conocimiento cualitativo puede resultar complicada, mucho más lo es la obtención de los datos numéricos. Por más que revisemos la bibliografía médica, difícimente vamos a encontrar más que una pequeña parte de la información que necesitamos, pues las descripciones que aparecen en la literatura son casi siempre cualitativas. Veamos como ejemplo la siguiente afirmación extraída de un libro especializado:

> El tumor primario *más común* en el corazón *adulto* es el mixoma y *el 75%* de ellos se localiza en la aurícula izquierda, *habitualmente* en mujeres. [Cursiva añadida.]

En esta breve cita, aparecen dos términos difusos, *adulto* y *habitualmente*. Esto nos plantea varios interrogantes: ¿Desde qué edad se considera a una persona como adulta? ¿Distingue entre *adultos* y *ancianos* o los engloba a todos en el mismo grupo? ¿Qué frecuencia debemos entender por *habitualmente*? Hay estudios psicológicos que pueden ofrecer una cierta ayuda a la hora de convertir las expresiones cualitativas en probabilidades numéricas, pero las variaciones en las asignaciones son tan grandes que resultan de poca utilidad.

El único número concreto que aparece, "*el 75%*", —no sabemos si se trata de un resultado experimental o de una estimación aproximada— tampoco es de gran ayuda, porque no indica la probabilidad de que haya un mixoma en la aurícula izquierda, sino de que, habiendo mixoma, se localice en la aurícula izquierda, lo cual no es un dato que se pueda introducir directamente en la red.

Con este sencillo ejemplo pretendemos mostrar por qué se hace necesario casi siempre recurrir a estimaciones subjetivas de expertos humanos, a pesar de que la labor es tediosa y compleja (cf. [37],[22, cap. 4]).

3.3 Funciones de utilidad en medicina

La obtención de funciones de utilidad en medicina es igualmente difícil. Algunos de los primeros trabajos utilizaban escalas subjetivas, graduadas de 0 a 100; este método fue criticado porque las unidades de medida eran arbitrarias, es decir, sin ningún significado médico objetivo, y variaban de una aplicación a otra dependiendo de cuáles fueran los extremos escogidos.

Por eso se desarrollaron otros métodos basados en datos objetivos, como la supervivencia a corto plazo (expresada en porcentajes) o la no morbilidad (el número de casos en que se curaba el paciente). La variable que con más frecuencia se ha empleado para determinar la utilidad es la esperanza de vida del paciente, medida normalmente en años, y con este criterio se han construido numerosos modelos y programas de ordenador desde la década de los 70. Sin embargo, no sólo es importante la duración de la vida, sino también la calidad, y por eso el criterio más adecuado en general es la **esperanza de vida en salud** (en inglés, "*quality-adjusted life-expectancy*"), que se define como el tiempo el tiempo que va a vivir el paciente multiplicado por la calidad de vida que va a tener; más exactamente, teniendo en cuenta que la calidad de vida varía con el tiempo, $c(t)$, la **vida en salud** para un paciente se define como

$$VS = \int c(t) \cdot dt \tag{4}$$

La unidad de medida se denomina en inglés "*quality-adjusted life-year*"; nosotros hemos propuesto como traducción el término "**año-salud**" [22, pág. 66].[8]

Sin embargo, hay casos en que las funciones de utilidad universales no tienen sentido. Por ejemplo, ante la posibilidad de un embarazo, unas parejas manifiestan más interés que otras por tener un niño (en unos casos la utilidad será positiva y en otros negativa), el riesgo que cada mujer está dispuesta a asumir es distinto, cada pareja valora de forma diferente las posibles malformaciones congénitas del futuro niño, etc. Por eso se han desarrollado métodos que intentan conocer y medir las preferencias de cada paciente. (Una discusión más extensa sobre la obtención de funciones de utilidad en medicina puede encontrarse en [22, cap. 3]).

Para concluir, comentamos que, cuando se trata de decidir si el coste económico de una terapia o un plan de actuación compensa las ventajas obtenidas, éstas pueden medirse de tres modos:

Análisis coste-efectividad: Mide la utilidad en alguna *unidad médica*, como el número de vidas salvadas o el porcentaje de hipertensos controlados.

Análisis coste-beneficio: Asigna un valor económico a los estados resultantes, incluida la vida o la muerte del paciente, con el fin de medir la utilidad en alguna *unidad monetaria*.

Análisis coste-utilidad: Valora la *calidad de vida* del paciente, generalmente teniendo en cuenta sus *preferencias*, como acabamos de explicar.

Naturalmente, los diagramas de influencia son capaces de englobar estos tres tipos de análisis dentro de un mismo formalismo, pues una vez conocida la función de utilidad el tratamiento matemático es idéntico.

4 De un MGP a un sistema experto

En la sección anterior hemos hablado sobre la construcción de modelos gráficos probabilistas. Sin embargo, tales modelos no pueden considerarse por sí mismos como sistemas expertos, pues para ello necesitan, como mínimo,

[8] Un *año-luz* es la distancia que la luz recorre en un año; análogamente, un *año-salud* es la "cantidad de salud" que una persona sana disfruta a lo largo de un año, o bien la salud que una persona con la calidad de vida reducida a la mitad disfruta en dos años, etc.

- un *interfaz gráfico amigable*, de modo que el usuario pueda introducir la información de algún modo que le resulte familiar, sin tener que buscar en la red la variable correspondiente a cada hallazgo;
- un *generador de informes* que selcccione las conclusiones más relevantes, pues el mostrar en una ventana la probabilidad a posteriori de cada variable es claramente insuficiente;
- un método de *explicación del razonamiento*, que justifique el diagnóstico y las recomendaciones ofrecidas por el sistema, de modo que el usuario pueda aceptarlas o rechazarlas según su criterio.

Sin estas tres características, es seguro que incluso la red bayesiana que calcule las probabilidades más exactas o el diagrama de influencia que encuentre las mejores decisiones en cada caso, nunca llegarán a aplicarse en la práctica. Por eso vamos a describir a continuación las distintas soluciones que se han propuesto para cada uno de estos tres puntos.

4.1 Interfaz de usuario

Una limitación de los primeros sistemas expertos —basados en reglas— es que dirigían la consulta mediante una serie de preguntas, sin que el usuario pudiera tomar la iniciativa sobre la información que deseaba introducir. En cambio, en la mayor parte de los modelos gráficos probabilistas, el problema es más bien el contrario, pues suelen limitarse a ofrecer una pantalla en la que se muestra la red, de modo que el usuario debe señalar los nodos a los que desea asignar un valor en función de los hallazgos disponibles (el ejemplo más conocido es MUNIN [2]).

Una paso adelante lo constituye PATHFINDER, que agrupa los posibles hallazgos en categorías, lo cual facilita al usuario localizar el que desea introducir, e incluso sugiere cuál es el síntoma o signo que debe buscar el usuario en función del coste-efectividad [18, figs. 1.2 a 1.6].

El sistema experto DIAVAL, en cambio, implementa lo que en la terminología de los sistemas expertos se conoce como *interfaz de iniciativa mixta*, pues, por un lado, ofrece una serie de pantallas en un orden determinado, con lo que se facilita la recogida sistemática de los hallazgos ecocardiográficos, mientras que, por otro, ofrece una serie de menús que permiten acceder directamente a la ventana deseada.

4.2 Generación de diagnósticos e informes

Algunos de los sistemas de diagnóstico probabilistas se limitan a ofrecer sus conclusiones en una pantalla donde muestran la probabilidad a posteriori para cada variable [2]. Otros, como PATHFINDER [18], presentan una lista de las

variables correspondientes al diagnóstico, ordenadas de mayor a menor probabilidad.

DIAVAL [10, cap. 13], en cambio, aborda el problema estableciendo dos *umbrales*, de *certeza* y de *relevancia*, y asignando a cada nodo dos factores: la *relevancia para diagnóstico positivo* (RDP) y la *relevancia para diagnóstico negativo* (RDN), en una escala subjetiva de 0 a 10;[9] naturalmente, las enfermedades tienen factores de relevancia más altos que los estados patofisiológicos y las alteraciones intermedias. Por otro lado, los nodos están agrupados en 21 *capítulos*, cada uno de los cuales corresponde a una parte del corazón (una válvula, el pericardio, etc.) o a un tipo de información (antecedentes familiares, factores de riesgo, etc.).

Tras propagar la evidencia, el programa selecciona dentro de cada capítulo aquellos nodos que superan tanto el umbral de certeza como el de relevancia; bajando estos umbrales, el usuario puede conseguir que se muestren diagnósticos menos probables o menos importantes, respectivamente. Esto permite presentar las conclusiones de forma ordenada, primero en una pantalla, donde el médico puede introducir las correcciones que estime oportunas, y después en un informe escrito que pasa a un procesador de texto y a una impresora.

4.3 Explicación del razonamiento

Hemos mencionado ya que, para que una red bayesiana pueda considerarse como verdadero sistema experto, hace falta que pueda explicar su proceso de razonamiento.[10] El objetivo principal de la capacidad de explicación es **justificar los resultados**, de modo que el usuario pueda decidir si las conclusiones que ofrece el programa son correctas o no; de hecho, es famoso el estudio de Teach y Shortliffe [42] en que se demuestra que los médicos son muy reticentes a aceptar el consejo de un ordenador si no pueden confirmar su fundamento.[11] Además, la

[9] Diagnósticos positivos son, por ejemplo, la estenosis mitral leve, moderada y severa. La ausencia de estenosis es un diagnóstico negativo.

[10] En los sistemas de diágnostico se habla a veces de *explicación* como un "conjunto de hipótesis capaz de justificar las anomalías observadas"; en cambio, aquí nos referimos a un concepto diferente: la *explicación del razonamiento* trata de mostrar cómo y por qué el sistema ha llegado a sus conclusiones.

[11] Conviene destacar en relación con este punto la evolución que se ha producido en las dos últimas décadas en la forma de entender la inteligencia artificial. Anteriormente, el objetivo principal era construir programas que *igualaran o superaran* la capacidad de los seres humanos; por eso, en la evaluación de los sistemas expertos médicos se trataba de demostrar que diagnosticaban igual o mejor que los propios especialistas. Hoy en día, la inteligencia artificial trata ante todo de construir sistemas que *colaboren* de forma simbiótica con el ser humano, aprovechando lo mejor de lo natural y de lo artificial; en esta línea, la evaluación más positiva de un sistema experto será aquélla

capacidad de explicación es sumamente útil durante la construcción del modelo para **depurar los errores** que de otro modo serían difíciles de detectar y corregir. Y una tercera ventaja de la capacidad de explicación es la **enseñanza**; por ejemplo, en la sección 2 hemos mencionado el sistema BANTER, que permite utilizar redes bayesianas para instruir a estudiantes de medicina. En esta sección vamos a describir algunos de los métodos de explicación propuestos para sistemas probabilistas.

Como hemos dicho en la sección 1.1, los primeros sistemas bayesianos de diagnóstico se basaban en el método probabilístico clásico. Expresando la ecuación (3) en forma logarítmica,

$$\log \frac{P(d_i|h_1,\ldots,h_m)}{P(d_j|h_1,\ldots,h_m)} = \log \frac{P(d_i)}{P(d_j)} + \sum_{k=1}^{n} \frac{P(h_k|d_i)}{P(h_k|d_j)} \tag{5}$$

se observa que el primer término del segundo miembro es independiente de la evidencia encontrada, de modo que son los términos del sumatorio los que aumentan o disminuyen la probabilidad de d_i frente a d_j; resulta así muy sencillo averiguar cuáles son los datos que más han contribuido a favor o en contra de una determinada hipótesis. Éste es básicamente el método utilizado en el sistema MEDAS [4], en GLADYS [38] y en el sistema para la clasificación de apoplejías de Reggia y Perricone [30] y es, en esencia, el mismo que usa Heckerman en PATHFINDER [18, sec. 6.2.1]

En la misma línea, y dentro ya de las redes bayesianas, Sember y Zuckerman [34] abordan un problema diferente: cómo explicar los cambios en la probabilidad a posteriori de un nodo en un poliárbol mediante el análisis de los mensajes π y λ.

Otro trabajo interesante es el de Jensen y colaboradores [20] sobre la detección de conflictos en redes bayesianas. Para un conjunto de hallazgos $\bar{S}$, definen una *medida del conflicto* como

$$\text{conf}(s_1,\ldots,s_n) = \log_2 \frac{P(s_1)\cdot\ldots\cdot P(s_n)}{P(s_1,\ldots,s_n)} \tag{6}$$

Esta expresión refleja la idea de que la medida del conflicto debe aumentar cuando la probabilidad de que los hallazgos se presenten de forma conjunta es mucho menor que la probabilidad de aparezcan independientemente. Como el cálculo se realiza a partir de medidas locales, es posible rastrear el origen del conflicto,

---

que demuestre que el médico ayudado por el sistema experto diagnostica más rápido y mejor que el médico solo. Y para que pueda darse esta simbiosis hombre-máquina es fundamental la explicación del razonamiento.

aunque con el inconveniente de que estas medidas locales no corresponden a la verdadera estructura de la red bayesiana, sino a la del árbol de cliques.

En una línea muy diferente, el sistema experto DIAVAL [10, cap. 8] ofrece un método de explicación que se basa en la distinción de seis tipos de enlaces, dos para el modelo general (influencia y parámetro) y cuatro para la puerta OR (causalidad, tipo, manifestación y observación), y ofrece varias opciones de explicación para cada nodo, enlace o dato cualitativo: probabilidad a priori, probabilidad a posteriori, causas, efectos, sensibilidad, especificidad, intervalos, fórmula con que se calcula, etc. En particular, la utilización generalizada de la puerta OR permite explicar en la mayor parte de los casos cuál es la causa más probable de cada anomalía. Por último, el interfaz gráfico permite navegar por la red observando los diferentes nodos y enlaces.

Los métodos descritos hasta ahora intentan explicar en qué medida la distribución de probabilidad de un nodo se ve afectada por las probabilidades de sus vecinos; es lo que se conoce como *nivel micro* [34]. En cambio, el *nivel macro* rastrea las principales líneas de razonamiento a lo largo de la red.

Por ejemplo, el sistema NESTOR, de Cooper [5,6] (véase la sec. 2), ofrecía dos posibilidades de macro-explicación. Una de ellas consistía en mostrar, en forma de texto, las cadenas de enlaces que relacionaban una hipótesis con los hallazgos. La otra mostraba numérica y gráficamente cómo se iban modificando las probabilidades de dos hipótesis seleccionadas a medida que se introducían la evidencia.

En la tesis de Suermondt [41], dirigida por Cooper, se presenta la metodología INSITE, que consiste en medir la influencia de los hallazgos sobre cada nodo de la red, con la posibilidad de examinar con más detalle ciertas cadenas de nodos. Como medida del impacto existen varias funciones posibles; por defecto, el sistema toma la entropía cruzada (*cross-entropy*).

