

Modelos gráficos probabilísticos. Redes Bayesianas



Laura Morlana Ledesma
Trabajo de fin de grado de Matemáticas
Universidad de Zaragoza

Directora: María Beatriz Lacruz Casaucau
Diciembre 2023

Resumen

En el ámbito del análisis estadístico multivariante se encuentran los modelos gráficos probabilísticos, que surgen como una poderosa herramienta para representar y comprender relaciones complejas entre variables aleatorias. Entre estos se encuentran las redes Bayesianas, que son modelos gráficos dirigidos que tienen como estructura un grafo acíclico dirigido. Estas redes ofrecen una representación gráfica de las dependencias e independencias condicionales entre un conjunto de variables que intervienen en un determinado problema. En la actualidad, se utilizan modelos probabilísticos muy complejos, lo que hace de la inferencia probabilística un problema difícil de tratar. El objetivo de las redes Bayesianas reside en aprovechar las relaciones de independencia reflejadas en su estructura, lo que permite reducir la complejidad computacional de los problemas estudiados. En otras palabras, buscan minimizar la cantidad de parámetros necesarios para abordar un problema, lo que conduce a una mayor eficiencia de los algoritmos de inferencia asociados.

Las redes Bayesianas destacan por su capacidad para modelar problemas complejos en numerosos ámbitos de la actualidad, especialmente cuando la cantidad de datos aumenta exponencialmente. Algunas de sus aplicaciones incluyen desde la inteligencia artificial, hasta ámbitos de medicina, ingeniería o economía, demostrando su versatilidad en la resolución de problemas complejos en diversas áreas.

El objetivo de este trabajo es ofrecer una comprensión detallada de las redes Bayesianas, estudiando los elementos básicos que intervienen en un modelo gráfico probabilístico, tanto dirigido como no dirigido y las relaciones de independencia condicional que cada uno de estos representa. Además, se llevará a cabo un análisis de algunos algoritmos de inferencia exacta para problemas con variables discretas.

Abstract

In the field of multivariate statistical analysis, probabilistic graphical models are emerging as powerful tools to represent and understand complex relationships between random variables. Among these models are Bayesian networks, which are directed graphical models structured as directed acyclic graphs. These networks provide a graphical representation of conditional dependencies and independencies among a set of variables involved in a particular problem. Presently, highly complex probabilistic models are used, making probabilistic inference a challenging issue. The objective of Bayesian networks lies in leveraging the independence relationships reflected in their structure, thereby reducing the computational complexity of the studied problems. In other words, they aim to minimize the number of parameters necessary to address a problem, leading to greater efficiency in associated inference algorithms.

Bayesian networks stand out for their ability to model complex problems in numerous contemporary fields, especially as the volume of data increases exponentially. Some of their applications range from artificial intelligence to domains such as medicine, engineering, or economics, demonstrating their versatility in solving complex problems across various areas.

The goal of this work is to provide a detailed understanding of Bayesian networks, studying the basic elements involved in a probabilistic graphical model, both directed and undirected, and the conditional independence relationships that each of these represents. Additionally, an analysis of some exact inference algorithms for problems with discrete variables will be conducted.

Índice general

Resumen	III
Abstract	V
1. Grafos dirigidos y no dirigidos	1
1.1. Introducción	1
1.2. Conceptos básicos de grafos	1
1.3. Grafos no dirigidos	3
1.4. Grafos dirigidos	5
2. Independencias condicionales en grafos	7
2.1. Separación en grafos no dirigidos. U-separación	7
2.2. Separación en grafos dirigidos. D-separación	8
2.3. Modelos gráficos no dirigidos o redes de Markov	10
2.4. Modelos gráficos dirigidos o redes Bayesianas	11
2.5. Modelos gráficos equivalentes	12
3. Inferencia probabilística	13
3.1. Propagación exacta en poliárboles	14
3.2. Métodos de condicionamiento	16
3.3. Métodos de agrupamiento	18
3.3.1. Métodos de agrupamiento en redes de Markov	18
3.3.2. Métodos de agrupamiento en redes Bayesianas	19
3.3.3. Propagación en árboles de unión	22
A. Material complementario	27
A.1. Ejemplo de la definición alternativa a la D-separación	27
A.2. Propagación en poliárboles. Cálculo de λ_i y ρ_i	28
A.3. Algoritmo de propagación mediante árboles de unión. Detalles	30

B. Bnlearn	33
B.1. Construcción y análisis de una red Bayesiana	33
B.2. Comprobación de las independencias condicionales	35
B.3. Ejemplo de propagación en poliárboles	36
B.4. Ejemplo de métodos de agrupamiento	37
Bibliografía	41

Capítulo 1

Grafos dirigidos y no dirigidos

1.1. Introducción

En este trabajo se estudian los modelos gráficos probabilísticos, conocidos como MGP. Estos modelos están formados por un conjunto de nodos, que representan las variables aleatorias del problema a estudiar, unas aristas que unen los nodos definiendo así las relaciones de dependencia que existen entre las variables aleatorias y una función de probabilidad conjunta que caracteriza el modelo. Entre estos modelos gráficos se encuentran las redes Bayesianas, que son el principal concepto que estudiaremos en el trabajo, para luego ver algunos algoritmos de inferencia en redes Bayesianas.

Los MGP juegan un papel muy importante en el análisis estadístico multivariante ya que nos permiten simplificar el número de parámetros de la función de probabilidad conjunta que define el modelo mediante las relaciones de independencia condicional que refleja su estructura de grafo.

Este trabajo se estructura de la siguiente forma: en este capítulo se definen los conceptos básicos sobre grafos y se estudian algunas propiedades, también se introduce un ejemplo sencillo que se utilizará a lo largo del trabajo. En el Capítulo 2 se estudiarán las relaciones de independencia condicional en grafos dirigidos y no dirigidos y cómo factorizar las funciones de probabilidad conjunta según el tipo de modelo gráfico que tengamos. En el Capítulo 3 se verán diferentes tipos de algoritmos de inferencia exacta ilustrados con ejemplos, estos desempeñan un papel muy importante ya que permiten calcular con precisión las probabilidades requeridas para resolver problemas específicos en el contexto de redes Bayesianas. Finalmente, a lo largo de todo el trabajo, se han ido identificando las distintas funciones del paquete *bnlearn* [9] que permiten trabajar con modelos gráficos. Estas se enumeran en el Anexo B, lo que permite analizar y garantizar el buen funcionamiento de estos algoritmos de manera práctica, también se incluyen en el Anexo, algunos conceptos y ejemplos que pueden servir de repaso o de ayuda para la comprensión del trabajo.

Los conceptos teóricos se han extraído principalmente de [1], aunque también se han consultado otras fuentes como [2], [3], [4, Cap. 1], [5], [6], [7] y todos los ejemplos son de elaboración propia.

1.2. Conceptos básicos de grafos

Definición 1 (Grafo). Un grafo es un par de conjuntos $G = (X, L)$, donde $X = \{X_1, \dots, X_n\}$ es un conjunto finito de elementos (nodos) y L es un conjunto de aristas, es decir, un subconjunto de pares ordenados de elementos distintos de X . Las aristas entre los nodos $X_i \rightarrow X_j$ y $X_j \rightarrow X_i$ se denotan L_{ij} y L_{ji} respectivamente. Y $L = \{L_{ij} | X_i \text{ conecta con } X_j, \text{ con } X_i \neq X_j\}$ es el conjunto de todas las aristas del grafo.

Definición 2 (Arista). Una arista L_{ij} se dice dirigida si dado un grafo $G = (X, L)$ se tiene que $L_{ij} \in L$, pero $L_{ji} \notin L$ y será no dirigida si $L_{ij} \in L$ y $L_{ji} \in L$.

Aunque existe una definición más general de *grafo*, en este trabajo estamos interesados en aquellos en los que no existe un arco de un nodo a sí mismo y entre dos nodos conectados existe una única arista. Los grafos pueden ser *dirigidos* o *no dirigidos*. Serán dirigidos si todas sus aristas son dirigidas, y serán no dirigidos si todas sus aristas son no dirigidas. Para ilustrar este y el resto de conceptos, se utilizarán los grafos que se muestran en la Figura 1.1.

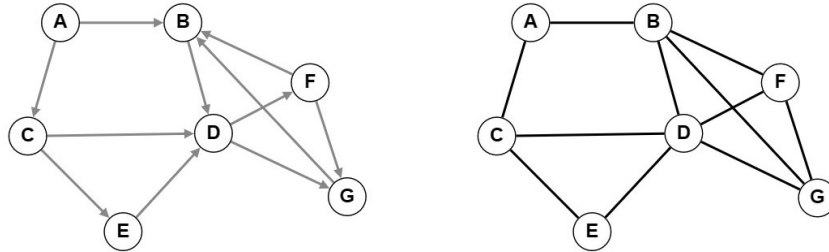


Figura 1.1: Un grafo dirigido (izquierda) y no dirigido (derecha).

Definición 3 (Conjunto adyacente). *El conjunto adyacente de un nodo X_i dado un grafo $G = (X, L)$ es el conjunto de nodos que son directamente alcanzables desde X_i , es decir, $\text{Ady}(X_i) = \{X_j \in X \mid L_{ij} \in L\}$.*

Ejemplo 1 (Conjunto adyacente). *En la Figura 1.2 se muestra que en el caso del grafo dirigido $\text{Ady}(D) = \{F, G\}$, mientras que en el grafo no dirigido $\text{Ady}(D) = \{B, C, E, F, G\}$.*

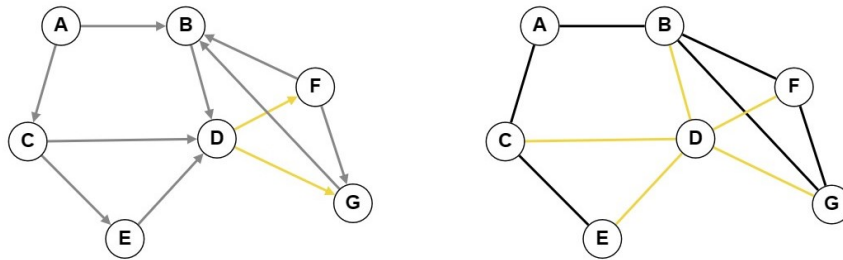


Figura 1.2: Conjunto adyacente del nodo D en un grafo dirigido (izquierda) y en un grafo no dirigido (derecha). Se resaltan con líneas amarillas los arcos que lo determinan.

Se observa entonces, que para determinar el conjunto de nodos adyacentes en un grafo dirigido es necesario tener en cuenta las direcciones de los arcos.

Definición 4 (Camino entre dos nodos). *Un camino entre dos nodos X_i y X_j es una sucesión de nodos $(X_{i_1}, \dots, X_{i_r})$, siendo $X_{i_1} = X_i$ el nodo inicial y $X_{i_r} = X_j$ el nodo final, de forma que existe una arista del nodo X_{i_k} al nodo $X_{i_{k+1}}$; con $k = 1, \dots, r-1$, es decir, $X_{i_{k+1}}$ es $\text{Ady}(X_{i_k})$.*

Definición 5 (Camino cerrado, bucle y ciclo). *Un camino $(X_{i_1}, \dots, X_{i_r})$ es cerrado si el nodo inicial coincide con el final, $X_{i_1} = X_{i_r}$. En grafos no dirigidos los caminos cerrados se denominan bucles, y en dirigidos se denominan ciclos.*

Para encontrar un camino entre dos nodos, como se ve en la Figura 1.3, hay que tener en cuenta la direccionalidad del grafo, por lo que el camino tomado en el no dirigido no lo podríamos tomar en el dirigido, pero al revés sí.

Ejemplo 2 (Camino, ciclos y bucles). *La Figura 1.3 muestra en el grafo dirigido (izquierda) el camino $A \rightarrow C \rightarrow E \rightarrow D \rightarrow F \rightarrow B \rightarrow D \rightarrow G$, que contiene el subcamino cerrado (ciclo) $D \rightarrow F \rightarrow B \rightarrow D$. Eliminando este ciclo del camino, se obtiene un camino más corto entre los nodos: $A \rightarrow C \rightarrow E \rightarrow D \rightarrow G$. Lo mismo sucede en el no dirigido (derecha) con el camino $A - B - D - E - C - D - G$, que contiene el subcamino cerrado (bucle) $D - E - C - D$, que cuando se elimina, se obtiene un camino más corto: $A - B - D - G$.*

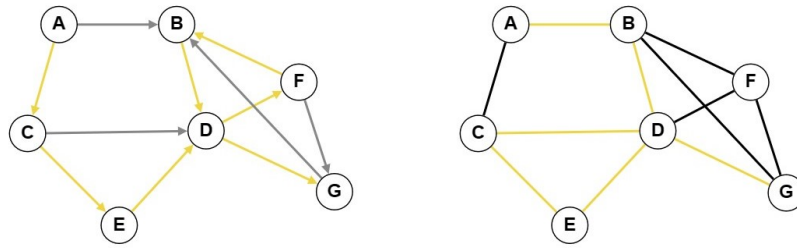


Figura 1.3: Caminos entre A-G con subcaminos cerrados.

Definición 6 (Grafo completo). Se dice que un grafo es completo si contiene una arista entre cada par de nodos. Además, un conjunto completo es un subconjunto de nodos S de un grafo G en el que existe una arista para cada par de nodos de S .

Ejemplo 3 (Subconjuntos completos). En la Figura 1.4 (izquierda) se muestra que además de los grafos de dos nodos unidos por una arista, como por ejemplo D_1 , tenemos los subconjuntos completos $D_3 = \{C, D, E\}$ y $D_4 = \{B, D, F, G\}$. Dentro de D_4 también se tienen los subconjuntos completos $D_{41} = \{B, D, F\}$ y $D_{42} = \{D, F, G\}$.

A lo largo de esta memoria usaremos, en general, grafos en los que existe al menos un camino (no necesariamente dirigido) entre todo par de nodos, denominados *grafos conexos*. Además, en el caso de grafos dirigidos solo estarán incluidos aquellos que no tengan ciclos, denominados *DAG (directed acyclic graphs)*, que son los que definen la estructura de una red Bayesiana.

1.3. Grafos no dirigidos

En este apartado se van a presentar algunos conceptos básicos sobre grafos no dirigidos.

Definición 7 (Clique). En un grafo no dirigido, un clique es un conjunto completo de nodos C que no es subconjunto propio de otro conjunto completo, es decir, es maximal.

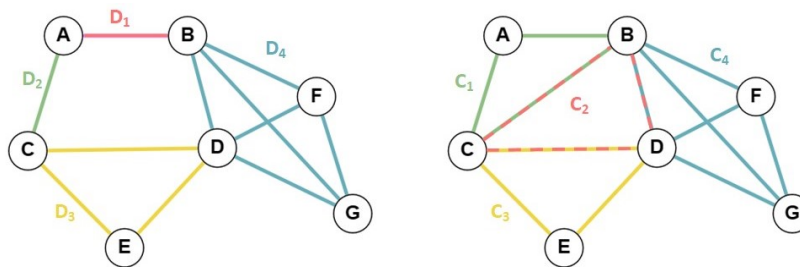


Figura 1.4: Conjuntos de cliques.

Ejemplo 4 (Clique). En la Figura 1.4 el grafo de la izquierda contiene los 4 cliques: D_1 (rosa), D_2 (verde), D_3 (amarillo), D_4 (azul). Pero si se añade por ejemplo la arista $C - B$, como se muestra en el grafo de la derecha, D_1 y D_2 dejan de ser conjuntos maximales, por lo que se tienen los cliques C_1 , C_2 , C_3 , C_4 que se muestran en el grafo de la derecha.

Definición 8 (Conjunto de vecinos). El conjunto de nodos adyacentes a un nodo X_i en un grafo no dirigido se denomina conjunto de vecinos de X_i , $\text{Vec}(X_i) = \{X_j | X_j \in \text{Ady}(X_i)\}$.

Un tipo muy importante de grafos son los *grafos triangulados*. Antes de ver cómo son estos grafos, se introduce una definición previa.

Definición 9 (Cuerda). Se denomina cuerda de un bucle a una arista que une dos nodos de un bucle y que no pertenece al bucle.