Hay otros dos métodos, desarrollados por Druzdzel y Henrion. El primero de ellos [13] se basa en el concepto de *escenario*, definido como la asignación de valores para las variables (discretas) de un subconjunto. El algoritmo de explicación selecciona las variables relevantes y halla los escenarios más probables; después ordena las variables y las une mediante frases, generando así una historia causal de cómo se han producido los hechos [13, sec. 8.3.3].

El otro método que proponen Druzdzel y Henrion [13], se basa en las *redes cualitativas* de Wellman, que, en vez de utilizar información numérica, sólo consideran si la influencia de un nodo sobre otro es positiva (si hace aumentar la probabilidad de los valores más altos), negativa, nula o ambigua (desconocida); la puerta OR desempeña un importante papel en este modelo [44]. Examinando el impacto de la propagación de evidencia entre nodos vecinos, se puede generar una cadena de enlaces entre las dos variables de interés, en que cada eslabón se explica dependiendo de si la influencia es positiva o negativa y del tipo de interac-

ción: abductiva, deductiva-predictiva o intercausal (lo explicamos en la próxima sección) .

5 Conclusión

5.1 Ventajas de los MGP

La principal ventaja de los MGP frente a métodos alternativos para el tratamiento de la incertidumbre es su **fundamento normativo**, es decir, que se basan en una teoría matemática que indica qué probabilidades son necesarias, cómo deben obtenerse y cómo han de combinarse.[12]

Por tanto, los MGP gozan de este apoyo que no tienen los sistemas desarrollados *ad hoc*, tales como el método subjetivo de PROSPECTOR, los factores de certeza del MYCIN o los innumerables métodos de inferencia difusa. La única justificación posible para estos métodos es construir un sistema y ver que funciona correctamente. Sin embargo, puede repetirse el caso de MYCIN, cuya evaluación resultó completamente satisfactoria, a pesar de que tenía graves inconsistencias, que habrían quedado en evidencia si se hubieran escogido los casos de prueba oportunos.

Otra ventaja de prácticamente todos los MGP desarrollados hasta la fecha es que emplean **razonamiento causal**, lo cual permite tres tipos de razonamiento: *abductivo* (de los efectos a las causas), *deductivo-predictivo* (de las causas a los efectos) e *intercausal* (entre dos causas de un mismo efecto). Dicho de otro modo: los MGP, por su fundamento normativo, son capaces de obtener *todas* y *únicamente* las inferencias que están justificadas. En cambio, los sistemas basados en reglas (ya utilicen factores de certeza o lógica difusa) sólo admiten un tipo de inferencia, de los hallazgos hacia las hipótesis, sin tener en cuenta la distinción entre estos tres distintos tipos de razonamiento, lo cual puede dar lugar a serias inconsistencias (cf. [10, sec. 2.4] y las referencias que contiene).

Esta capacidad de los MGP es consecuencia directa del **tratamiento explícito de las dependencias e independencias condicionales**. Paradójicamente, el

[12] Aunque muchos de los principales partidarios de las redes bayesianas defienden la interpretación subjetivista de la probabilidad, nuestra postura personal es que en el campo de la medicina es posible y por tanto deseable utilizar probabilidades objetivas procedentes de estudios epidemiológicos [40]. Sólo en el caso de que no se hayan realizado los estudios estadísticos necesarios tendremos que recurrir a la estimación de los expertos humanos, pero siendo conscientes de que estamos intentando obtener *estimaciones subjetivas* de *magnitudes objetivas*. Aunque debate sobre la interpretación de la probabilidad es bastante complejo, afortunadamente todos los que trabajamos en el campo de los MGP estamos de acuerdo en los principios axiomáticos y en la forma de construir las redes, aunque las interpretaciones filosóficas sean diferentes.

argumento que con más frecuencia se utilizó en torno a los años 70 para negar un papel relevante a los modelos probabilistas en la inteligencia artificial, especialmente en aplicaciones médicas, era que incluían hipótesis injustificadas de independencia condicional. La situación se invirtió a partir de los trabajos de Pearl, Lauritzen, Spiegelhalter y otros, que demostraron que las redes bayesianas podían representar correctamente las relaciones de dependencia e independencia, y por otro lado, los trabajos de Heckerman, Horvitz y otros, que demostraron que los métodos basados en reglas contenían hipótesis de independencia condicional más estrictas y más difíciles de justificar —por no decir que eran generalmente falsas— que las contenidas en los MGP.

5.2 Limitaciones de los MGP en medicina

Uno de los inconvenientes principales de los MGP es que los algoritmos actuales tienen **complejidad exponencial** para redes generales; de hecho, la computación de la probabilidad en redes bayesianas y, por consiguiente, la evaluación de diagramas de influencia, es un problema NP-difícil tanto para los métodos exactos como para los aproximados, según se ha comentado en capítulos anteriores. Sin embargo, algunos de los algoritmos actuales son capaces de computar modelos médicos bastante complejos en intervalos de tiempo razonables, e incluso existen aproximaciones y modelos simplificados que permiten abordar problemas de mayor tamaño.

Otro de los obstáculos para la aplicación de los MGP a problemas médicos es la dificultad para **construir las redes**. No vamos a insistir más en ello porque ya hemos comentado en la sección 3.1 tanto la escasez de bases de datos completas como las carencias de conocimiento que dificultan la construcción con ayuda de expertos humanos. Sin embargo, conviene señalar que ésta no es una deficiencia de los MGP, sino una dificultad intrínseca de los problemas que estamos abordando. Hay otros métodos que no requieren tantos parámetros, incluso hay técnicas puramente cualitativas, pero en nuestra opinión, estas alternativas no aportan ninguna ventaja, sino que simplemente prescinden de información que resulta difícil de obtener, pero que es esencial.

Por otro lado, aunque hemos criticado anteriormente los sistemas basados en reglas, debemos reconocer que tienen una ventaja importante frente a los MGP —al menos en la actualidad— que es la facilidad para **controlar el razonamiento**, fijando objetivos y generando las preguntas oportunas. En teoría, los MGP tendrían ventaja en este punto, pues la teoría de la información y de la decisión permiten determinar exactamente cuál es la relación coste-efectividad de cada nuevo hallazgo; sin embargo, los algoritmos generales son impracticables por la desorbitada cantidad de tiempo que necesitarían. Existen métodos que introducen hipótiesis simplificadores con el fin de seleccionar las preguntas adecuadas,

pero aún no hay una metodología sólidamente establecida. De hecho, según nuestros conocimientos, los únicos sistemas comerciales que utilizan estos métodos son PATHFINDER [18] y los modelos de diagnóstico que Microsoft incorpora en Windows'95 y Windows'98.

Finalmente, otra limitación de los MGP es la dificultad para **explicar el razonamiento**, pues los métodos y modelos presentados en la sección 4.3 están aún lejos de ofrecer explicaciones comprensibles y satisfactorias para los médicos que pudieran utilizar los MGP desarrollados actualmente.

Nota. En la bibliografía hemos incluido solamente aquéllas referencias relativas a los MGP; el resto puede encontrarse en [10] (las relativas a las funciones de utilidad en medicina están en [22]). Muchas de ellas aparecen también en [27] o en [18].

Referencias

1. S. Andreassen, R. Hovorka, J. Benn, K. G. Olesen y E. R. Carson. A model-based approach to insulin adjustment. En: *Proceedings of the Third Conference on Artificial Intelligence in Medicine*, págs. 239–248, Maastrich, The Netherlands, 1991. Springer-Verlag.
2. S. Andreassen, M. Woldby, B. Falck y S. K. Andersen. MUNIN — A causal probabilistic network for interpretation of electromyographic findings. En: *Proceedings of the 10th International Joint Conference on Artificial Intelligence (IJCAI-87)*, págs. 366–372, Milan, 1987.
3. I. A. Beinlich, H. J. Suermondt, R. M. Chávez y G. F. Cooper. The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. En: *Proceedings of the 2nd European Conference on AI and Medicine*, págs. 247–256, London, 1989. Springer-Verlag, Berlin.
4. M. Ben-Bassat, R. W. Carlson, V. K. Puri, M. D. Davenport, J. A. Schriver, M. Latif, R. Smith, L. D. Portigal, E. H. Lipnick y M. H. Weil. Pattern-based interactive diagnosis of multiple disorders: The MEDAS system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **2**:148–160, 1980.
5. G. F. Cooper. *NESTOR: A Computer-Based Medical Diagnostic Aid that Integrates Causal and Probabilistic Knowledge.* Tesis doctoral, Dept. Computer Science, Stanford University, STAN-CS-84-1031, 1984.
6. G. F. Cooper. A dignostic method that uses causal knowledge and linear programming in the application of Bayes' formula. *Computer Methods and Programms in Biomedicine*, **22**:223–237, 1986.
7. R. Davis, B. G. Buchanan y E. H. Shortliffe. Retrospective on "Production rules as a representation for a knowledge-based consultation program". *Artificial Intelligence*, **59**:181–189, 1993.
8. F. T. de Dombal, J. R. Leaper, J. R. Staniland, A. McCann y J. Horrocks. Computer-aided diagnosis of acute abdominal pain. *British Medical Journal*, **2**:9–13, 1972.
9. F. J. Díez. Parameter adjustement in Bayes networks. The generalized noisy OR-gate. En: *Proceedings of the 9th Conference on Uncertainty in Artificial Intelligence (UAI'93)*, págs. 99–105, Washington D.C., 1993. Morgan Kaufmann, San Mateo, CA.
10. F. J. Díez. *Sistema Experto Bayesiano para Ecocardiografía.* Tesis doctoral, Dpto. Informática y Automática, UNED, Madrid, 1994.
11. F. J. Díez. Causality and probabilistic independence in graphical models. En: *EURO XV - INFORMS XXXIV*, Barcelona, 1997. Las transparencias se encuentran en `ftp://ftp.dia.uned.es/pub/research/bayes-nets/slides/barcelona.ps`.
12. F. J. Díez, J. Mira, E. Iturralde y S. Zubillaga. DIAVAL, a Bayesian expert system for echocardiography. *Artificial Intelligence in Medicine*, **10**:59–73, 1997.
13. M. J. Druzdzel. *Probabilistic Reasoning in Decision Support Systems: From Computation to Common Sense.* Tesis doctoral, Dept. Engineering and Public Policy, Carnigie Mellon University, 1993.
14. G. A. Gorry. Computer-assisted clinical decision making. *Methods of Information in Medicine*, **12**:45–51, 1973.

15. G. A. Gorry y G. O. Barnett. Experience with a model of sequential diagnosis. *Computers and Biomedical Research*, **1**:490–507, 1968.
16. P. Haddawy, J. Jacobson y C. E. Kahn Jr. BANTER, a Bayesian network tutoring shell. *Artificial Intelligence in Medicine*, **10**:177–200, 1997.
17. P. Haddawy, C. E. Kahn Jr. y M. Butarbutar. A Bayesian network model for radiological diagnosis and procedure seleccion: Work-up of suspected gallbladder disease. *Medical Physics*, **21**:1185–1192, 1994.
18. D. E. Heckerman. *Probabilistic Similarity Networks*. Tesis doctoral, Dept. Computer Science, Stanford University, STAN-CS-90-1316, 1990.
19. M. E. Hernando, E. J. Gómez, F. del Pozo y R. Corcoy. DIABNET: A qualitative model-based advisory system for therapy planning in gestational diabetes. *Medical Informatics*, **21**:359–374, 1996.
20. F. V. Jensen, B. Chamberlain, T.Ñordahl y F. Jensen. Analysis in HUGIN of data conflict. En: P. P. Bonissone, M. Henrion, L.Ñ. Kanal y J. F. Lemmer (eds.), *Uncertainty in Artificial Intelligence 6*, págs. 519–528. Elsevier Science Publishers, Amsterdam, 1991.
21. C. E. Kahn Jr., L. M. Roberts, K. A. Shaffer y P. Haddawy. Construction of a Bayesian network for mammographic diagnosis of breast cancer. *Computers in Biology and Medicine*, **27**:19–29, 1997.
22. P. Juez Martel y F. J. Díez Vegas. *Probabilidad y Estadística en Medicina. Aplicaciones en la Práctica Clínica y en la Gestión Sanitaria*. Ed. Díaz de Santos, Madrid, 1996.
23. S. L. Lauritzen y D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society, Series B*, **50**:157–224, 1988.
24. H. P. Lehmann y E. H. Shortliffe. THOMAS: building Bayesian statistical expert systems to aid in clinical decision making. *Computer Methods and Programs in Biomedicine*, **35**:251–260, 1991.
25. Y. C. Li. *Automated Probabilistic Transformation of a Large Medical Diagnostic Support System*. Tesis doctoral, Dept. of Medical Informatics, School of Medicine, University of Utah, 1995.
26. K. G. Olesen, U. Kjærulff, F. Jensen, F. V. Jensen, B. Falck, S. Andreassen y S. K. Andersen. A MUNIN network por the median nerve. A case study on loops. *Applied Artificial Intelligence*, **3**:385–403, 1989.
27. J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988. Reimpreso con correcciones en 1991.
28. G. Provan. Abstraction in belief networks: The role of intermediate states in diagnostic reasoning. En: *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence (UAI'95)*, págs. 464–471, Montreal, 1995. Morgan Kaufmann, San Francisco, CA.
29. G. M. Provan y J. R. Clarke. Dynamic network construction and updating techniques for for the diagnosis of acute abdominal pain. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **15**, 1993.

30. J. A. Reggia y B. T. Perricone. Answer justification in medical decision support systems based on Bayesian classification. *Computers in Biology and Medicine*, **15**:161–167, 1985.
31. S. Ríos-Insua, M. C. Bielza, M. Gómez, M. Fernández del Pozo, J. A. Sánchez Luna y S. Caballero. An intelligent decision system for jaundice management in newborn babies. En: F. J. Girón (ed.), *Cases Studies in Decision Analysis*. Springer-Verlag, Berlin, 1998. En prensa.
32. G. W. Rutledge, S. K. Andersen, J. X. Polaschek y L. M. Fagan. A belief network model for interpretation of ICU data. En: *Proceedings of the Fourteenth Annual Symposium of Computer Applications in Medical Care*, Washington, DC, 1990.
33. S. M. Schwartz, J. Baron y J. R. Clarke. A causal Bayesian model for the diagnosis of appendicitis. En: J. F. Lemmer y L.Ñ. Kanal (eds.), *Uncertainty in Artificial Intelligence 2*, págs. 423–434. Elsevier Science Publishers, Amsterdam, 1988.
34. P. Sember y I. Zukerman. Strategies for generating micro explanations for Bayesian belief networks. En: *Proceedings of the 5th Workshop on Uncertainty in Artificial Intelligence*, págs. 295–302, Windsor, Ontario, 1989.
35. M. A. Shwe, B. Middleton, D. E. Heckerman, M. Henrion, E. J. Horvitz, H. P. Lehmann y G. F. Cooper. Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base. Part I — The probabilistic model and inference algorithms. *Methods of Information in Medicine*, **30**:241–255, 1991.
36. B. Sierra y P. Larrañaga. Predicting the survival in malignant skin melanoma using Bayesian networks automatically induced by genetic algorithms. An empirical comparision between different approaches. *Artificial Intelligence in Medicine*, 1998. En prensa.
37. D. J. Spiegelhalter, R. C. G. Frankling y K. Bull. Assessment, criticism and improvement of imprecise subjective probabilities. En: M. Henrion, R. D. Shachter, L.Ñ. Kanal y J. F. Lemmer (eds.), *Uncertainty in Artificial Intelligence 5*, págs. 285–294. Elsevier Science Publishers, Amsterdam, 1990.
38. D. J. Spiegelhalter y R. P. Knill-Jones. Statistical and knowledge-based approaches to clinical decision support systems, with an application to gastroenterology. *Journal of the Royal Statistical Society, Series A*, **147**:35–77, 1984.
39. L. E. Súcar y D. F. Gillies. Probabilistic reasoning in high-level vision. *Image and Vision Computing*, **12**:42–60, 1994.
40. L. E. Súcar, D. F. Gillies y D. A. Gillies. Objective probabilities in expert systems. *Artificial Intelligence*, **61**:187–208, 1993.
41. H. J. Suermondt. *Explanation in Bayesian Belief Networks*. Tesis doctoral, Dept. Computer Science, Stanford University, STAN-CS-92-1417, 1992.
42. R. L. Teach y E. H. Shortliffe. An analysis of physicians' attitudes regarding computer-based clinical consultation systems. En: B. G. Buchanan y E. H. Shortliffe (eds.), *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, cap. 34, págs. 635–652. Addison-Wesley, Reading, MA, 1984.
43. H. R. Warner, A. F. Toronto, L. G. Veasy y R. Stephenson. A mathematical approach to medical diagnosis: Application to congenital heart disease. *Journal of the American Medical Association*, **177**:177–183, 1961.

44. M. P. Wellman y M. Henrion. Qualitative intercausal relations, or explaining "explaining away". En: J. A. Allen, R. Fikes y E. Sandewall (eds.), *Principles of Knowledge Representation: Proceedings of the Second International Conference*, págs. 535–546, San Mateo, CA, 1991. Morgan Kaufmann.

Algunas Aplicaciones de las Redes Bayesianas en Ingeniería

E. Castillo[1], J. M. Gutiérrez[1] y A. S. Hadi[2]

[1] Dpto. Matemática Aplicada y Ciencias de la Computación
Universidad de Cantabria
Avda. de los Castros s/n
39005 Santander
correo-e: castie@ccaix3.unican.es
[2] Department of Statistical Sciences
Universidad de Cornell, USA

Resumen

En este capítulo se presentan tres ejemplos: el tanque de presión, el sistema de distribeución de energía y el modelo de daño en vigas de hormigón armado. Con ellos se ilustran los tres casos típicos de redes bayesianas que se presentan en la práctica: modelos discretos, continuos y mixtos. Se comienza analizando la definición del problema y la elección de las variables que intervienen, así como las relaciones de independencia entre ellas mediante una representación gráfica apropiada. Una vez diseñada la estructura, se asignan las probabilidades condicionales completando así el proceso de definición de la red bayesiana. Seguidamente se ilustran los diversos métodos de propagación de evidencia (exacta, aproximada y simbólica), mediante su aplicación a diferentes hipótesis de evidencia y se discuten los resultados.

1 Introducción.

En este capítulo se aplica la metodología presentada en los capítulos anteriores a tres casos de la vida real:

- El problema del tanque de presión (Sección 2).
- El problema del sistema de distribución de energía (Sección 3).
- El problema de daño en vigas de hormigón armado (Secciones 4 y 5).

Con estos tres ejemplos se ilustran las etapas que deben seguirse cuando se analizan casos reales con los diferentes modelos probabilísticos que se han introducido en los capítulos anteriores.

Tal como cabe esperar, estas aplicaciones son más complicadas que los simples ejemplos que se utilizan para ilustrar ciertos métodos. Por otra parte, muchas de las hipótesis que se hacen para simplificar las cosas no suelen verificarse en la práctica (véase Pearl [32]. Por ejemplo:

- Las variables pueden ser discretas (binarias, categóricas, etc.), continuas, o mixtas (algunas discretas y otras continuas).
- Las relaciones entre las variables pueden ser muy complicadas y, como consecuencia, la especificación de los modelos probabilísticos puede ser difícil y dar problemas (véase Pearl [30] y Campos and Moral [2]).
- La propagación de evidencia puede requerir mucho tiempo, debido al gran número de parámetros y a la complejidad de las estructuras de la red (véase Díez [22]).

En los tres ejemplos que se presentan en este capítulo aparecen algunos de los problemas anteriores.

Todos los cálculos se han hecho utilizando los programas de ordenador escritos por los autores.[1]

El lector interesado en otros ejemplos puede consultar Díez [21], Castillo, Gutiérrez y Hadi [6,12], Castillo, Hadi y Solares [15].

2 El Sistema del Tanque de Presión

2.1 Definición del problema

La Figura 1 muestra el diagrama de un tanque de presión con sus elementos más importantes. Se trata de un tanque para almacenar un fluido a presión, que se introduce con la ayuda de una bomba activada por un motor eléctrico. Se sabe que el tanque no tiene problemas si la bomba funciona durante un periodo inferior a un minuto. Por tanto, se incorpora un mecanismo de seguridad, basado en un relé, F, que interrumpe la corriente tras funcionar 60 segundos. Además, un interruptor de presión, A, corta también la corriente si la presión en el tanque alcanza un cierto valor umbral, que se considera peligroso. El sistema incluye un interruptor, E, que inicia la operación del sistema; un relé, D, que suministra corriente tras la etapa de iniciación y la interrumpe tras la activación del relé F; y el relé C, que activa la operación del circuito eléctrico del motor. El objetivo del estudio consiste en conocer la probabilidad de fallo del tanque de presión, así como analizar la influencia de causas comunes de fallo.

2.2 Representación mediante una red bayesiana

Puesto que se está interesado en el análisis de todas las posibles causas de fallo del tanque B, se introduce una nueva variable K que denota este suceso. Se

[1] Estos programas pueden obtenerse en la dirección World Wide Web (WWW) http://ccaix3.unican.es/~AIGroup.

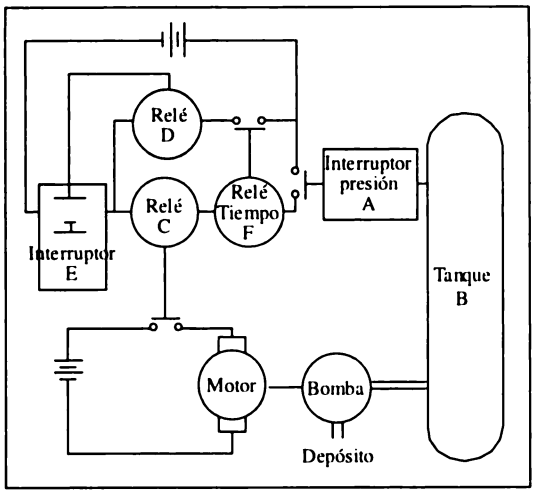


Figura 1. Un diagrama del sistema del tanque de presión.

usará la notación $K = 1$ para indicar el fallo del tanque, y $K = 0$ para el suceso complementario de no fallo. Similarmente, se utiliza el valor 1 para representar los fallos de las respectivas componentes $A, \ldots, F$ y el valor 0 para representar los sucesos correspondientes al no fallo.

Basándose en la descripción previa del problema, se puede escribir la siguiente expresión lógica para el fallo del tanque:

$$(B = 1) \vee (C = 1) \vee ((A = 1) \wedge ((E = 1) \vee (D = 1) \vee (F = 1))), \qquad (1)$$

donde los símbolos $\vee$ y $\wedge$ se usan para *o* e *y*, respectivamente. Esta expresión se obtiene combinando todas las posibilidades de fallo de las diferentes componentes que conducen al fallo del tanque. Esta ecuación puede expresarse de una forma mucho más intuitiva usando lo que se llama un *árbol de fallos*. La Figura 2(a) muestra el árbol de fallos correspondiente a la expresión (1). En este árbol, los fallos de los relés D y F se combinan para dar una causa de fallo intermedia, G; seguidamente G se combina con E para definir otra causa intermedia, H, y así sucesivamente. Este árbol incluye las variables iniciales $\{A, \ldots, F\}$ así como los fallos intermedios $\{G, \ldots, J\}$ que implican el fallo del tanque. Por tanto, el conjunto final de variables usadas en este ejemplo es $X = \{A, \ldots, K\}$.

Puesto que los fallos de las diferentes componentes del sistema son las causas de los fallos intermedios y, finalmente, del fallo del tanque, se puede obtener un grafo dirigido que reproduzca estas dependencias entre las variables que intervienen en el

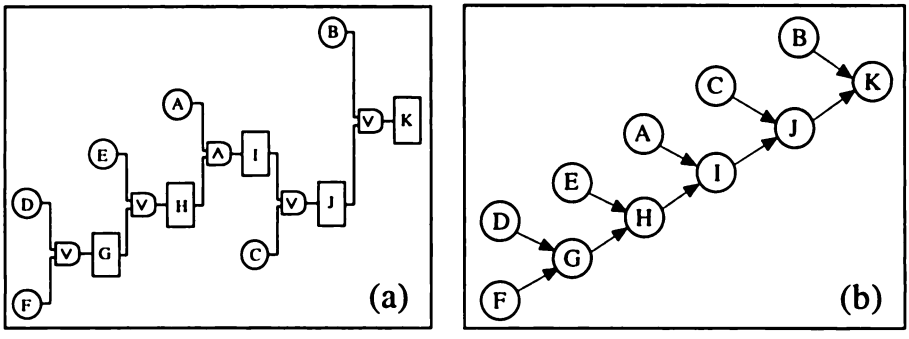


Figura 2. (a) Árbol de fallos del sistema del tanque de presión, y (b) grafo dirigido asociado.

modelo (véase la Figura 2(b)). Este grafo contiene la estructura de dependencia de la red bayesiana. De este grafo se deduce que la función de probabilidad conjunta de todos los nodos puede escribirse en la forma

$$p(x) = p(a)p(b)p(c)p(d)p(e)p(f)p(g|d,f)p(h|e,g)p(i|a,h)p(j|c,i)p(k|b,j). \quad (2)$$

Las distribuciones de probabilidad condicionales asociadas a las causas intermedias en el árbol de fallos se muestran en la Tabla 1, donde se dan sólo las probabilidades condicionales de los fallos, puesto que $p(no\ fallo) = 1 - p(fallo)$. Por otra parte, las probabilidades marginales asociadas a las componentes del sistema representan las probabilidades iniciales de fallo de cada una de sus componentes. Supóngase que las probabilidades son

$$\begin{array}{lll} p(A=1)=0.002, & p(B=1)=0.001, & p(C=1)=0.003, \\ p(D=1)=0.010, & p(E=1)=0.001, & p(F=1)=0.010. \end{array} \quad (3)$$

El grafo de la Figura 2(b), junto con las tablas de probabilidad que se muestran en (3) y en la Tabla 1, define una red bayesiana que corresponde al ejemplo del tanque de presión. La correspondiente función de probabilidad conjunta se da en (2).

2.3 Propagación de Evidencia

El grafo de la Figura 2(b) es un poliárbol, lo que significa que se puede utilizar el algoritmo para poliárboles para la propagación de la evidencia. Supóngase, en primer lugar, que no hay evidencia disponible. En este caso se obtienen las

D	F	$p(G=1\|D,F)$
1	1	1
1	0	1
0	1	1
0	0	0

E	G	$p(H=1\|E,G)$
1	1	1
1	0	1
0	1	1
0	0	0

A	H	$p(I=1\|A,H)$
1	1	1
1	0	0
0	1	0
0	0	0

C	I	$p(J=1\|C,I)$
1	1	1
1	0	1
0	1	1
0	0	0

B	J	$p(K=1\|B,J)$
1	1	1
1	0	1
0	1	1
0	0	0

Tabla 1. Probablidades condicionales de fallo de las variables intermedias en el sistema del tanque de presión.

probabilidades marginales de los nodos que se muestran en la Figura 3 (valores en la parte superior de cada nodo). Nótese que la probabilidad de fallo inicial del tanque es $p(K=1)=0.004$.

Supóngase ahora que las componentes F y D fallan, es decir, se tiene la evidencia $\{F=1, D=1\}$. Las nuevas probabilidades condicionales de los nodos se muestran en la Figura 3 (valores en la parte intermedia de cada nodo). Nótese que los fallos de los relés F y D inducen el fallo de los nodos intermedios G y H, pero la probabilidad de fallo del tanque es todavía pequeña ($p(K=1)=0.006$).

Para continuar la ilustración, supóngase que finalmente el interruptor de presión A también falla ($A=1$). Si se propaga la evidencia acumulada ($F=1, D=1, A=1$) se obtienen las nuevas probabilidades condicionales de los nodos que se muestran en la Figura 3 (valores en la parte inferior de cada nodo). Ahora, puesto que $p(K=1)=1$, el fallo de estas componentes F, D y A, implican el fallo de todos los nodos intermedios y el fallo del tanque.

2.4 Considerando Causas Comunes de Fallo

Supóngase ahora que hay una causa común de fallo para los relés C, D y F. Por ejemplo, supóngase que estos relés han sido construidos en las mismas circunstancias. Por ello, una posibilidad consiste en añadir un nuevo nodo Z (ver Figura 4), que representa la causa común de fallo, por ejemplo, fallo en la fabricación de los relés (se suponen procedentes del mismo proceso de fabricación). Este nuevo nodo se enlaza con los tres relés para indicar su efecto de causalidad

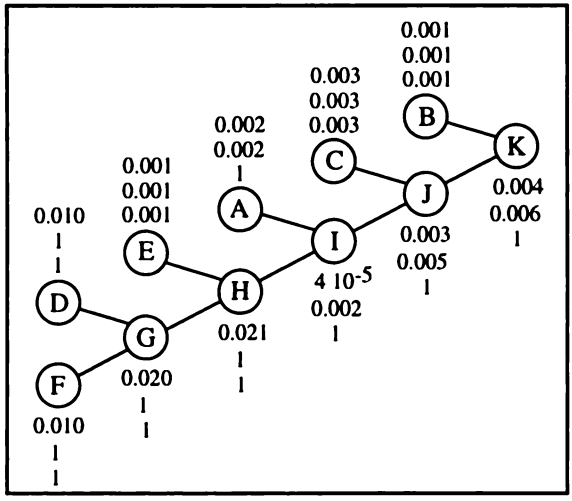


Figura 3. Las probabilidades marginales (arriba) y probabilidades condicionales dadas las evidencias $\{F = 1, D = 1\}$ (en medio), y $\{F = 1, D = 1, A = 1\}$ (abajo) para el tanque de presión.

común de fallo. Ahora, el grafo de la Figura 4 es un grafo múltiplemente conexo, y el algoritmo para poliárboles ya no puede aplicarse. En este caso, tiene que utilizarse un algoritmo de propagación más general, tal como el de agrupamiento, para propagar la evidencia en un árbol de unión asociado al grafo.