Los bucles de longitud 3 son los únicos que no pueden poseer cuerdas, es decir, son los menores elementos en los que puede descomponerse un bucle mediante la incorporación de cuerdas en el grafo. Se denominan *triángulos* a estos bucles.

Definición 10 (Grafo triangulado o cordal). Un grafo triangulado (o cordal) es un grafo no dirigido en el que cada bucle de longitud mayor o igual que 4 contiene al menos una cuerda.

Se puede convertir un grafo en triangulado mediante la *triangulación*, que consiste en añadir cuerdas. Aunque hay varias formas de triangular un grafo, para preservar al máximo la topología original de un grafo, la triangulación debe ser *minimal*, que consiste en añadir el mínimo número de aristas posibles.

Definición 11 (Triangulación minimal). Se dirá que la triangulación es minimal si contiene el mínimo número de cuerdas por debajo del cual no es posible triangular el grafo original.

Ejemplo 5 (Grafo triangulado y triangulación minimal). La Figura 1.5 muestra un ejemplo de que la triangulación no es única, ya que en el grafo del centro se ha añadido una cuerda entre B y C, pero podría haberse añadido la cuerda A – D o incluso las dos, como en el grafo de la derecha. En este caso, el grafo del centro tiene una triangulación minimal, mientras que el de la derecha no.

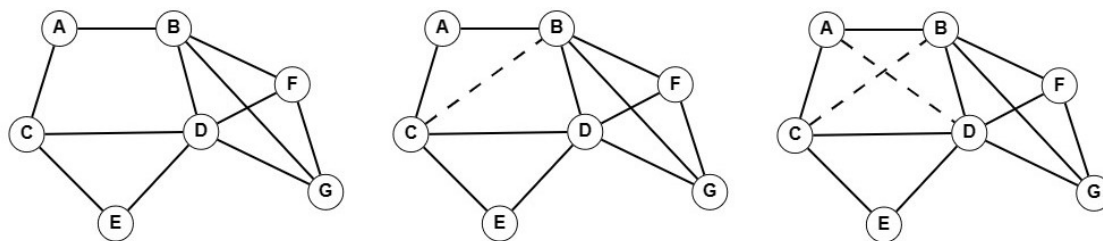


Figura 1.5: Grafo no dirigido no triangulado (izquierda) y grafos no dirigidos triangulados (centro y derecha).

Una propiedad interesante de los grafos triangulados que utilizaremos más adelante en los modelos gráficos no dirigidos es la *propiedad de intersección dinámica*.

Definición 12 (Propiedad de intersección dinámica). Se dice que un conjunto de cliques C_1, \dots, C_m cumplen la propiedad de intersección dinámica si el conjunto $C_i \cap (C_1 \cup \dots \cup C_{i-1})$ está contenido en al menos uno de los cliques $\{C_1, \dots, C_{i-1}\}, \forall i = 1, \dots, m$.

Es decir, se pueden ordenar los cliques de un grafo de forma que cumplan esta propiedad, a esta sucesión de cliques se le denomina *cadena de cliques*.

Ejemplo 6 (Propiedad de intersección dinámica). La Figura 1.4 (izquierda) muestra una sucesión de cliques que no cumplen la propiedad de intersección dinámica, ya que $D_4 \cap (D_1 \cup \dots \cup D_3) = \{B, D\}$, que no está contenido en ninguno de los cliques D_1, D_2 y D_3 . Pero el grafo de la derecha sí muestra una sucesión que la cumple, pues se puede comprobar que:

$$\begin{aligned} C_2 \cap C_1 &= \{B, C\} \subset C_1; \\ C_3 \cap (C_1 \cup C_2) &= \{C, D\} \subset C_2; \\ C_4 \cap (C_1 \cup C_2 \cup C_3) &= \{B, D\} \subset C_2. \end{aligned}$$

Es decir, $\{C_1, C_2, C_3, C_4\}$ forman una cadena de cliques.

El siguiente teorema caracteriza el tipo de grafos que tienen asociados una cadena de cliques.

Teorema 1 (Cadena de cliques). Un grafo no dirigido tiene asociada una cadena de cliques si y solo si es triangulado.

1.4. Grafos dirigidos

Una vez visto qué son los grafos no dirigidos, vamos a estudiar los grafos dirigidos. En particular en este trabajo nos interesan los grafos acíclicos dirigidos.

Definición 13 (Padres e hijos). *Dados dos nodos X_i y X_j , si existe una arista dirigida $X_i \rightarrow X_j$, del nodo X_i al nodo X_j , entonces se dice que el nodo X_i es padre del nodo X_j , y que el nodo X_j es hijo del nodo X_i .*

Definición 14 (Familia de un nodo). *Se denomina familia de un nodo al conjunto formado por un nodo y sus padres.*

Ejemplo 7 (Padres de un nodo). *El conjunto de padres del grafo de la Figura 1.6 es: $\text{Padres}(A) = \{\emptyset\}$, $\text{Padres}(B) = \{A, F\}$, $\text{Padres}(C) = \{A\}$, $\text{Padres}(D) = \{B, C\}$, $\text{Padres}(E) = \{C\}$, $\text{Padres}(F) = \{\emptyset\}$, $\text{Padres}(G) = \{F\}$, $\text{Padres}(H) = \{D, G, I\}$.*

En grafos dirigidos el conjunto de hijos de un nodo coincide con el conjunto de nodos adyacentes.

Ejemplo 8 (Familia de un nodo). *En la Figura 1.6 la familia asociadas al nodo B es $\{A, B, F\}$.*

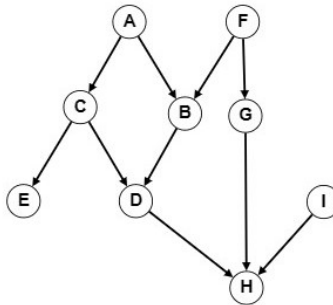


Figura 1.6: Grafo dirigido

Definición 15 (Ascendientes y descendientes). *Se dice que X_j es ascendiente de un nodo X_i si existe un camino de X_j a X_i y se dice que X_j es descendiente de X_i si existe un camino de X_i a X_j .*

Ejemplo 9 (Ascendientes y descendientes). *En la Figura 1.6, A es ascendiente de $\{B, C, D, E, H\}$ porque todos estos nodos son alcanzables a través de un camino que comienza en el nodo A. Al igual que C y D son algunos de los ascendientes del nodo H. Por otra parte, $\{B, C, D, E, H\}$ es el conjunto de descendientes del nodo A.*

Definición 16 (Esqueleto o grafo asociado). *Dado un grafo dirigido G , el grafo no dirigido obtenido al reemplazar cada arista dirigida por la correspondiente arista no dirigida, se denomina grafo no dirigido asociado a G o esqueleto de G .*

Definición 17 (V-estructura). *Una terna de nodos (X, Z, Y) se dice que es una v-estructura si las aristas desde los nodos X y Y convergen al nodo Z y además, no existe ninguna arista entre los nodos X e Y .*

Ejemplo 10 (V-estructura). *En el grafo de la Figura 1.6 las v-estructuras son las ternas de nodos (A, B, F) , (C, D, B) , (D, H, G) , (G, H, I) , (D, H, I) .*

Definición 18 (Grafo moral). *El grafo moral de un grafo dirigido G es el grafo no dirigido asociado que se obtiene al añadir una arista entre los nodos padre en una v-estructura en el grafo no dirigido.*

Ejemplo 11 (Esqueleto y grafo moral). *La Figura 1.7 muestra un grafo dirigido y su correspondiente esqueleto y grafo moral.*

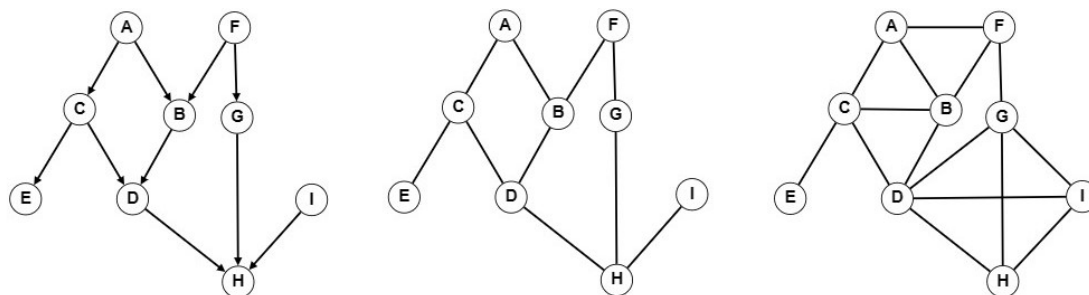


Figura 1.7: Grafo dirigido (izquierda), su esqueleto (centro) y su grafo moral (derecha).

Definición 19 (Árbol simple y poliárbol). Un grafo dirigido se denomina *árbol* si existe un único camino (no necesariamente dirigido) entre cada par de nodos. Se dice que es un *árbol simple* si cada nodo tiene como máximo un padre, y se dice que es un *poliárbol* en caso contrario.

Este tipo de grafos tendrán un papel relevante en los algoritmos de inferencia exacta del Capítulo 3.

Vamos a introducir un ejemplo sencillo de modelo gráfico dirigido o red Bayesiana que modela el riesgo de que haya un incendio en el norte y en el sur de una región según ciertos factores que intervienen. En este capítulo se definirán las variables aleatorias del problema y las relaciones existentes entre ellas, mientras que en el Capítulo 2 se estudiarán las relaciones de independencia existentes entre las variables. Finalmente, en el Capítulo 3 se utilizarán la inferencia probabilística para calcular la probabilidad de que ocurra un incendio según ciertos factores dados.

Ejemplo 12 (Riesgo de incendio en una región). En la Figura 1.8 se muestra una red Bayesiana con 5 variables aleatorias binarias (nodos) y las aristas representan cómo están relacionadas estas. Se han definido las variables con solo 2 posibles valores por simplicidad en la descripción del modelo: el nodo V representa la estación del año, tomará el valor $V = 1$ para las estaciones verano/primavera, y el valor $V = 0$ para invierno/otoño. El nodo N representa la situación en el mapa, tomará el valor $N = 1$ si se trata del norte, y el valor $N = 0$ si se trata del sur. El nodo C representa si es un día caluroso, tomando el valor $C = 1$ si hace calor, y el valor $C = 0$ si hace frío. El nodo L representa si es un día lluvioso, tomando el valor $L = 1$ si llueve, y el valor $L = 0$ si no llueve. Finalmente, el nodo I representa la probabilidad de que haya un incendio, tomando el valor $I = 1$ si hay incendio y el valor $I = 0$ si no lo hay.

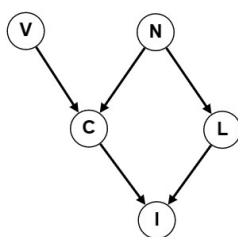


Figura 1.8: Grafo dirigido que representa las relaciones entre las variables del problema.

En un grafo dirigido, las relaciones directas (que en determinadas ocasiones se pueden interpretar como causas) entre las variables se indican mediante flechas entre estas. Por lo que este grafo representa las siguientes relaciones mediante su estructura: que un día sea caluroso, depende de la estación del año en la que nos encontremos y de la región del mapa que se está observando. Sin embargo, que un día sea lluvioso se considera que solo depende de la región del mapa (aunque normalmente estaría influenciada por más factores, se considera así en este caso por simplicidad). Por último, que haya un incendio dependerá de la temperatura y las precipitaciones de ese día.

Capítulo 2

Independencias condicionales en grafos

En este capítulo se estudian los conceptos de separación en grafos tanto dirigidos como no dirigidos, que nos permitirán determinar las relaciones de independencia condicional entre las variables del grafo. Estas relaciones de independencia juegan un papel muy importante ya que son las que permiten factorizar y simplificar la función de probabilidad conjunta que caracteriza un modelo gráfico probabilístico. La siguiente definición se ha tomado de [6, pág. 28].

Definición 20 (Independencia condicional). Sean X , Y y Z variables aleatorias, se dice que X e Y son condicionalmente independientes dado Z si se cumple que:

- $P(X = x, Y = y | Z = z) = P(X = x | Z = z)P(Y = y | Z = z)$, $\forall z$ tal que $P(Z = z) > 0$, si las variables son discretas; o bien
- $f_{X,Y|Z}(x, y | z) = f_{X|Z}(x | z)f_{Y|Z}(y | z)$, si las variables admiten una función de densidad continua respecto a una medida producto.

Dados X , Y y Z tres subconjuntos disjuntos de nodos de un grafo, se utilizará la notación $I(X, Y | Z)$, para expresar la independencia condicional entre X e Y dado Z . En caso de que $Z = \emptyset$, se dirá que la independencia es marginal, y se denotará como $I(X, Y | \emptyset)$; mientras que para indicar la dependencia, se utiliza $D(X, Y | Z)$.

2.1. Separación en grafos no dirigidos. U-separación

Para representar la independencia condicional por medio de grafos no dirigidos se define el criterio de separación gráfica *U-separación*, que permite obtener la lista de relaciones de independencia asociadas a un grafo no dirigido.

Definición 21 (U-separación). Sean X , Y y Z conjuntos disjuntos de nodos en un grafo no dirigido G , entonces Z separa X e $Y \Leftrightarrow$ cada camino entre X e Y contiene algún nodo de Z .

Ejemplo 13 (U-separación). Para ilustrar este concepto la Figura 2.1 muestra mediante 4 ejemplos algunas de las relaciones de dependencia e independencia condicional. Aunque las relaciones que nos interesan realmente son las de independencia, se ven también las de dependencia para mayor claridad del concepto.

1. A y G son condicionalmente independientes dado D , ya que cada camino entre A y G contiene al nodo D .
2. Los subconjuntos $\{A, C\}$ y $\{F, G\}$ son condicionalmente independientes dado el subconjunto $\{B, D\}$ pues cada camino entre los dos subconjuntos contiene al menos un nodo del conjunto $\{B, D\}$.

3. A y G son dependientes dado D , ya que existe al menos un camino $A - B - F - G$ que no contiene al nodo D .
4. Los subconjuntos $\{A, C\}$ y $\{F, G\}$ son dependientes dado el subconjunto $\{D, E\}$, pues el camino $A - B - F - G$ no pasa por ninguno de los nodos D, E .

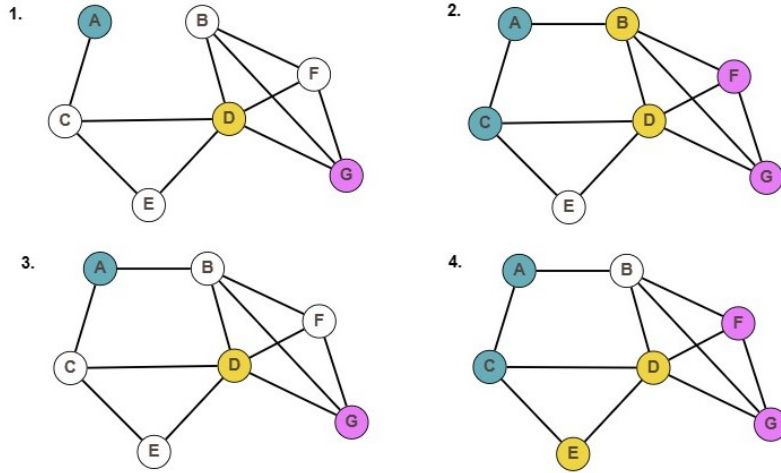


Figura 2.1: Ejemplos de dependencia e independencia condicional en grafos no dirigidos: el grafo 1 representa $I(A, G|D)$, el 2 representa $I(\{A, C\}, \{F, G\} | \{B, D\})$, el 3 $D(A, G|D)$ y por último, el 4 $D(\{A, C\}, \{F, G\} | \{D, E\})$. Los colores azul y rosa corresponden a los nodos pertenecientes a los conjuntos X e Y y el amarillo a los nodos del conjunto separador Z en la notación $I(X, Y|Z)$ y $D(X, Y|Z)$.

2.2. Separación en grafos dirigidos. D-separación

Para representar las relaciones de independencia en un grafo dirigido G , se define el criterio de separación gráfica *D-separación*. Para definir este criterio necesitaremos recordar la Definición 17 de *v-estructura* o *nodo de aristas convergentes*.