Según el grafo de la Figura 4, la función de probabilidad conjunta de los nodos puede factorizarse en la forma

$$p(x) = p(a)p(b)p(c|z)p(d|z)p(e)p(f|z)p(g|d,f) \\ p(h|e,g)p(i|h,a)p(j|c,i)p(k|b,j)p(z). \qquad (4)$$

Las correspondientes funciones de probabilidad condicionales se dan en la Tabla 2.

Para usar el algoritmo de agrupamiento, se necesita en primer lugar moralizar y triangular el grafo de la Figura 4. Uno de los grafos no dirigidos moralizados y triangulados obtenido aplicando el algoritmo de máxima cardinalidad (ver Castillo, Gutiérrez, and Hadi [13]) se muestra en la Figura 5.

Los conglomerados de este grafo son

$$C_1 = \{A, H, I\}, C_2 = \{E, G, H\}, C_3 = \{B, J, K\}, C_4 = \{J, I, C\}, \\ C_5 = \{C, Z, I\}, C_6 = \{I, Z, H\}, C_7 = \{H, G, Z\}, C_8 = \{Z, F, G, D\}.$$

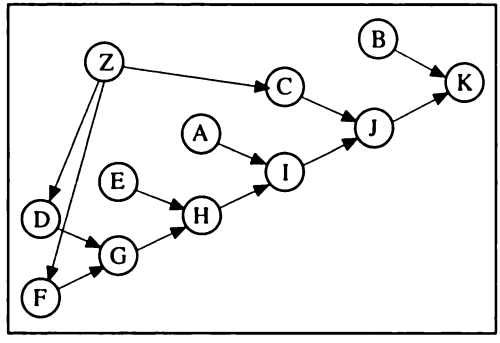


Figura 4. Grafo dirigido para el caso del tanque de presión cuando se considera una causa común de fallo Z.

A	$p(A)$
1	0.002

B	$p(B)$
1	0.001

E	$p(E)$
1	0.001

Z	$p(Z)$
1	0.003

Z	$p(C=1\|Z)$
1	0.9
0	0.001

Z	$p(D=1\|Z)$
1	0.9
0	0.001

Z	$p(F=1\|Z)$
1	0.9
0	0.001

D	F	$p(G=1\|D,F)$
1	1	1
1	$\bar{1}$	1
0	1	1
0	0	0

E	G	$p(H=1\|E,G)$
1	1	1
1	0	1
0	1	1
0	0	0

A	H	$p(I=1\|A,H)$
1	1	1
1	0	0
0	1	0
0	0	0

C	I	$p(J=1\|C,I)$
1	1	1
1	0	1
0	1	1
0	0	0

B	J	$p(K=1\|B,J)$
1	1	1
1	0	1
0	1	1
0	0	0

Tabla 2. Probabilidades de fallo para el tanque de presión cuando se considera una causa común de fallo Z.

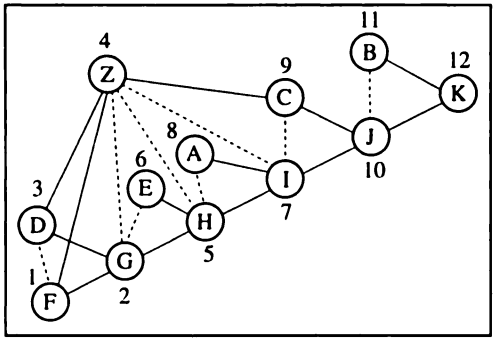


Figura 5. Un grafo moralizado y triangulado asociado al grafo dirigido de la Figura 4. Se muestra una numeración perfecta de los nodos.

Por ello, (4) puede escribirse también usando la representación potencial

$$p(x) = \psi(a,h,i)\psi(e,g,h)\psi(b,j,k)\psi(j,i,c)\psi(c,z,i)\psi(i,z,h)\psi(h,g,z)\psi(z,f,g,d),$$

donde

$$\begin{array}{ll} \psi(a,h,i) = p(a)p(i|h,a) & ; \psi(e,g,h) = p(e)p(h|e,g); \\ \psi(b,j,k) = p(b)p(k|b,j) & ; \psi(j,i,c) = p(j|c,i); \\ \psi(c,z,i) = p(z)p(c|z) & ; \psi(i,z,h) = 1; \\ \psi(h,g,z) = 1 & ; \psi(z,f,g,d) = p(d|z)p(f|z)p(g|d,f). \end{array}$$

EL árbol de unión obtenido se muestra en la Figura 6.

Supóngase, en primer lugar, que no hay evidencia disponible. Aplicando el algoritmo de agrupamiento a este árbol de unión u otros métodos (véase Cano [3]), se obtienen las probabilidades marginales iniciales de los nodos que corresponden a los valores superiores mostrados en la Figura 7. La probabilidad inicial de fallo del tanque es $p(K = 1) = 0.004$. Nótese que esta probabilidad coincide con la correspondiente al caso de no considerar las causas comunes.

Ahora se considera la evidencia $F = 1$ y $D = 1$, y se obtienen las probabilidades condicionales que corresponden a los valores intermedios mostrados en la Figura 7. La probabilidad condicional actualizada de fallo es ahora $p(K = 1) = 0.899$. Nótese que con esta misma evidencia la probabilidad de fallo del tanque en el caso de no considerar causas comunes de fallo era $p(K = 1) = 0.006$. La razón que explica esta diferencia es que se ha considerado que el relé C tiene una causa común de fallo con los relés F y D, por lo que, el fallo de aquel relé implica un aumento considerable de la probabilidad de fallo de éstos.

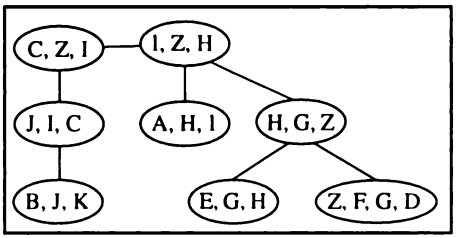


Figura 6. Un árbol de unión obtenido a partir del grafo no dirigido moralizado y triangulado en 5.

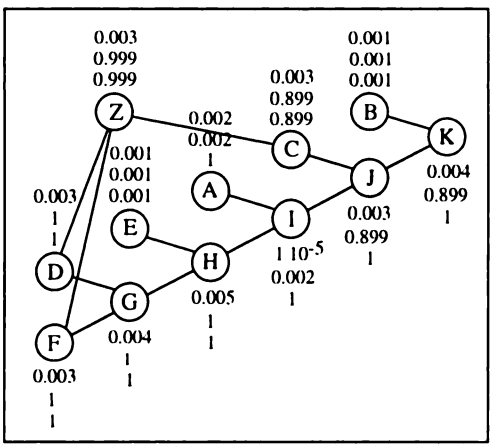


Figura 7. Probabilidades marginales iniciales de los nodos (arriba) y probabilidades condicionales dadas las evidencias $\{F = 1, D = 1\}$ (en medio), y $\{F = 1, D = 1, A = 1\}$ (abajo) para el tanque de presión con causa común de fallo.

Finalmente, cuando se considera la evidencia adicional $A = 1$, se obtiene $p(K = 1) = 1$, lo que indica que el tanque falla en este caso (véanse los valores inferiores de la Figura 7).

2.5 Propagación Simbólica de Evidencia

En esta sección se aplican los métodos de propagación simbólica de evidencia (véase Castillo, Gutiérrez y Hadi [5,7–9] o Castillo, Gutiérrez, Hadi y Solares [14]) para realizar un análisis de sensibilidad (ver Castillo, Gutiérrez y Hadi [10], Castillo, Solares y Gómez [16–19] o Castillo, Sarabia, Solares y Gómez [20]; es decir, se desea estudiar el efecto de cambiar las probabilidades asociadas a algunos nodos en las probabilidades de otros nodos de la red. Como ejemplo, modifiquemos algunas de las probabilidades condicionales en (3) y la Tabla 2 incluyendo algunos parámetros simbólicos para los nodos Z y D. Se reemplazan las probabilidades de los nodos Z y D por

$$p(D = 1|Z = 0) = 0.001, \quad p(D = 1|Z = 1) = q, p(Z = 1) = p.$$

donde $0 < p < 1$ y $0 < q < 1$ son los parámetros simbólicos.

Para el caso sin evidencia, usando el método simbólico descrito por Castillo, Gutiérrez y Hadi [13], se obtienen las probabilidades marginales de los nodos que se muestran en la Tabla 3.

En esta tabla, se ve que las probabilidades marginales de los nodos C, F y Z dependen de p pero no de q. También se puede ver que las probabilidades marginales de los nodos D, G, H, J, y K dependen de ambas p y q. Sin embargo, las probabilidades marginales de los nodos G, H, J y K son mucho más sensibles a p que a q (los coeficientes de p son mucho mayores que los de q). También, la probabilidad marginal del nodo I depende de q débilmente.

Los métodos simbólicos pueden usarse también para calcular las probabilidades condicionales de los nodos dada cualquier evidencia. Por ejemplo, la Tabla 3 (en su parte derecha) da las probabilidades condicionales de los nodos dada la evidencia $F = 1$.

Castillo, Gutiérrez y Hadi [13] muestran cómo pueden usarse las expresiones simbólicas, tales como las de la Tabla 3, para obtener cotas para las probabilidades marginales y condicionales de los nodos. Para el caso sin evidencia, la Tabla 4 muestra las probabilidades marginales iniciales de los nodos y sus correspondientes cotas inferior y superior, que se obtienen cuando los parámetros simbólicos toman sus valores extremos (los llamados *casos canónicos*):

$$\begin{aligned} p_{00} &= (p = 0, q = 0), \quad p_{01} = (p = 0, q = 1), \\ p_{10} &= (p = 1, q = 0), \quad p_{11} = (p = 1, q = 1). \end{aligned} \tag{5}$$

X_i	$p(X_i = 1)$	$p(X_i = 1 \mid F = 1)$
A	0.002	0.002
B	0.001	$\frac{0.001pq}{0.001 - 0.001p + pq}$
C	$0.001 + 0.899p$	$\frac{0.1pq}{0.001 - 0.001p + pq}$
D	$0.001 - 0.001p + pq$	1
E	0.001	0.001
F	$0.001 + 0.899p$	$\frac{0.9pq}{0.001 - 0.001p + pq}$
G	$0.002 + 0.898p + 0.1pq$	1
H	$0.003 + 0.897p + 0.1pq$	1
I	$0.0018p + 0.0002pq$	$\frac{0.2pq}{0.001 - 0.001p + pq}$
J	$0.001 + 0.899p + 0.00002pq$	$\frac{0.9002pq}{0.001 - 0.001p + pq}$
K	$0.002 + 0.898p + 0.00002pq$	$\frac{0.9pq}{0.001 - 0.001p + pq}$
Z	p	$\frac{pq}{0.001 - 0.001p + pq}$

Tabla 3. Probabilidades de los nodos sin evidencia y dada la evidencia $F = 1$, como función de los parámetros p y q.

Nótese que el rango de la variable, es decir, la diferencia entre las cotas superior e inferior, puede utilizarse como un indicador para medir la sensibilidad de las probabilidades a cambios en los valores de los parámetros (un rango reducido significa que es poco sensible).

X_i	p_{00}	p_{01}	p_{10}	p_{11}	Inf.	Sup.	Rango
$A = 1$	0.002	0.002	0.002	0.002	0.002	0.002	0.000
$B = 1$	0.001	0.001	0.001	0.001	0.001	0.001	0.000
$C = 1$	0.001	0.001	0.900	0.900	0.001	0.900	0.899
$D = 1$	0.001	0.001	0.000	1.000	0.000	1.000	1.000
$E = 1$	0.001	0.001	0.001	0.001	0.001	0.001	0.000
$F = 1$	0.001	0.001	0.900	0.900	0.001	0.900	0.899
$G = 1$	0.002	0.002	0.900	1.000	0.002	1.000	0.998
$H = 1$	0.003	0.003	0.900	1.000	0.003	1.000	0.997
$I = 1$	0.000	0.000	0.002	0.002	0.000	0.002	0.002
$J = 1$	0.001	0.001	0.900	0.900	0.001	0.900	0.899
$K = 1$	0.002	0.002	0.900	0.900	0.002	0.900	0.898
$Z = 1$	0.000	0.000	1.000	1.000	0.000	1.000	1.000

Tabla 4. Probabilidades marginales iniciales de los nodos y sus correspondientes cotas inferior y superior para los casos canónicos en (5).

3 Sistema de Distribución de Energía

3.1 Definición del problema

La Figura 8 muestra un sistema de distribución con tres motores, 1, 2, y 3, y tres temporizadores, A, B y C, que están normalmente cerrados. Una pulsación momentánea del pulsador F suministra energía de una batería a los relés G e I. A partir de ese instante G e I se cierran y permanecen activados eléctricamente. Para comprobar si los tres motores están operando propiamente, se envía una señal de prueba de 60 segundos a través de K. Una vez que K se ha cerrado, la

energía de la batería 1 llega a los relés R y M. El cierre de R arranca el motor 1. El cierre de T envía energía de la batería 1 a S. El cierre de S arranca el motor 3.

Tras un intervalo de 60 segundos, K debe abrirse, interrumpiendo la operación de los tres motores. Si K dejase de cerrarse tras los 60 segundos, los tres temporizadores A, B y C se abrirían, dejando sin energía a G y por tanto parando el sistema. Supóngase que K se abre para dejar sin energía a G y el motor 1 para. B y C actúan de forma similar para parar el motor 2 ó el motor 3, por lo que M o S deberían dejar de estar cerrados. En lo que sigue se analiza sólo el efecto sobre el motor 2. El análisis de los motores 1 y 3 se dejan como ejercicio al lector.

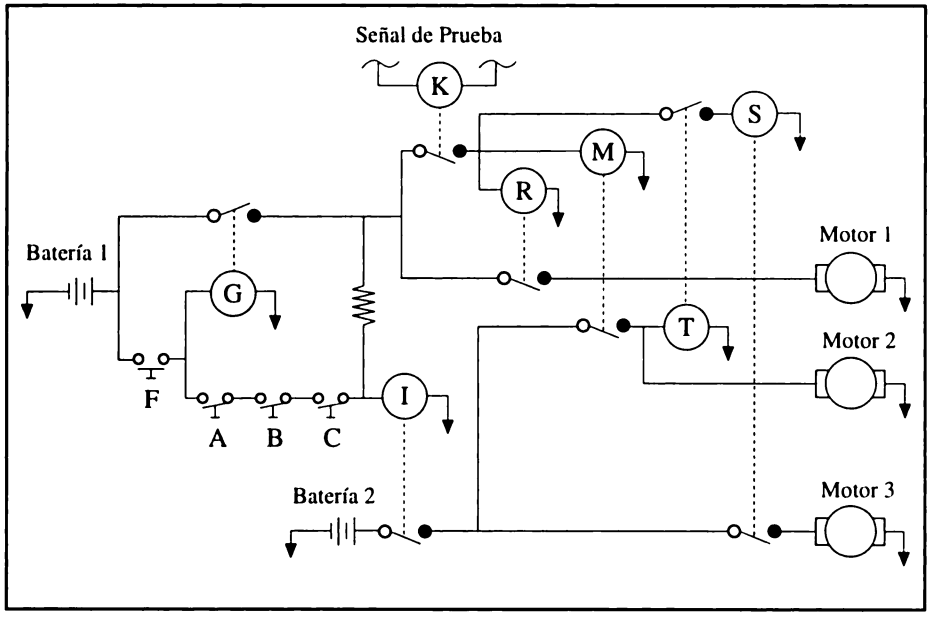


Figura 8. Un diagrama del sistema de distribución de energía.

3.2 Selección de Variables

Se está interesado en conocer el estado de operación del motor 2. Denotemos a esta variable aleatoria por Q y utilicemos la misma notación que en el ejemplo

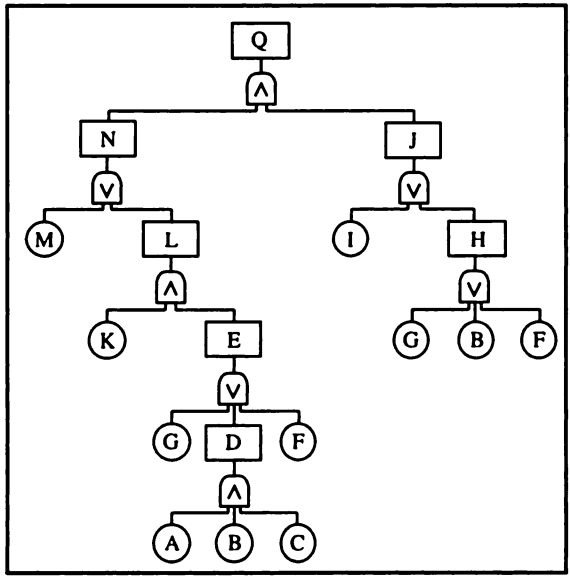


Figura 9. Árbol de fallos para el motor 2.

anterior ($Q = 1$ indica fallo y $Q = 0$ indica no fallo). La Figura 9 muestra el árbol de fallos y los conjuntos que conducen al fallo del sistema.

Este árbol de fallos conduce al grafo dirigido de la Figura 10 como modelo gráfico para una red bayesiana cuya función de probabilidad conjunta puede factorizarse en la forma

$$p(x) = p(a)p(b)p(c)p(d|a,b,c)p(e|d,f,g)p(f)p(g)p(h|b,f,g)$$
$$p(i)p(j|h,i)p(k)p(l|e,k)p(m)p(n|l,m)p(q|j,n).$$

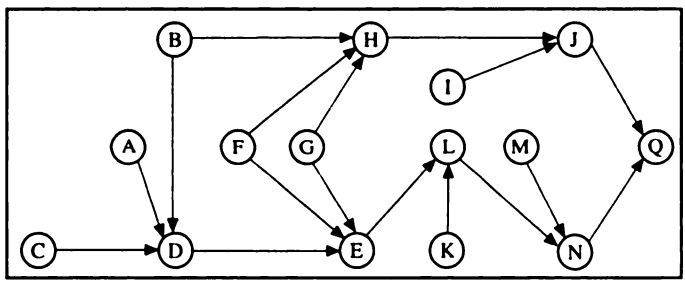


Figura 10. Grafo dirigido múltiplemente conexo para el sistema de distribución de energía (motor 2).

Las funciones de probabilidad condicionada necesarias para definir la función de probabilidad conjunta se dan en la Tabla 5 (se dan sólo las probabilidades de fallo puesto que $p(no\ fallo) = 1 - p(fallo)$). Las probabilidades marginales de los nodos terminales A, B, C, F, G, I, K y M son

$$p(A = 1) = 0.010,\quad p(B = 1) = 0.010,\quad p(C = 1) = 0.010,\quad p(F = 1) = 0.011$$
$$p(G = 1) = 0.011,\quad p(I = 1) = 0.001,\quad p(K = 1) = 0.002,\quad p(M = 1) = 0.003.$$

Para ilustrar mejor el procedimiento de trabajo, se usa un método exacto y otro aproximado para la propagación de evidencia en esta red bayesiana.

3.3 Propagación Exacta de Evidencia

La Figura 11 muestra el grafo no dirigido moralizado y triangulado que corresponde al grafo dirigido de la Figura 10. En la Figura 11 se da un grafo moralizado y triangulado asociado, junto con una numeración perfecta de los nodos.

I	H	$p(J=1\|I,H)$
1	1	1
1	0	1
0	1	1
0	0	0

E	K	$p(l=1\|E,K)$
1	1	1
1	0	0
0	1	0
0	1	0

L	M	$p(N=1\|L,M)$
1	1	1
1	0	1
0	1	1
0	0	0

J	N	$p(Q=1\|J,N)$
1	1	1
1	0	0
0	1	0
0	0	0

A	B	C	$p(D=1\|A,B,C)$
1	1	1	1
1	1	0	0
1	0	1	0
1	0	0	0
0	1	1	0
0	1	0	0
0	0	1	0
0	0	0	0

D	F	G	$p(E=1\|D,F,G)$
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	1
0	1	1	1
0	1	0	1
0	0	1	1
0	0	0	0

B	F	G	$p(H=1\|B,F,G)$
1	1	1	1
1	1	0	1
1	0	1	1
1	0	0	1
0	1	1	1
0	1	0	1
0	0	1	1
0	0	0	0

Tabla 5. Probabilidades condicionales de fallo de las variables del sistema de distribución de energía (motor 2).

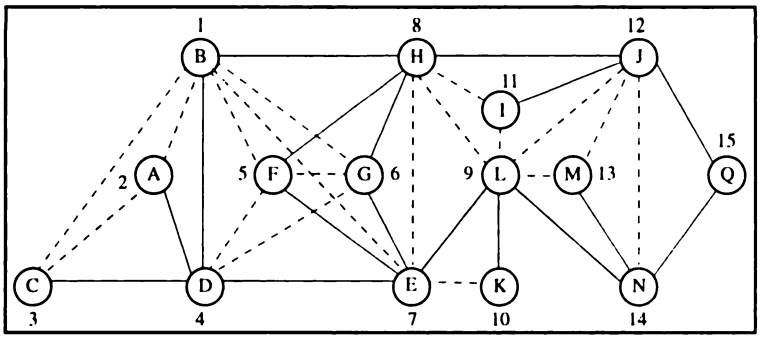


Figura 11. Grafo moralizado y triangulado asociado al grafo dirigido de la Figura 10. Se muestra una numeración perfecta de los nodos.

Los conglomerados, que pueden obtenerse del grafo de la Figura 11, son

$$C_1 = \{A, B, C, D\},\ C_2 = \{B, D, E, F, G\},\ C_3 = \{B, E, F, G, H\}$$
$$C_4 = \{E, H, L\},\quad C_5 = \{H, I, J, L\},\quad C_6 = \{E, K, L\},$$
$$C_7 = \{J, L, M, N\},\ C_8 = \{J, N, Q\},$$

lo que implica que la función de probabilidad conjunta de los nodos puede escribirse como función de las funciones potenciales como sigue:

$$\begin{aligned} p(x) = {} & \psi(a,b,c,d)\ \psi(b,d,e,f,g)\ \psi(b,e,f,g,h)\ \psi(e,h,l) \\ & \times \psi(h,i,j,l)\ \psi(e,k,l)\ \psi(j,l,m,n)\ \psi(j,n,q), \end{aligned} \tag{6}$$

donde

$$\begin{aligned} &\psi(a,b,c,d) = p(a)p(b)p(c)p(d|a,b,c)\ ;\ \psi(b,d,e,f,g) = p(f)p(g)p(e|d,f,g), \\ &\psi(b,e,f,g,h) = p(h|b,f,g) \qquad ;\ \psi(e,h,l) = 1, \\ &\psi(h,i,j,l) = p(i)p(j|i,h) \qquad ;\ \psi(e,k,l) = p(k)p(l|e,k), \\ &\psi(j,l,m,n) = p(m)p(n|l,m) \qquad ;\ \psi(j,n,q) = p(q|j,n). \end{aligned} \tag{7}$$

El árbol de unión correspondiente se muestra en la Figura 12.

Se usa el algoritmo de agrupamiento para obtener las probabilidades marginales iniciales de los nodos cuando no hay evidencia disponible. Estas probabilidades se muestran en la Figura 13. Supóngase ahora que se tiene la evidencia $K = 1$. Las probabilidades condicionales de los nodos dada esta evidencia se muestran en la Figura 13. En este caso, la probabilidad de fallo aumenta pasando del valor inicial $p(Q = 1) = 0.0001$ al valor $p(Q = 1|K = 1) = 0.022$.

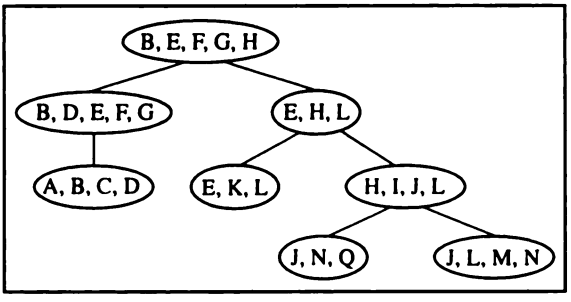


Figura 12. Un árbol de unión obtenido del grafo moralizado y triangulado de la Figura 11.

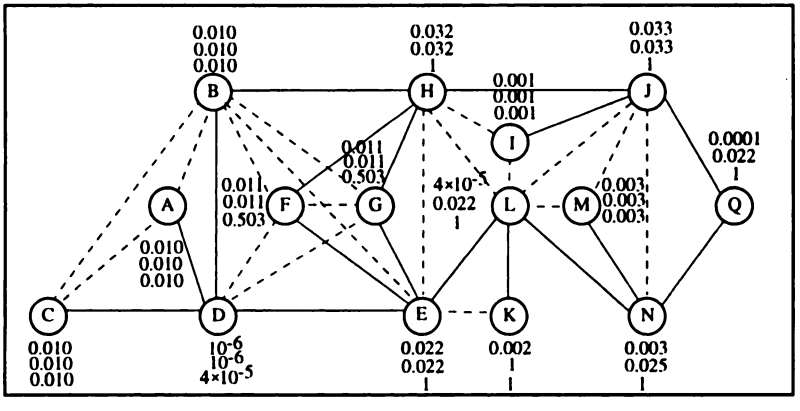


Figura 13. Probabilidades marginales (arriba), dada la evidencia $K = 1$ (en medio) y dada la evidencia $\{E = 1, K = 1\}$ (abajo) de los nodos para el sistema de distribución de energía (motor 2).

Cuando se introduce la evidencia adicional $E = 1$, entonces L y N también fallan. Consecuentemente, el sistema falla: $p(Q = 1|E = 1, K = 1) = 1$ (véase la Figura 13).

3.4 Propagación Aproximada de Evidencia

En un capítulo anterior se han introducido varios algoritmos para propagar la evidencia de forma aproximada (véase Castillo, Bouckaert, Sarabia y Solares [4],Bouckaert, Castillo y Gutiérrez [1], Castillo, Gutiérrez y Hadi [11], Pearl [27–29,31]). Se ha visto que el método de la *verosimilitud pesante* es uno de los más eficientes dentro de los métodos estocásticos y que el *muestreo sistemático* y el de *búsqueda de la máxima probabilidad* son los más eficientes dentro de los de tipo determinista en el caso de redes con probabilidades extremas. En este caso, se tiene una red bayesiana con tablas de probabilidad que contienen valores extremos (ceros y unos), una situación en la que el método de la verosimilitud pesante se sabe que es ineficiente. Sin embargo, en lo que sigue se comparan los métodos anteriores en el caso de esta red bayesiana (antes y después de conocer la evidencia $E = 1, K = 1$).

La Tabla 6 da el error de la aproximación,

$$\text{error} = |\text{exacta} - \text{aproximada}|,$$

para ambos métodos y para diferente número de réplicas. Claramente, el algoritmo de muestreo sistemático vence al de la verosimilitud pesante, conduciendo a errores mucho más pequeños para el mismo número de réplicas. La ineficiencia del algoritmo de la verosimilitud pesante es parcialmente debida a las probabilidades extremas. Puesto que la mayoría de las ocurrencias tienen asociada una probabilidad nula, el método más eficiente aquí es el de *búsqueda de la máxima probabilidad.* Por ejemplo, aún en el caso de que se considere un número de ocurrencias tan bajo como 10, el error obtenido (no mostrado) es menor que 3×10^{-6}.

4 Daño en Vigas de Hormigón Armado

En las Secciones 2 y 3 se han usado modelos de redes probabilísticas para definir, de una forma sencilla, funciones de probabilidad conjunta consistentes y directa para el caso de dos problemas de la vida real. El uso de redes bayesianas fue sencillo en esos casos porque todas las variables eran discretas y las relaciones de dependencia entre las variables no eran complicadas. En esta sección se presenta un problema en el que se mezclan variables discretas y continuas.