En ambos casos, los criterios de separación no solo sirven para representar las relaciones de independencia, sino también son útiles para dado un grafo G , determinarlas.

Definición 22 (D-separación). Sean X, Y y Z conjuntos disjuntos de nodos en un grafo acíclico dirigido G , entonces, se dice que Z *D-separa* X e $Y \Leftrightarrow$ a lo largo de todo camino no dirigido entre cualquier nodo de X y cualquier nodo de Y , existe un nodo intermedio A tal que:

1. A es nodo de aristas convergentes en el camino y ni A ni sus descendientes están en Z , o bien,
2. A no es nodo de aristas convergentes y A está en Z .

Entonces, si se puede encontrar un nodo en algún camino no dirigido que no cumpla ninguna de las condiciones anteriores, se dirá que X e Y son dependientes dado Z .

Ejemplo 14 (D-separación). Para ilustrar estos conceptos utilizamos el grafo de la Figura 1.8 del ejemplo 12, donde vamos a ver algunas de las relaciones de independencia que nos servirán más adelante para simplificar la función de probabilidad. Al igual que para grafos no dirigidos, también veremos las relaciones de dependencia.

1. El nodo L y el conjunto $\{C, V\}$ son condicionalmente independientes dado N , ya que los caminos entre L y $\{C, V\}$ o bien contienen a N que no es de aristas convergentes y está contenido en N

(cumple la condición 2); o bien contiene al nodo I que es de aristas convergentes y ni I ni sus descendientes están contenidos en N (cumple la condición 1).

2. El nodo I es condicionalmente independiente del conjunto $\{V, N\}$ dado los nodos $\{C, L\}$ ya que todo camino entre I y $\{V, N\}$ contiene a los nodos C y L que no son de aristas convergentes y pertenecen a $\{C, L\}$ (cumple la condición 2).
3. Los nodos N e I son dependientes dado L ya que existe un camino $N \rightarrow C \rightarrow I$ que contiene al nodo C que no es de aristas convergentes y no pertenece a L , luego no cumple ninguna de las condiciones.
4. Los conjuntos $\{V, N\}$ y $\{C, L\}$ son dependientes dado I ya que existe un camino entre L y V , $L \rightarrow I \leftarrow C \leftarrow V$ que contiene al nodo I que es de aristas convergentes y pertenece a I , por lo que no cumple ninguna condición.

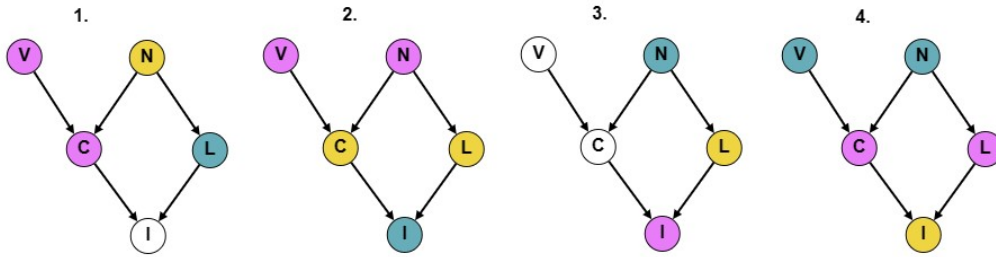


Figura 2.2: Ejemplos de dependencia e independencia condicional en grafos dirigidos: el grafo 1 representa $I(L, \{C, V\} | N)$, el 2 representa $I(I, \{V, N\} | \{C, L\})$, el 3 $D(N, I | L)$ y por último, el 4 $D(\{V, N\}, \{C, L\} | I)$. Los colores azul y rosa corresponden a los nodos pertenecientes a los conjuntos X e Y y el amarillo a los nodos del conjunto separador Z en la notación $I(X, Y | Z)$ y $D(X, Y | Z)$.

5. Por último, los nodos N y V son marginalmente independientes, es decir, $I(N, V | \emptyset)$ ya que todos los caminos entre N y V contienen al nodo C que es un nodo de aristas convergentes y no está contenido en \emptyset .

Las independencias condicionales en un DAG también pueden leerse en su grafo moral asociado (Definición 18) mediante el criterio de U-separación. Por lo que se introduce a continuación este resultado como alternativa a la D-separación que es más fácil de aplicar en la práctica, para esto se introduce la definición previa de *conjunto ancestral*, para la cual se recuerda también la Definición 15 de *ascendientes*.

Definición 23 (Conjunto ancestral). Un conjunto de nodos S se denomina conjunto ancestral si contiene los ascendientes de todos sus nodos.

Proposición 1. Sean X , Y y Z tres subconjuntos disjuntos de nodos en un grafo acíclico dirigido, entonces: Z D-separa X e $Y \Leftrightarrow Z$ separa X e Y en el grafo moral del menor subconjunto ancestral que contenga a los nodos de X , Y y Z .

La demostración de esta proposición se encuentra en [6, pág. 48]. Para ilustrar este concepto se utilizará el grafo y las relaciones condicionales del Ejemplo 14, y se comprobará que las cumple en el Anexo A.1.

Una vez hemos visto cómo se obtienen las relaciones de independencia condicional en cada tipo de grafo, vamos a ver cómo se factoriza la función de probabilidad conjunta asociada a cada uno de estos.

2.3. Modelos gráficos no dirigidos o redes de Markov

Para construir la distribución de probabilidad conjunta asociada a un grafo no dirigido necesitamos recordar el concepto de *clique* dado en la Definición 7.

Definición 24 (Modelo gráfico no dirigido o red de Markov). Una red de Markov o modelo gráfico no dirigido es un par (G, ψ) donde G es un grafo no dirigido definido sobre un conjunto de variables aleatorias $X = \{X_1, \dots, X_n\}$ y $\psi = \{\psi_1, \dots, \psi_m\}$ es un conjunto de funciones no negativas (denominadas funciones potenciales) definidas en los cliques C_1, \dots, C_m de G que permiten definir una función de probabilidad $P(X_1, \dots, X_n)$ por medio de:

$$P(X_1, \dots, X_n) \propto \prod_{i=1}^m \psi_i(C_i). \quad (2.1)$$

Por lo tanto la ecuación (2.1) define una factorización de la función de probabilidad de una red de Markov mediante el producto de factores potenciales. El problema de esta factorización es que las funciones potenciales no tienen por qué ser funciones de probabilidad y la asignación de valores numéricos para la definición del modelo no es sencilla.

Ejemplo 15 (Factorización de un modelo no dirigido). El grafo de la Figura 2.3 es un ejemplo clásico donde su distribución se puede factorizar mediante funciones potenciales pero no mediante funciones de probabilidad. Su factorización mediante la ecuación (2.1) es:

$$P(A, B, C, D) \propto \psi_1(A, B) \psi_2(B, C) \psi_3(C, D) \psi_4(D, A).$$

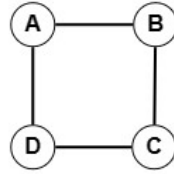


Figura 2.3: Grafo de 4 nodos

Este grafo muestra dos independencias condicionales: $I(A, C | \{B, D\})$ y $I(B, D | \{A, C\})$, si aplicamos la regla de la cadena obtenemos una factorización donde solo se puede aplicar una de las independencias. En caso de tener la factorización $P(A, B, C, D) = P(A|B, C, D)P(B|C, D)P(C|D)P(D)$ solo podríamos utilizar $I(A, C | \{B, D\})$, por lo que la distribución quedaría de la forma:

$$P(A, B, C, D) = P(A|B, D)P(B|C, D)P(C|D)P(D),$$

es decir, la factorización mediante la regla de la cadena, en general, no es equivalente a la factorización mediante funciones potenciales y por tanto estas no se pueden poner como distribuciones de probabilidad condicionada.

En cambio si el grafo es *triangulado* (Definición 10), podemos obtener una factorización como producto de funciones de probabilidad condicionada y dan lugar a los *modelos descomponibles*. Para definirlos necesitamos utilizar la *propiedad de intersección dinámica* introducida en la Definición 12.