Número de Simulaciones	Error			
	Sin Evidencia		$E = 1, K = 1$	
	Verosimilitud	Sistemático	Verosimilitud	Sistemático
100	0.00205	0.00023	0.19841	0.00650
1,000	0.00021	5.25×10^{-6}	0.04300	0.00292
2,000	6.26×10^{-5}	3.91×10^{-6}	0.01681	0.00109
10,000	1.49×10^{-5}	4.35×10^{-7}	0.00302	3.34×10^{-5}
20,000	9.36×10^{-6}	1.22×10^{-7}	0.00265	1.78×10^{-5}
50,000	5.79×10^{-6}	3.08×10^{-8}	0.00053	7.66×10^{-6}
100,000	1.26×10^{-6}	3.06×10^{-9}	0.00011	2.08×10^{-6}

Tabla 6. Rendimiento de los métodos de la verosimilitud pesante y de muestreo sistemático con diferentes números de réplicas.

4.1 Definición del problema

En este caso, el objetivo consiste en determinar el daño de vigas de hormigón armado. En esta sección se ilustra este problema usando un modelo mixto con variables discretas y continuas. Alternativamente, en la Sección 5, se usan modelos de redes bayesianas normales (Gausianas) en los que todas las variables son continuas. Este ejemplo, que está tomado de Liu y Li [26] (véase también Castillo, Gutiérrez y Hadi [13]), ha sido modificado ligeramente por motivos ilustrativos. La primera parte de la formulación del modelo consta de dos etapas: selección de las variables e identificación de las dependencias.

4.2 Selección de las Variables

El proceso de la formulación del modelo comienza generalmente con la selección o especificación de un conjunto de variables de interés. Esta especificación corresponde a los expertos humanos en la especialidad (ingenieros civiles, en este caso). En nuestro ejemplo, la variable objetivo (el daño de una viga de hormigón armado) se denota por X_1. Un ingeniero civil identifica inicialmente 16 variables $(X_9, \ldots, X_{24})$ como las variables principales que influyen en el daño de una viga de hormigón armado. Además, el ingeniero identifica siete variables intermedias no observables $(X_2, \ldots, X_8)$ que definen estados parciales de la estructura. La Tabla 7 muestra la lista de variables y sus respectivas definiciones. La tabla también muestra el carácter continuo o discreto de cada variable. Las variables se miden usando una escala que está ligada directamente a la variable objetivo, es

decir, cuanto mayor es el valor de la variable mayor es la posibilidad de daño. Sea $X = \{X_1, \ldots, X_{24}\}$ el conjunto de todas las variables.

X_i	Tipo	Valores	Definición
X_1	Discreta	$\{0, 1, 2, 3, 4\}$	Daño de la viga
X_2	Discreta	$\{0, 1, 2\}$	Estado de agrietamiento
X_3	Discreta	$\{0, 1, 2\}$	Agrietamiento por cortante
X_4	Discreta	$\{0, 1, 2\}$	Corrosión del acero
X_5	Discreta	$\{0, 1, 2\}$	Agrietamiento por flexión
X_6	Discreta	$\{0, 1, 2\}$	Agrietamiento por retracción
X_7	Discreta	$\{0, 1, 2\}$	Peor grieta por flexión
X_8	Discreta	$\{0, 1, 2\}$	Estado de corrosión
X_9	Continua	$(0 - 10)$	Debilidad de la viga
X_{10}	Discreta	$\{0, 1, 2\}$	Flecha de la viga
X_{11}	Discreta	$\{0, 1, 2, 3\}$	Posición de la peor grieta de cortante
X_{12}	Discreta	$\{0, 1, 2\}$	Tamaño de la peor grieta de cortante
X_{13}	Discreta	$\{0, 1, 2, 3\}$	Posición de la peor grieta de flexión
X_{14}	Discreta	$\{0, 1, 2\}$	Tamaño de la peor grieta de flexión
X_{15}	Continua	$(0 - 10)$	Longitud de la peor grieta de flexión
X_{16}	Discreta	$\{0, 1\}$	Recubrimiento
X_{17}	Continua	$(0 - 100)$	Edad de la estructura
X_{18}	Continua	$(0 - 100)$	Humedad
X_{19}	Discreta	$\{0, 1, 2\}$	PH del aire
X_{20}	Discreta	$\{0, 1, 2\}$	Contenido de cloro en el aire
X_{21}	Discreta	$\{0, 1, 2, 3\}$	Número de grietas de cortante
X_{22}	Discreta	$\{0, 1, 2, 3\}$	Número de grietas de flexión
X_{23}	Discreta	$\{0, 1, 2, 3\}$	Retracción
X_{24}	Discreta	$\{0, 1, 2, 3\}$	Corrosión

Tabla 7. Definiciones de las variables que intervienen en el problema de daño de vigas de hormigón armado.

4.3 Identificación de las Dependencias

La etapa siguiente en la formulación del modelo consiste en la identificación de la estructura de las dependencias entre las variables seleccionadas. Esta identificación corresponde también a un ingeniero civil y se hace normalmente identificando

el menor conjunto de variables, $Vec(X_i)$, para cada variable X_i tales que

$$p(x_i|x \setminus x_i) = p(x_i|Vec(X_i)), \tag{8}$$

donde el conjunto $Vec(X_i)$ se llama el conjunto de *vecinos* de X_i. La ecuación (8) indica que la variable X_i es condicionalmente independiente del conjunto $R_i = X \setminus \{X_i, Vec(X_i)\}$ dado $Vec(X_i)$. Por ello, utilizando la notación de independencia condicional, se puede escribir $I(X_i, R_i|Vec(X_i))$. Las variables y sus correspondientes vecinos se muestran en las dos primeras columnas de la Tabla 8. Se sigue que si $X_j \in Vec(X_i)$, entonces $X_i \in Vec(X_j)$.

X_i	$Vec(X_i)$	Π_i
X_1	$\{X_9, X_{10}, X_2\}$	$\{X_9, X_{10}, X_2\}$
X_2	$\{X_3, X_6, X_5, X_4, X_1\}$	$\{X_3, X_6, X_5, X_4\}$
X_3	$\{X_{11}, X_{12}, X_{21}, X_8, X_2\}$	$\{X_{11}, X_{12}, X_{21}, X_8\}$
X_4	$\{X_{24}, X_8, X_5, X_2, X_{13}\}$	$\{X_{24}, X_8, X_5, X_{13}\}$
X_5	$\{X_{13}, X_{22}, X_7, X_2, X_4\}$	$\{X_{13}, X_{22}, X_7\}$
X_6	$\{X_{23}, X_8, X_2\}$	$\{X_{23}, X_8\}$
X_7	$\{X_{14}, X_{15}, X_{16}, X_{17}, X_8, X_5\}$	$\{X_{14}, X_{15}, X_{16}, X_{17}, X_8\}$
X_8	$\{X_{18}, X_{19}, X_{20}, X_7, X_4, X_6, X_3\}$	$\{X_{18}, X_{19}, X_{20}\}$
X_9	$\{X_1\}$	ϕ
X_{10}	$\{X_1\}$	ϕ
X_{11}	$\{X_3\}$	ϕ
X_{12}	$\{X_3\}$	ϕ
X_{13}	$\{X_5\}$	ϕ
X_{14}	$\{X_7\}$	ϕ
X_{15}	$\{X_7\}$	ϕ
X_{16}	$\{X_7\}$	ϕ
X_{17}	$\{X_7\}$	ϕ
X_{18}	$\{X_8\}$	ϕ
X_{19}	$\{X_8\}$	ϕ
X_{20}	$\{X_8\}$	ϕ
X_{21}	$\{X_3\}$	ϕ
X_{22}	$\{X_5\}$	ϕ
X_{23}	$\{X_6\}$	ϕ
X_{24}	$\{X_4\}$	ϕ

Tabla 8. Variables y sus correspondientes vecinos, $Vec(X_i)$ y padres, Π_i, para el caso del daño de una viga de hormigón armado.

Adicionalmente, pero opcionalmente, el ingeniero puede imponer ciertas relaciones de causa-efecto entre las variables, es decir, especificar qué variables entre las del conjunto $Vec(X_i)$ son causas directas de X_i y cuáles son los efectos directos de X_i. El conjunto de las causas directas de X_i se conoce como el conjunto de *padres* de X_i y se denota por Π_i.

En nuestro ejemplo, el ingeniero especifica las siguientes relaciones de causa-efecto, tal como se muestra en la Figura 14. La variable objetivo X_1, depende fundamentalmente de tres factores: X_9, la debilidad de la viga, disponible en la forma de un factor de daño, X_{10}, la flecha de la viga, y X_2, su estado de agrietamiento. El estado de agrietamiento, X_2, depende de cuatro variables: X_3, el estado de agrietamiento por cortante, X_6, el agrietamiento por retracción, X_4, la corrosión del acero, y X_5, el estado de agrietamiento por flexión. El agrietamiento por retracción, X_6, depende de la retracción, X_{23} y el estado de corrosión, X_8. La corrosión del acero, X_4, está ligada a X_8, X_{24}, X_{13} y X_5. El estado de agrietamiento por cortante, X_3, depende de cuatro factores: X_{11}, la posición de la peor grieta de cortante, X_{12}, la anchura de la misma, X_{21}, el número de grietas de cortante, y X_8. En el estado de agrietamiento por flexión, X_5 influyen tres variables: X_{13}, la posición de la peor grieta de flexión, X_{22}, el número de grietas de flexión, y X_7, la peor grieta de flexión. La variable X_{13} depende de X_4. La variable X_7 es una función de cinco variables: X_{14}, la anchura de la peor grieta de flexión, X_{15}, la longitud de la peor grieta de flexión, X_{16}, el recubrimiento, X_{17}, la edad de la estructura, y X_8, el estado de corrosión. La variable X_8 está ligada a tres variables: X_{18}, la humedad, X_{19}, el PH del aire, y X_{20}, el contenido de cloro en el aire.

El conjunto, de padres Π_i de cada una de las variables de la Figura 14 se muestra en la tercera columna de la Tabla 8. Si no se diesen relaciones causa-efecto las relaciones se representarían mediante aristas no dirigidas (una línea que conecta dos nodos).

4.4 Especificación de Distribuciones Condicionales

Una vez que se ha especificado la estructura gráfica, el ingeniero suministra un conjunto de probabilidades condicionales sugeridas por el grafo. Para simplificar la asignación de probabilidades condicionales, el ingeniero supone que éstas pertenecen a familias paramétricas (por ejemplo, Binomial, Beta, etc.). El conjunto de probabilidades condicionales se da en la Tabla 9, donde las cuatro variables continuas se suponen de tipo $Beta(a, b)$ con los parámetros indicados y las variables discretas se suponen *Binomiales* $B(n, p)$. La razón para esta elección es que la distribución beta tiene rango finito y una gran variedad de formas dependiendo de la elección de los parámetros.

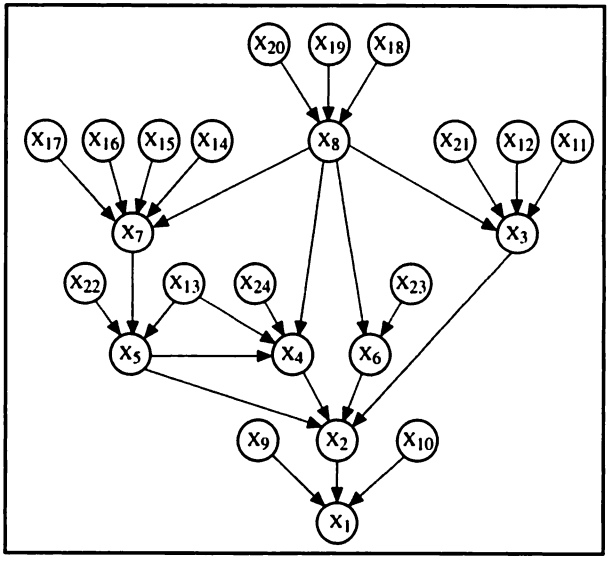


Figura 14. Grafo dirigido acíclico correspondiente al problema del daño en la viga de hormigón armado.

La variable X_1 puede tomar sólo cinco valores (estados): $0, 1, 2, 3, 4$. Con 0 se indica que la viga está libre de daño y con 4 que está seriamente dañada. Los valores intermedios, entre 0 y 4, son estados intermedios de daño. Las restantes variables se definen de forma similar usando una escala que está directamente ligada a la variable objetivo, es decir, cuanto mayores sean sus valores mayor es el daño.

X_i	$p(x_i\|u_i)$	Familia
X_1	$p(x_1\|x_9, x_{10}, x_2)$	$B(4, p_1(x_9, x_{10}, x_2))$
X_2	$p(x_2\|x_3, x_6, x_4, x_5)$	$B(2, p_2(x_3, x_6, x_4, x_5))$
X_3	$p(x_3\|x_{11}, x_{12}, x_{21}, x_8)$	$B(2, p_3(x_{11}, x_{12}, x_{21}, x_8))$
X_4	$p(x_4\|x_{24}, x_8, x_5)$	$B(2, p_4(x_{24}, x_8, x_5, x_{13}))$
X_5	$p(x_5\|x_{13}, x_{22}, x_7)$	$B(2, p_5(x_{13}, x_{22}, x_7))$
X_6	$p(x_6\|x_{23}, x_8)$	$B(2, p_6(x_{23}, x_8))$
X_7	$p(x_7\|x_{14}, x_{15}, x_{16}, x_{17}, x_8)$	$B(2, p_7(x_{14}, x_{15}, x_{16}, x_{17}, x_8))$
X_8	$p(x_8\|x_{18}, x_{19}, x_{20})$	$B(2, p_8(x_{18}, x_{19}, x_{20}))$
X_9	$f(x_9)$	$10 * Beta(0.5, 8)$
X_{10}	$p(x_{10})$	$B(2, 0.1)$
X_{11}	$p(x_{11})$	$B(3, 0.2)$
X_{12}	$p(x_{12})$	$B(2, 0.1)$
X_{13}	$p(x_{13})$	$B(3)$
X_{14}	$p(x_{14})$	$B(2, 0.1)$
X_{15}	$f(x_{15})$	$10 * Beta(1, 4)$
X_{16}	$p(x_{16})$	$B(1, 0.1)$
X_{17}	$f(x_{17})$	$100 * Beta(2, 6)$
X_{18}	$f(x_{18})$	$100 * Beta(2, 6)$
X_{19}	$p(x_{19})$	$B(2, 0.2)$
X_{20}	$p(x_{20})$	$B(2, 0.2)$
X_{21}	$p(x_{21})$	$B(3, 0.2)$
X_{22}	$p(x_{22})$	$B(3, 0.2)$
X_{23}	$p(x_{23})$	$B(3, 0.1)$
X_{24}	$p(x_{24})$	$B(3, 0.1)$

Tabla 9. Probabilidades marginales y condicionadas correspondientes a la red de la Figura 14.

Todas las variables discretas se supone que siguen una distribución binomial con parámetros N y p, con $N+1$ estados posibles para cada variable. Sin embargo,

estas distribuciones pueden reemplazarse por otras cualesquiera. El parámetro $0 \underline{E} p \underline{E} 1$ se especifica como sigue. Sean π_i los valores observados de los padres de un nodo dado X_i. La función $p_i(\pi_i)$, $i = 1, \ldots, 8$, de la Tabla 9 es una función que toma π_i como dato y produce una probabilidad asociada al nodo X_i, es decir, $p_i(\pi_i) = h(\pi_i)$. Por simplicidad considérese $\Pi_i = \{X_1, \ldots, X_m\}$. Entonces, algunos posibles ejemplos de $h(\pi_i)$ son

$$h(\pi_i) = \sum_{j=1}^{m} \frac{x_j/u_j}{m} \tag{9}$$

y

$$h(\pi_i) = 1 - \prod_{j=1}^{m} (1 - x_j/u_j), \tag{10}$$

donde u_j es una cota superior (por ejemplo, el valor máximo) de la variable aleatoria X_j. Las funciones $h(\pi_i)$ en (9) y (10) crecen con valores crecientes de Π_i. También satisfacen el axioma de la probabilidad $0 \underline{E} h(\pi_i) \underline{E} 1$. Debe señalarse aquí que estas funciones son sólo ejemplos, dados con la intención de ilustrar, y que pueden reemplazarse por otras funciones con las mismas propiedades.

La Tabla 10 da las funciones $h(\pi_i)$ utilizadas para calcular las probabilidades condicionales de la Tabla 9. Alternativamente, pudiera darse una tabla de distribuciones marginales o condicionales para cada variable discreta.

X_i	$p(\pi_i)$	$h(\pi_i)$
X_1	$p_1(x_9, x_{10}, x_2)$	Ec. (10)
X_2	$p_2(x_3, x_6, x_5, x_4)$	Ec. (9)
X_3	$p_3(x_{11}, x_{12}, x_{21}, x_8)$	Ec. (9)
X_4	$p_4(x_{24}, x_8, x_5, x_{13})$	Ec. (9)
X_5	$p_5(x_{13}, x_{22}, x_7)$	Ec. (9)
X_6	$p_6(x_{23}, x_8)$	Ec. (9)
X_7	$p_7(x_{14}, x_{15}, x_{16}, x_{17}, x_8)$	Ec. (9)
X_8	$p_8(x_{18}, x_{19}, x_{20})$	Ec. (9)

Tabla 10. Funciones de probabilidad requeridas para calcular las probabilidades condicionales de la Tabla 9.

4.5 Propagación de Evidencia

En este ejemplo se trata con variables discretas y continuas en la misma red. Por ello, se necesita un método de propagación de evidencia para tratar este tipo de red. El caso de variables continuas complica las cosas porque las sumas deben reemplazarse por integrales y el número de posibles resultados se hace infinito. Los métodos de propagación exacta no pueden ser usados aquí porque son aplicables sólo cuando las variables son discretas o pertenecen a familias simples (tales como la normal), y no existen métodos generales para redes mixtas de variables (para un caso especial véase Lauritzen y Wermouth [25]).

Sin embargo, se pueden utilizar los métodos de propagación aproximada. Por su eficacia computacional y generalidad, se elige el de la verosimilitud pesante. La propagación de evidencia se hace usando el conjunto de probabilidades marginales y condicionales de la Tabla 9. Para ilustrar la propagación de evidencia y para responder a ciertas preguntas del ingeniero, se supone que éste examina una viga de hormigón y obtiene los valores $x_9, \ldots, x_{24}$ correspondientes a las variables observables $X_9, \ldots, X_{24}$. Nótese que estos valores pueden medirse secuencialmente. En este caso, la inferencia puede hacerse también secuencialmente. La Tabla 11, muestra las probabilidades de daño X_1 de una viga dada para varios tipos de evidencia que van desde la evidencia nula al conocimiento de los valores que toman todas las variables $x_9, \ldots, x_{24}$. Los valores de la Tabla 11 se explican e interpretan a continuación.

Como ejemplo ilustrativo, supóngase que se desea determinar el daño (la variable objetivo X_1) en cada una de las situaciones siguientes:

- **No hay evidencia disponible.** La fila correspondiente a la evidencia acumulada "Ninguna"de la Tabla 10 da la probabilidad marginal inicial de cada uno de los estados de la variable objetivo X_1. Por ejemplo, la probabilidad de que una viga seleccionada al azar no esté dañada ($X_1 = 0$) es 0.3874 y la probabilidad de que esté seriamente dañada ($X_1 = 4$) es 0.1285. Estas probabilidades pueden ser interpretadas como que el 39% de las vigas son seguras y el 13% están seriamente dañadas.
- **Evidencia de daño alto.** Supóngase que se tienen los datos de todas las variables observables que se dan en la Tabla 11, donde la evidencia se obtiene secuencialmente en el orden dado en la tabla. Las probabilidades en la fila i-ésima de la Tabla 11 se calculan usando $x_9, \ldots, x_i$, es decir, se basan en evidencias acumuladas. Excepto para las variables clave X_9 y X_{10}, los valores de las restantes variables alcanzan valores altos, lo que da lugar a altas probabilidades de daño. Como puede verse en la última fila de la tabla, cuando se consideran todas las evidencias, se obtiene $p(X_1 = 4) \simeq 1$, una indicación de que la viga está seriamente dañada.

Evidencia	$p(X_1 = x_1 \mid evidencia)$				
Disponible	$x_1 = 0$	$x_1 = 1$	$x_1 = 2$	$x_1 = 3$	$x_1 = 4$
Ninguna	0.3874	0.1995	0.1611	0.1235	0.1285
$X_9 = 0.01$	0.5747	0.0820	0.1313	0.1002	0.1118
$X_{10} = 0$	0.6903	0.0651	0.0984	0.0606	0.0856
$X_{11} = 3$	0.6154	0.0779	0.1099	0.0783	0.1185
$X_{12} = 2$	0.5434	0.0914	0.1300	0.0852	0.1500
$X_{13} = 3$	0.3554	0.1033	0.1591	0.1016	0.2806
$X_{14} = 2$	0.3285	0.1052	0.1588	0.1043	0.3032
$X_{15} = 9.99$	0.3081	0.1035	0.1535	0.1096	0.3253
$X_{16} = 1$	0.2902	0.1054	0.1546	0.1058	0.3440
$X_{17} = 99.9$	0.2595	0.1029	0.1588	0.1064	0.3724
$X_{18} = 99.9$	0.2074	0.1027	0.1513	0.1010	0.4376
$X_{19} = 2$	0.1521	0.0937	0.1396	0.0908	0.5238
$X_{20} = 2$	0.1020	0.0813	0.1232	0.0786	0.6149
$X_{21} = 3$	0.0773	0.0663	0.1062	0.0698	0.6804
$X_{22} = 3$	0.0325	0.0481	0.0717	0.0437	0.8040
$X_{23} = 3$	0.0000	0.0000	0.0000	0.0001	0.9999
$X_{24} = 3$	0.0000	0.0000	0.0001	0.0000	0.9999

Tabla 11. Distribución aproximada del daño, X_1, dadas las evidencias acumuladas de $x_9, \ldots, x_{24}$ tal como indica la tabla. Los resultados se basan en 10000 réplicas.

– **Evidencia de daño bajo.** Ahora, supóngase que se tienen los datos de las variables observables dados en la Tabla 12, donde los datos se miden secuencialmente en el orden dado en la tabla. En este caso todas las variables toman valores bajos, lo que indica que la viga está en buenas condiciones. Cuando se considera toda la evidencia, la probabilidad de ausencia daño es tan alta como 1.

Evidencia	$p(X_1 = x_1 \mid evidencia)$				
Disponible	$x_1 = 0$	$x_1 = 1$	$x_1 = 2$	$x_1 = 3$	$x_1 = 4$
Ninguna	0.3874	0.1995	0.1611	0.1235	0.1285
$X_9 = 0$	0.5774	0.0794	0.1315	0.1002	0.1115
$X_{10} = 0$	0.6928	0.0630	0.0984	0.0603	0.0855
$X_{11} = 0$	0.7128	0.0550	0.0872	0.0615	0.0835
$X_{12} = 0$	0.7215	0.0571	0.0883	0.0551	0.0780
$X_{13} = 0$	0.7809	0.0438	0.0685	0.0469	0.0599
$X_{14} = 0$	0.7817	0.0444	0.0686	0.0466	0.0587
$X_{15} = 0$	0.7927	0.0435	0.0680	0.0441	0.0517
$X_{16} = 0$	0.7941	0.0436	0.0672	0.0421	0.0530
$X_{17} = 0$	0.8030	0.0396	0.0630	0.0428	0.0516
$X_{18} = 0$	0.8447	0.0330	0.0525	0.0316	0.0382
$X_{19} = 0$	0.8800	0.0243	0.0434	0.0269	0.0254
$X_{20} = 0$	0.9079	0.0217	0.0320	0.0217	0.0167
$X_{21} = 0$	0.9288	0.0166	0.0274	0.0172	0.0100
$X_{22} = 0$	0.9623	0.0086	0.0125	0.0092	0.0074
$X_{23} = 0$	0.9857	0.0030	0.0049	0.0037	0.0027
$X_{24} = 0$	1.0000	0.0000	0.0000	0.0000	0.0000

Tabla 12. Probabilidades aproximadas del daño, X_1, dada la evidencia acumulada de $x_9, \ldots, x_{24}$ como se indica en la tabla. Los resultados se basan en 10000 réplicas.

5 Daño en Vigas de Hormigón Armado: El Modelo Normal

5.1 Especificación del modelo

En esta sección se presenta una formulación alternativa al ejemplo de daño en vigas de hormigón armado introducido en la Sección 4. Aquí se supone que todas las variables son continuas y se distribuyen según una distribución normal.

Es importante notar que en la práctica diferentes especialistas pueden desarrollar diferentes estructuras de dependencia para el mismo problema. Por otra parte, el desarrollo de una red probabilística consistente y no redundante es una tarea dura, a menos que el problema pueda ser descrito mediante una red bayesiana o Markoviana, que automáticamente conducen a consistencia. En la Sección 4 se ha estudiado este problema desde un punto de vista práctico, describiendo las etapas a seguir para generar un diagrama causa-efecto único y consistente. Ahora se supone que la función de densidad conjunta de $X = \{X_1, X_2, \ldots, X_{24}\}$ es normal multivariada $N(\mu, \Sigma)$, donde μ es el vector de medias de dimensión 24, Σ es la matriz de covarianzas de dimensión 24×24, y las variables $X_1, \ldots, X_{24}$ se miden utilizando una escala continua que es consistente con la hipótesis de normalidad. Entonces, la función de densidad conjunta de X puede escribirse como

$$f(x_1, \ldots, x_{24}) = \prod_{i=1}^{24} f_i(x_i|\pi_i), \tag{11}$$

donde

$$f_i(x_i|\pi_i) \sim N\left(m_i + \sum_{j=1}^{i-1} \beta_{ij}(x_j - \mu_j); v_i\right), \tag{12}$$

m_i es la media condicional de X_i, v_i es la varianza condicional de X_i dados los valores de Π_i, y β_{ij} es el coeficiente de regresión asociado a X_i y X_j. Nótese que si $X_j \notin \Pi_i$ entonces $\beta_{ij} = 0$.

Alternativamente, se puede definir la función de densidad conjunta dando el vector de medias y la matriz de covarianzas. Shachter and Kenley [33] dan un método para pasar de una a otra forma de representación.

Por ello, se puede considerar el grafo de la Figura 14 como la estructura de una red bayesiana normal. Entonces, la etapa siguiente consiste en la definición de la función de densidad conjunta usando (11). Supóngase que las medias iniciales de todas las variables son ceros, los coeficientes β_{ij} de (12) se definen como se indica en la Figura 15, y las varianzas condicionales están dadas por

$$v_i = \begin{cases} 10^{-4}, & \text{si } X_i \text{ es no observable,} \\ 1, & \text{en otro caso.} \end{cases}$$

Nótese que los coeficientes de regresión son todos positivos, pues todas las variables están positivamente correladas. Valores mayores indican mayor daño de la viga. Entonces la red bayesiana normal está dada por (11). En lo que sigue se dan ejemplos que ilustran la propagación numérica y simbólica de evidencia.

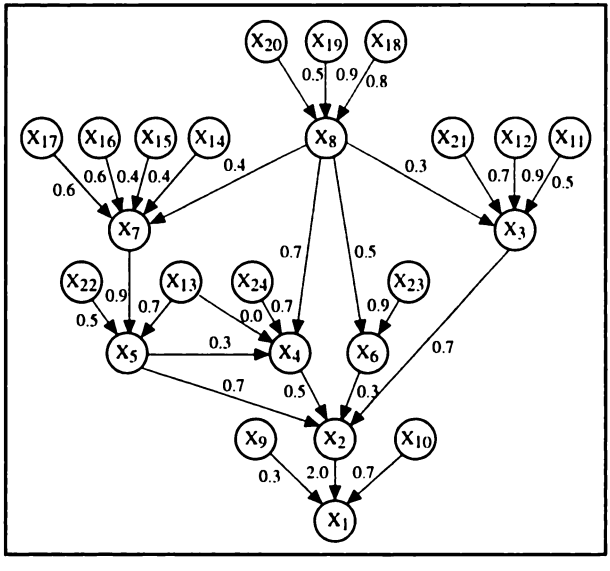


Figura 15. Grafo dirigido para evaluar el daño de una viga de hormigón armado. Los números cercanos a los enlaces son los coeficientes de regresión β_{ij} en (12) usados para definir la red bayesiana.

5.2 Propagación Numérica de Evidencia

Para propagar evidencia en la red bayesiana anterior, se usa el algoritmo incremental descrito en Castillo, Gutiérrez and Hadi [13]. Para ilustrar el proceso, se supone que el ingeniero examina una viga y obtiene secuencialmente los valores $\{x_9, x_{10}, \ldots, x_{24}\}$ correspondientes a las variables observables $X_9, \ldots, X_{24}$. Por simplicidad, supóngase que la evidencia es $e = \{X_9 = 1, \ldots, X_{24} = 1\}$, que indica que la viga está seriamente dañada.