Definición 25 (Modelo gráfico descomponible). Sean $\{C_1, \dots, C_m\}$ los cliques de un grafo no dirigido triangulado ordenados de forma que cumplen la propiedad de intersección dinámica. Sean $S_i = C_i \cap (C_1 \cup \dots \cup C_{i-1})$, $i = 2, \dots, m$ los conjuntos separadores. Dado que $S_i \in C_i$ se definen los conjuntos residuales como $R_i = C_i \setminus S_i$. La factorización resultante es de la forma:

$$P(X_1, \dots, X_n) = \prod_{i=1}^m P(R_i | S_i). \quad (2.2)$$

Por lo tanto, en el caso de modelos descomponibles, las funciones potenciales pueden ser definidas mediante funciones de probabilidad condicionada, $\psi_i(C_i) = P(R_i|S_i)$, $i = 1, \dots, m$, donde m es el número de cliques del grafo.

Ejemplo 16 (Factorización de un modelo no dirigido descomponible). Tomando el grafo de la Figura 2.3 y aplicando una triangulación minimal, obtenemos el grafo de la Figura 2.4. Este nuevo grafo mediante la ecuación (2.1) admite la factorización:

$$P(A, B, C, D) \propto \psi_1(A, B, C) \psi_2(B, C, D),$$

que dados los conjuntos separadores $S_1 = \{\emptyset\}$ y $S_2 = \{B, D\}$ y los conjuntos residuales $R_1 = \{A, B, D\}$ y $R_2 = \{C\}$ asociados a cada clique, se puede factorizar mediante la ecuación (2.2) tal que:

$$P(A, B, C) = P(A, B, C)P(C|B, D).$$

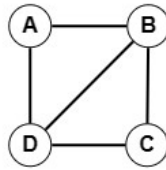


Figura 2.4: Grafo de 4 nodos con la triangulación minimal que se tiene mediante la arista $B - D$.

2.4. Modelos gráficos dirigidos o redes Bayesianas

Definición 26 (Modelo gráfico dirigido o red Bayesiana). Un modelo gráfico dirigido o red Bayesiana es un par (G, P) , donde G es un grafo acíclico dirigido definido sobre un conjunto de variables aleatorias $X = \{X_1, \dots, X_n\}$ y $P = \{P(X_1|\Pi_1), \dots, P(X_n|\Pi_n)\}$ es un conjunto de n funciones de probabilidad condicionada, una para cada variable, donde Π_i representa el conjunto de padres del nodo X_i en G . El conjunto P define una función de probabilidad asociada mediante la factorización

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i|\Pi_i). \quad (2.3)$$

Ejemplo 17 (Riesgo de incendio en una región, continuación). Volviendo al Ejemplo 12 del apartado 1.4, una factorización del modelo mediante la regla de la cadena o de la multiplicación sería:

$$P(V, N, C, L, I) = P(V) P(N|V) P(C|V, N) P(L|V, N, C) P(I|C, L, N, V). \quad (2.4)$$

Si las variables toman valores 0 y 1, para definir el modelo mediante esta factorización necesitaríamos en general $2^n - 1$ parámetros, siendo n el número de nodos del grafo. En este caso, se necesitarían 31 parámetros. Si aplicamos las independencias condicionales que se pueden leer del grafo y que se han obtenido ya en el Ejemplo 14 de D -separación, podemos simplificar la ecuación (2.4) de la forma:

1. Utilizando que $I(N, V|\emptyset)$, reemplazamos el factor $P(N|V) \equiv P(N)$.
2. Utilizando que $I(L, \{C, V\}|N)$, reemplazamos el factor $P(L|C, V, N) \equiv P(L|N)$.
3. Por último, utilizando que $I(I, \{V, N\}|\{C, L\})$, reemplazamos el factor $P(I|V, N, C, L) \equiv P(I|C, L)$.

Obteniendo la factorización vista en la ecuación (2.5) que requiere tan solo 12 parámetros. Un ejemplo de esto se muestra en la Figura 2.5.

$$P(V, N, C, L, I) = P(V) P(N) P(C|V, N) P(L|N) P(I|C, L). \quad (2.5)$$

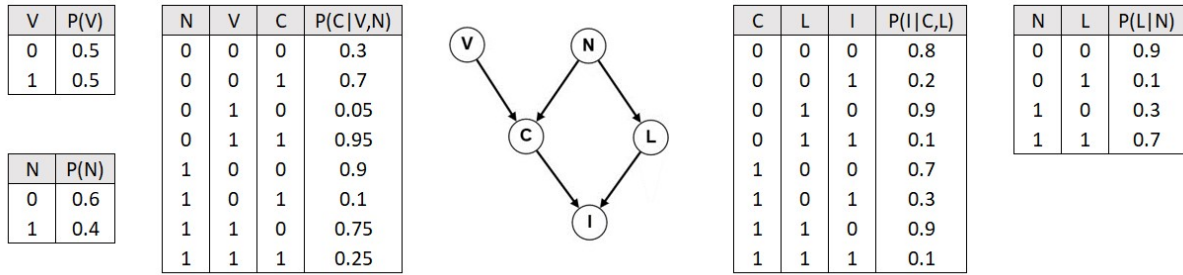


Figura 2.5: Red Bayesiana con las tablas de probabilidad necesarias para definir el modelo completo: las funciones de probabilidad marginal $P(V)$ y $P(N)$ y las funciones de probabilidad condicional $P(C|V,N)$, $P(L|N)$ y $P(I|C,L)$.

Así, en caso de tener una red Bayesiana, la factorización de la función de probabilidad se obtiene de manera directa a partir del grafo dirigido, teniendo en cuenta las distribuciones de probabilidad que relacionan a cada nodo X_i con sus padres Π_i . A diferencia de una red de Markov, que requiere algunos pasos previos.

Para terminar este capítulo vamos a ver qué relación existe entre las estructuras gráficas y el conjunto de relaciones de independencia condicional que representan.

2.5. Modelos gráficos equivalentes

Dos grafos no dirigidos diferentes siempre representan diferentes relaciones de independencia condicional, mientras que dos grafos dirigidos distintos pueden estar asociados al mismo conjunto de independencias condicionales, que determinaría una misma función de probabilidad.

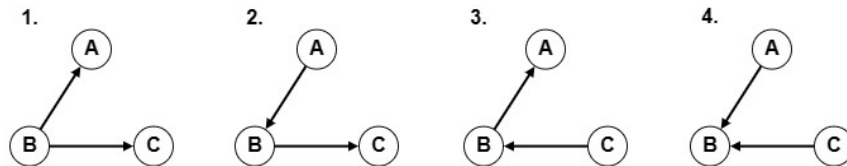


Figura 2.6: Grafos dirigidos de 3 nodos y 2 arcos

Ejemplo 18. En los grafos 1, 2 y 3 tenemos que A y C son independientes dado B , $I(A,C|B)$ por lo que la factorización de la distribución de probabilidad será igual en los tres casos. Sin embargo, el caso del grafo 4, representa un modelo en el que A y C son incondicionalmente independientes, $I(A,C|\emptyset)$.

Definición 27. Dos modelos gráficos se dicen equivalentes si los grafos correspondientes tienen asociado el mismo conjunto de relaciones de independencia condicional.

¿Qué se tiene que cumplir para que 2 grafos dirigidos sean equivalentes? El siguiente teorema nos da una solución a la pregunta.

Teorema 2. Dos grafos dirigidos son equivalentes si tienen asociados tanto el mismo esqueleto como las mismas v -estructuras.

Ejemplo 19. Por lo tanto, en la Figura 2.6, los grafos 1, 2 y 3 representan modelos gráficos equivalentes.

Capítulo 3

Inferencia probabilística

Una de las tareas más importantes de los MGP es la de resolver los problemas de inferencia probabilística. La *inferencia probabilística*, también conocida como *propagación de evidencia*, trata de calcular la probabilidad de una serie de variables dado un conjunto de variables observadas, a las que llamamos *variables evidenciales*. Lo que se utiliza para obtener conclusiones y predicciones basadas en una serie de datos conocidos, denominados *evidencia*.

Definición 28 (Evidencia). La evidencia en un modelo gráfico es un conjunto de nodos $E_v \subset X$ cuyos valores son conocidos, es decir, $E_v = e_v$ en una situación dada.

Existen varios tipos de algoritmos de propagación, en este capítulo nos centraremos en los de propagación exacta y se verán las ventajas e inconvenientes de cada uno de ellos. También existen los algoritmos de propagación aproximada, que se utilizan cuando los exactos no son aplicables o son computacionalmente muy costosos, aunque no los trataremos en este trabajo.

Los problemas de propagación de evidencia requieren resolver algunos problemas como el cálculo de la distribución marginal de un conjunto particular de nodos o la distribución condicional, como por ejemplo cuando se tiene cierta evidencia y se quiere calcular la probabilidad de un nodo dado el conocimiento de esa información, $P(X_i = x_i | E_v = e_v)$.

Es decir, se quiere propagar la evidencia en un conjunto de variables aleatorias $X = \{X_1, \dots, X_n\}$ (en este trabajo se considera además que son discretas), del cual se puede tener evidencia E_v , o no.

- Si no se dispone de evidencia, el proceso de propagación consiste en calcular las probabilidades marginales $P(X_i = x_i)$ para todos los valores posibles de X_i ,

$$P(X_i = x_i) = \sum_{X \setminus X_i} P(X_1, \dots, X_i = x_i, \dots, X_n).$$

- Si se dispone de evidencia $E_v \subset X$, entonces la inferencia consiste en calcular las funciones de probabilidad condicionada $P(X_i = x_i | E_v = e_v)$ para cada valor de la variable X_i y para toda variable X_j de la que no se tenga evidencia, es decir $X_j \notin E_v$,

$$P(X_i = x_i | E_v = e_v) = \frac{P(X_i = x_i, E_v = e_v)}{P(E_v = e_v)} \propto P(X_i = x_i, E_v = e_v),$$

donde siendo $k = \frac{1}{P(E_v = e_v)}$ una constante de normalización, se tiene que $P(X_i = x_i, E_v = e_v)$,

$$P(X_i = x_i, E_v = e_v) = k \sum_{X \setminus X_i, E_v} P_{E_v}(X_1, \dots, X_i = x_i, \dots, X_n).$$

$P_{E_v}(X_1, \dots, X_i = x_i, \dots, X_n)$ es la función de probabilidad conjunta $P(X_1, \dots, X_n)$ en la que se han sustituido las variables con evidencia por sus valores correspondientes.

A continuación se estudian algunos algoritmos de inferencia exacta que tienen en cuenta la estructura de independencia de un grafo. Algunos de estos métodos son aplicables tanto a redes de Markov como a redes Bayesianas, mientras que otros solo se podrán aplicar a redes Bayesianas, ya que también aprovechan la estructura de los grafos dirigidos.

Primero se estudiará el algoritmo de propagación en árboles y poliárboles, su principal inconveniente es que solo sirve cuando las redes tienen esta estructura, por lo que es necesario definir otros tipos de algoritmos como el de condicionamiento, el de agrupamiento o el de agrupamiento en árboles de unión.

3.1. Propagación exacta en poliárboles

La propagación exacta en poliárboles es muy útil cuando se tiene una red Bayesiana con estructura de árbol o poliárbol, ya que permite calcular las distribuciones de manera eficiente y exacta.

En un poliárbol todos los nodos están unidos por un único camino (no necesariamente dirigido), por lo que cada nodo divide el grafo en dos poliárboles inconexos: uno que contiene a los padres y a los nodos a los que está conectado pasando por los padres, y otro que contiene a los hijos y a los nodos que está conectado pasando por los hijos. En el grafo de la Figura 3.1, el nodo D separa el conjunto de los ancestros de D , $\{A, B, E\}$ del de los descendientes de D , $\{C, F, G\}$. El proceso de propagación que se define a continuación combina información procedente de ambos subgrafos mediante el envío de mensajes. Por esta razón el algoritmo también se llama de paso de mensajes (*message-passing*).

Suponer que se tiene un poliárbol formado por los nodos $X = \{X_1, \dots, X_n\}$ y que se conoce la evidencia E_v . La propagación de evidencia consiste en calcular las probabilidades $P(X_i|E_v)$ para todas las variables X_i de las que no se tiene evidencia, es decir, $X_i \notin E_v$. Para facilitar el cálculo se descompone el conjunto E_v en dos subconjuntos disjuntos separados por la variable X_i : el subconjunto E_{vi}^+ que incluye los padres de X_i y los nodos accesibles a través de sus padres y E_{vi}^- que incluye los hijos de X_i y los nodos accesibles a través de sus hijos. Por el *Teorema de Bayes* y considerando que X_i separa E_{vi}^+ y E_{vi}^- se puede calcular $P(X_i|E_v)$ como:

$$\begin{aligned} P(X_i|E_v) &= P(X_i|E_{vi}^+, E_{vi}^-) = k P(X_i) P(E_{vi}^+, E_{vi}^-|X_i) = k P(X_i) P(E_{vi}^+|X_i) P(E_{vi}^-|X_i) \\ &= k P(X_i, E_{vi}^+) P(E_{vi}^-|X_i). \end{aligned}$$

Definiendo las funciones $\lambda_i(X_i) = P(E_{vi}^-|X_i)$ y $\rho_i(X_i) = P(X_i, E_{vi}^+)$, donde $\lambda_i(X_i)$ son los mensajes que tienen en cuenta la evidencia procedente de los hijos de X_i y $\rho_i(X_i)$ son los mensajes que tienen en cuenta la evidencia procedente de los padres de X_i , se obtiene la siguiente expresión:

$$P(X_i|E_v) = k \lambda_i(X_i) \rho_i(X_i). \quad (3.1)$$

La inferencia se realiza mediante un mecanismo de paso de mensajes, en donde cada nodo envía los mensajes correspondientes a sus padres y a sus hijos. Al final de la inferencia, cada nodo tiene asociado un vector λ y un vector ρ , y se obtiene la probabilidad de cada nodo aplicando la ecuación (3.1).

Para calcular los mensajes $\lambda_i(X_i)$ y $\rho_i(X_i)$, consideramos la situación genérica donde se tiene un poliárbol en el que un nodo arbitrario X_i tiene p padres $U = \{U_1, \dots, U_p\}$ y c hijos $Y = \{Y_1, \dots, Y_c\}$. Gracias a la estructura de poliárbol del grafo, los conjuntos de evidencia E_{vi}^+ y E_{vi}^- pueden descomponerse en p y c subconjuntos disjuntos respectivamente:

$$E_{vi}^+ = \{E_{U_1X_i}^+, \dots, E_{U_pX_i}^+\}, \quad E_{vi}^- = \{E_{X_iY_1}^-, \dots, E_{X_iY_c}^-\},$$

donde $E_{U_jX_i}^+$ es el subconjunto de E_{vi}^+ contenido en el subgrafo asociado al nodo U_j cuando se elimina la arista $U_j \rightarrow X_i$ y $E_{X_iY_j}^-$ es el subconjunto de E_{vi}^- contenido en el subgrafo asociado al nodo Y_j cuando se elimina la arista $X_i \rightarrow Y_j$. Y de nuevo se utilizará la estructura de poliárbol que hace que los conjuntos

$\{U_j, E_{U_j X_i}^+\}$ y $\{U_k, E_{U_k X_i}^+\}$ sean condicionalmente independientes para $k \neq j$. Los detalles del cálculo de las funciones ρ_i y λ_i se encuentran en el Anexo A.2.

La descripción completa de la propagación en poliárboles viene dada en el Algoritmo 1, se incluye a continuación un ejemplo aplicado a la red Bayesiana de la Figura 3.1, las soluciones de los cálculos de este ejemplo se incluyen en la Figura 3.2.