De nuevo, se desea evaluar el daño (la variable objetivo, X_1). El vector de medias y la matriz de covarianzas condicionales de las variables restantes $Y = (X_1, \ldots, X_8)$ dada la evidencia e, que se han obtenido usando el algoritmo incremental, son

$$E(y|e) = (2.2, 3.32, 2., 4.188, 3.4964, 2.76, 7.2118, 15.4236),$$

$$Var(y|e) = \begin{pmatrix} 0.00010 \ldots & 0.00009 & 0.00003 & 0.00012 & 0.00023 \\ 0.00006 \ldots & 0.00008 & 0.00002 & 0.00015 & 0.00029 \\ 0.00005 \ldots & 0.00004 & 0.00001 & 0.00009 & 0.00018 \\ 0.00005 \ldots & 0.00010 & 0.00002 & 0.00022 & 0.00043 \\ 0.00009 \ldots & 0.00019 & 0.00003 & 0.00020 & 0.00039 \\ 0.00003 \ldots & 0.00003 & 0.00011 & 0.00011 & 0.00021 \\ 0.00012 \ldots & 0.00020 & 0.00010 & 0.00045 & 0.00090 \\ 0.00023 \ldots & 0.00039 & 0.00021 & 0.00090 & 1.00200 \end{pmatrix}.$$

Por ello, la distribución condicional de las variables en Y es normal con el vector de medias y la matriz de covarianzas anterior.

Nótese que en este caso, todos los elementos de la matriz de covarianzas excepto la varianza condicionada de X_1 son cercanos a cero, lo que indica que los valores medios son muy buenos estimadores de $E(X_2, \ldots, X_8)$ y razonables de $E(X_1)$.

Se puede considerar también la evidencia en forma secuencial. La Tabla 13 muestra la media y la varianza condicionales de X_1 suponiendo que la evidencia se obtiene secuencialmente en el orden indicado en la tabla. La evidencia oscila desde ausencia total de evidencia a un completo conocimiento de todas las variables $X_9, X_{10}, \ldots, X_{24}$. Por ejemplo, la media y la varianza inicial de X_1 son $E(X_1) = 0$ y $Var(X_1) = 19.26$, respectivamente; y la media y la varianza condicionales de X_1 dado $X_9 = 1$ son $E(X_1|X_9 = 1) = 0.30$ y $Var(X_1|X_9 = 1) = 19.18$. Nótese que tras observar la evidencia clave $X_9 = 1$, la media de X_1 aumenta de 0 a 0.3 y la varianza decrece de 19.26 a 19.18. Como puede verse en la última fila de la tabla, cuando se consideran todas las evidencias, $E(X_1|X_9 = 1, \ldots, X_{24} = 1) = 15.42$ y $Var(X_1|X_9 = 1, \ldots, X_{24} = 1) = 1.0$, una indicación de que la viga está seriamente dañada. En la Figura 16 se muestran varias de las funciones de densidad de X_1 resultantes de añadir nuevas evidencias. La figura muestra el daño creciente de

la viga en las diferentes etapas, tal como cabría esperar. Nótese que la media aumenta y la varianza disminuye, una indicación de que la incertidumbre decrece.

Etapa	Evidencia Disponible	Daño	
		Media	Varianza
0	Ninguna	0.00	19.26
1	$X_9 = 1$	0.30	19.18
2	$X_{10} = 1$	1.00	18.69
3	$X_{11} = 1$	1.98	17.73
4	$X_{12} = 1$	3.24	16.14
5	$X_{13} = 1$	4.43	17.72
6	$X_{14} = 1$	5.35	13.88
7	$X_{15} = 1$	6.27	13.04
8	$X_{16} = 1$	6.88	12.66
9	$X_{17} = 1$	7.49	12.29
10	$X_{18} = 1$	8.70	10.92
11	$X_{19} = 1$	10.76	6.49
12	$X_{20} = 1$	12.63	2.99
13	$X_{21} = 1$	13.33	2.51
14	$X_{22} = 1$	14.18	1.78
15	$X_{23} = 1$	14.72	1.49
16	$X_{24} = 1$	15.42	1.00

Tabla 13. Medias y varianzas del daño, X_1, dada la evidencia acumulada de $x_9, x_{10}, \ldots, x_{24}$.

Puede verse de los ejemplos anteriores que cualquier pregunta hecha por el ingeniero puede ser contestada simplemente mediante la propagación de evidencia usando el algoritmo incremental.

5.3 Cálculo Simbólico

Supóngase ahora que se está interesado en analizar el efecto de la flecha de la viga, X_{10}, en la variable objetivo, X_1. Entonces, se considera X_{10} como un nodo simbólico. Sea $E(X_{10}) = m, Var(X_{10}) = v, Cov(X_{10}, X_1) = Cov(X_1, X_{10}) = c$. Las medias y varianzas condicionales de todos los nodos pueden calcularse aplicando el algoritmo para propagación simbólica en redes bayesianas normales. Las medias y varianzas condicionales de X_1 dadas las evidencias secuenciales

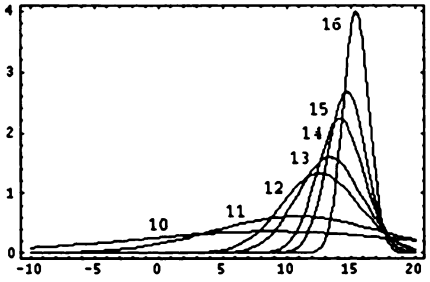


Figura 16. Distribuciones condicionadas del nodo X_1 correspondientes a la evidencia acumulada de la Tabla 13. El número de la etapa se muestra junto a cada gráfica.

$X_9 = 1, X_{10} = 1, X_{11} = x_{11}, X_{12} = 1, X_{13} = x_{13}, X_{14} = 1$, se muestran en la Tabla 14. Nótese que algunas evidencias (X_{11}, X_{13}) se dan en forma simbólica.

Nótese que los valores de la Tabla 13 son un caso especial de los de la Tabla 14. Pueden ser obtenidos haciendo $m = 0$, $v = 1$ y $c = 0.7$ y considerando los valores evidenciales $X_{11} = 1, X_{13} = 1$. Por ello, las medias y varianzas de la Tabla 13 pueden en realidad obtenerse de la Tabla 14 sin más que reemplazar los parámetros por sus valores. Por ejemplo, para el caso de la evidencia $X_9 = 1, X_{10} = 1, X_{11} = x_{11}$, la media condicional de X_1 es $(c - cm + 0.3v + 0.98vx_{11})/v = 1.98$. Similarmente, la varianza condicional de X_1 es $(-c^2 + 18.22v)/v = 17.73$.

Evidencia Disponible	Daño	
	Media	Varianza
Ninguna	0	19.26
$X_9 = 1$	0.3	19.18
$X_{10} = 1$	$\frac{c - cm + 0.3v}{v}$	$\frac{-c^2 + 19.18v}{v}$
$X_{11} = x_{11}$	$\frac{c - cm + 0.3v + 0.98vx_{11}}{v}$	$\frac{-c^2 + 18.22v}{v}$
$X_{12} = 1$	$\frac{c - cm + 1.56v + 0.98vx_{11}}{v}$	$\frac{-c^2 + 16.63v}{v}$
$X_{13} = x_{13}$	$\frac{c - cm + 1.56v + 0.98vx_{11} + 1.19vx_{13}}{v}$	$\frac{-c^2 + 15.21v}{v}$
$X_{14} = 1$	$\frac{c - cm + 2.48v + 0.98vx_{11} + 1.19vx_{13}}{v}$	$\frac{-c^2 + 14.37v}{v}$

Tabla 14. Medias y varianzas condicionales de X_1, inicialmente y tras la evidencia acumulada.

Referencias

1. Bouckaert, R., Castillo, E., and Gutiérrez, J. M. A Modified Simulation Scheme for Inference in Bayesian Networks. *International Journal of Approximate Reasoning*, 14:55–80, 1996.
2. Campos, L. M. D. and Moral, S. (1995), Independence Concepts for Convex Sets of Probabilities. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers, San Francisco, CA, 108–115.
3. Cano, J., Delgado, M., and Moral, S. (1993), An Axiomatic Framework for Propagating Uncertainty in Directed Acyclic Networks. *International Journal of Approximate Reasoning*, 8:253–280.
4. Castillo, E., Bouckaert, R., Sarabia, J. M., and Solares, C. Error Estimation in Approximate Bayesian Belief Network Inference. In *Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence (UAI'95)*, volume 2, pages 55–62, San Francisco, California, 1995. Morgan Kaufmann Publishers.
5. Castillo, E., Gutiérrez, J. M., and Hadi, A. S. Parametric Structure of Probabilities in Bayesian Networks. In *Lecture Notes in Artificial Intelligence 946, Proceedings of the European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty, ECSQARU'95*, pages 89–98, Copenhagen, 1995.
6. Castillo, E., Gutiérrez, J. M., and Hadi, A. S. An Introduction to Expert Systems for Medical Diagnoses. *Biocybernetics and Biomedical Engineering*, 15:63–84, 1995.
7. Castillo, E., Gutiérrez, J. M., and Hadi, A. S. Symbolic Propagation in Discrete and Continuous Bayesian Networks. In *Proceedings of the International Mathematica Symposium IMS'95*, pages 77–84, Southampton, 1995.
8. Castillo, E., Gutiérrez, J. M., and Hadi, A. S. Goal Oriented Symbolic Propagation in Bayesian Networks. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI'96).*, Portland (Oregon), 1263–1268, 1996.
9. Castillo, E., Gutiérrez, J. M., and Hadi, A. S. Symbolic Propagation in Bayesian Networks. *Networks*, Vol. 28, 31-43, 1996.
10. Castillo, E., Gutiérrez, J. M., and Hadi, A. S. Sensitivity Analysis in Discrete Bayesian Networks. *IEEE Transactions on Systems, Man and Cybernetics*, Vol 26, N. 7, 412–423, 1996.
11. Castillo, E., Gutiérrez, J. M., and Hadi, A. S. Improving Search-Based Inference in Bayesian Networks. Application to the MAP Problem. *The Eighth International Conference on Environmetrics*, Innsbruck, Austria, 1997.
12. Castillo, E., Gutiérrez, J. M., and Hadi, A. S. Combining Multiple Direct Graphical Representations into a Single Probabilistic Model. *CAEPIA'97.*, Torremolinos, Málaga, Spain, 645–651.
13. Castillo, E., Gutiérrez, J. M., and Hadi, A. S. *Expert Systems and Probabilistic Network Models*. Springer Verlag, New York, 1997.
14. Castillo, E., and Gutiérrez, J. M., Hadi, A. S., and Solares, C.. Symbolic Propagation and Sensitivity Analysis in Gaussian Bayesian Networks with Application to Damage Assessment,. *Artificial Intelligence in Engineering*, 11:173–181, 1997.
15. Castillo, E., Hadi, A. S., and Solares, C. Learning and Updating of Uncertainty in Dirichlet Models. *Machine Learning, Volume 26, Page 43-56*, 1996.

16. Castillo, E., Solares, C., and Gómez, P. Tail Sensitivity Analysis in Bayesian Networks. In *Proceedings of the Twelfth Conference on Uncertainty in Artificial Intelligence (UAI'96)*, Portland (Oregon), Morgan Kaufmann Publishers, San Francisco, California, 133-140, 1996.
17. Castillo, E., Solares, C., and Gómez, P. Estimating Extreme Probabilities Using Tail Simulated Data. *International Journal of Approximate Reasoning*, Vol 17 (02), 163-190, (1997).
18. Castillo, E., Solares, C., and Gómez, P. High Probability One-Sided Confidence Intervals in Reliability Models. *Nuclear Science and Engineering*, Vol. 126, 158–167, 1997.
19. Castillo, E., Solares, C., and Gómez, P. Tail Uncertainty Analysis in Complex Systems. *Artificial Intelligence* 96(2), 395-419, 1997.
20. Castillo, E., Sarabia, J. M., Solares, C., and Gómez, P. Uncertainty Analyses Using an Improved Fast Probability Integration Method. *The Eighth International Conference on Environmetrics*, Innsbruck, Austria, 1997.
21. Díez, F. J. (1994), *Sistema Experto Bayesiano para Ecocardiografía.* Ph.D. Thesis, Departamento de Informática y Automática, U.N.E.D., Madrid.
22. Díez, F. J. (1996), Local Conditioning in Bayesian Networks. *Artificial Intelligence.*
23. Larrañaga, P. (1995), *Aprendizaje Estructural y Descomposición de Redes Bayesianas Via Algoritmos Genéticos.* Ph.D. Thesis, Departamento de Ciencias de la Computación e Inteligencia Artificial, Universidad del País Vasco. Spain.
24. Larrañaga, P., Kuijpers, C., Murga, R., and Yurramendi, Y. (1996), Searching for the Best Ordering in the Structure Learning of Bayesian Networks. *IEEE Transactions on Systems, Man and Cybernetics*, 26. In press.
25. Lauritzen, S. L. and Wermuth, N. (1989), Graphical Models for Association Between Variables, Some of Which are Qualitative and Some Quantitative. *Annals of Statistics*, 17:31–54.
26. Liu, X. and Li, Z. (1994), A Reasoning Method in Damage Assessment of Buildings. *Microcomputers in Civil Engineering, Special Issue on Uncertainty in Expert Systems*, 9:329–334.
27. Pearl, J. (1984), *Heuristics.* Addison-Wesley, Reading, MA.
28. Pearl, J. (1986a), A Constraint-Propagation Approach to Probabilistic Reasoning. In Kanal, L.N. and Lemmer, J. F., editors, *Uncertainty in Artificial Intelligence.* North Holland, Amsterdam, 357–369.
29. Pearl, J. (1986b), Fusion, Propagation and Structuring in Belief Networks. *Artificial Intelligence*, 29:241–288.
30. Pearl, J. (1987a), Distributed Revision of Compatible Beliefs. *Artificial Intelligence*, 33:173–215.
31. Pearl, J. (1987b), Evidential Reasoning Using Stochastic Simulation of Causal Models. *Artificial Intelligence*, 32:245–257.
32. Pearl, J. (1988), *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, San Mateo, CA.
33. Shachter, R. and Kenley, C. (1989), Gaussian Influence Diagrams. *Management Science*, 35(5):527–550.

Un Sistema Experto es una herramienta informática que es capaz de simular el comportamiento de un experto humano en una materia especializada. Un problema clave en el desarrollo de sistemas expertos es encontrar la forma de representar y usar el conocimiento que los expertos humanos en esa materia poseen y utilizan. Este problema se hace más difícil por el hecho de que, en muchos campos, el conocimiento de los expertos es a menudo impreciso o incierto y, sin embargo, los expertos son capaces de llegar a conclusiones útiles.

Por tanto, todo sistema experto que pretenda razonar "como si" lo hiciese un ser humano debe ser capaz de trabajar con este tipo de información. Uno de los formalismos más potentes y mejor desarrollados para el tratamiento del conocimiento incierto es la Teoría de la Probabilidad, que nos permite medir la creencia que tenemos en la ocurrencia de un determinado suceso.

Este libro recoge los trabajos presentados en el VIII Curso de Verano de Informática: *Sistemas Expertos Probabilísticos*, por parte de un grupo de relevantes investigadores nacionales en el tema.

ISBN 84-89958-35-1

9 788489 958357

Ediciones de la Universidad de Castilla-La Mancha