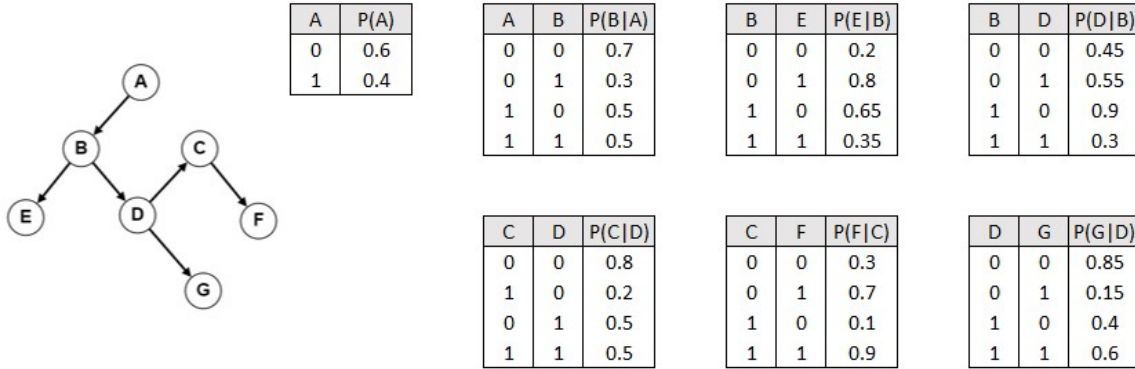


Figura 3.1: Red Bayesiana sobre la que se aplica el algoritmo de propagación en poliárboles.

Ejemplo 20 (Propagación en poliárboles). Consideramos la red Bayesiana de la Figura 3.1, en la que se conoce la evidencia $E_v = \{B = 1, C = 0\}$. Si no se dispone de evidencia, se omite la primera fase de la etapa inicial y se continua con el algoritmo.

■ Etapa de iniciación

1. Definir las funciones λ y ρ para los nodos evidenciales:

$$\begin{aligned} \rho_B(B=0) &= 0, & \lambda_B(B=0) &= 0, & \rho_C(B=0) &= 1, & \lambda_C(B=0) &= 1, \\ \rho_B(B=1) &= 1, & \lambda_B(B=1) &= 1, & \rho_C(B=1) &= 0, & \lambda_C(B=1) &= 0. \end{aligned}$$

2. Asignar a los nodos sin padres la función $\rho_{X_i}(X_i) = P(X_i)$:

$$\begin{aligned} \rho_A(A=0) &= P(A=0) = 0.6, \\ \rho_A(A=1) &= P(A=1) = 0.4. \end{aligned}$$

3. Asignar a todos los nodos sin hijos la función $\lambda_i(X_i) = 1$:

$$\begin{aligned} \lambda_E(E=0) &= 1, & \lambda_F(F=0) &= 1, & \lambda_G(G=0) &= 1, \\ \lambda_E(E=1) &= 1, & \lambda_F(F=1) &= 1, & \lambda_G(G=1) &= 1. \end{aligned}$$

■ Etapa iterativa

Paso 1 (Iteraciones).

- Primera iteración.

Nodo A. Se ha calculado la función $\rho_A(A)$ y B es el único hijo de A, por lo que se puede enviar el mensaje $\rho_{AB}(A) = \rho_A(A)$.

Nodo B. Se tiene $\lambda_B(B)$ y dado que A es su único padre, se puede enviar el mensaje $\lambda_{AB}(A)$ tal que $\lambda_{AB}(A) = \sum_B \lambda_B(B)P(B|A)$: $\lambda_{AB}(A=0) = 0.3$ y $\lambda_{AB}(A=1) = 0.5$.

Nodo C. Se tiene $\lambda_E(E)$ y su único padre es B, por lo que puede enviar los mensajes $\lambda_{EB}(B)$ tal que $\lambda_{EB}(B) = \sum_E \lambda_E(E)P(E|B)$: $\lambda_{EB}(B=0) = 1$ y $\lambda_{EB}(B=1) = 1$.

Nodo D. En el nodo D ni se ha recibido ni se han calculado las funciones λ o ρ , por lo que no se puede efectuar ninguna operación en esta iteración.

Nodo C. Se ha calculado $\lambda_C(C)$ y su único padre es D por lo que de manera análoga a los nodos B y E se pueden calcular los mensajes $\lambda_{CD}(D=0)=0.8$ y $\lambda_{CD}(D=1)=0.5$. Además, para este nodo también se ha calculado la función $\rho_C(C)$, y su único hijo es F , por lo que se pueden enviar los mensajes $\rho_{CF}(C)=\rho_C(C)$: $\rho_{CF}(C=0)=1$ y $\rho_{CF}(C=1)=0$.

Nodos G y F. Para los nodos G y F se tienen las funciones λ y ambos tienen un único padre, luego de forma análoga a los casos anteriores se pueden calcular: $\lambda_{GD}(D=0)=1$, $\lambda_{GD}(D=1)=1$, $\lambda_{FC}(C=0)=1$, y $\lambda_{FC}(C=1)=1$.

Nodo F. Además, F ha recibido los mensajes ρ de todos sus padres, luego se puede calcular $\rho_F(F)$ tal que $\rho_F(F)=\sum_C P(F|C)\rho_{CF}(C)$, entonces: $\rho_F(F=0)=0.3$ y $\rho_F(F=1)=0.7$.

- *Segunda iteración.*

Nodo A. El nodo A ha recibido los mensajes λ de todos sus hijos, por lo que se puede calcular $\lambda_A(A)=\lambda_{BA}(A)$, entonces: $\lambda_A(A=0)=0.3$ y $\lambda_A(A=1)=0.5$.

Nodo B. Se tiene $\rho_B(B)$ y B ha recibido el mensaje λ_{EB} de su hijo E , por lo que puede enviar a su otro hijo D el mensaje $\rho_{BD}(B)$ tal que $\rho_{BD}(B)=\rho_B(B)\lambda_{EB}(B)$, es decir: $\rho_{BD}(B=0)=0$ y $\rho_{BD}(B=1)=1$.

Nodo D. El nodo D ha recibido el mensaje ρ_{BD} de su único padre, por lo que se puede calcular su función $\rho_D(D)=\sum_B P(D|B)\rho_{BD}(B)$, y se tiene: $\rho_D(D=0)=0.9$ y $\rho_D(D=1)=0.1$.

Análogamente se calculan también en este paso los mensajes $\rho_{DC}(D)$, $\rho_{DG}(D)$, $\lambda_{DB}(B)$ y la función $\lambda_D(D)$.

- *Tercera iteración.*

Nodo B. Se ha calculado la función $\rho_B(B)$ y B ha recibido λ_{DB} de su hijo D , luego puede enviar el mensaje ρ_{BE} a su otro hijo E tal que $\rho_{BE}(B)=\rho_B(B)\lambda_{DB}(B)$, y se tiene: $\rho_{BE}(B=0)=0$ y $\rho_{BE}(B=1)=0.77$.

Nodo E. Para terminar esta iteración solo queda calcular las funciones $\rho_E(E)$ y $\rho_G(G)$, de forma que $\rho_E(E)=\sum_{b=0}^1 P(E|B)\rho_{BE}(B)$, y se tiene: $\rho_E(E=0)=0.5005$ y $\rho_E(E=1)=0.2695$.

Nodo G. Análogamente para el nodo G se tienen: $\rho_G(G=0)=0.632$ y $\rho_G(G=1)=0.138$.

Por lo que ya tenemos las funciones necesarias para calcular las probabilidades asociadas a cada nodo, lo siguiente es pasar a los Pasos 2 y 3, multiplicando $\lambda_i(X_i) \cdot \rho_i(X_i)$, $\forall X_i \notin E_v$ y normalizando el resultado para obtener $P(X_i|E_v)$. Estos resultados se muestran en la Figura 3.2.

La ventaja de este algoritmo es que el número de operaciones aumenta linealmente con el tamaño de la red, y su principal inconveniente es que no se aplica a redes múltiplemente conectadas ya que carece de generalidad, para este tipo de estructura de redes se presentan a continuación otros algoritmos. Dos de los métodos más importantes son los *métodos de condicionamiento* y los *métodos de agrupamiento*.

3.2. Métodos de condicionamiento

A diferencia de los poliárboles, las redes múltiplemente conexas se caracterizan por tener más de un camino entre dos o más nodos, es decir, de la existencia de bucles (Definición 5) en el grafo no dirigido asociado (Definición 16). La idea principal de los métodos de condicionamiento es cortar los múltiples caminos existentes entre dos nodos mediante la asignación de valores a un conjunto de variables contenidas en los bucles, este conjunto de variables se denomina *conjunto de corte*, $C = \{C_1, \dots, C_m\}$. Es decir, se puede cortar un bucle considerando como *nodo evidencial* a un nodo o un conjunto de nodos contenidos en él, y así transformar el grafo en un poliárbol donde aplicar el Algoritmo 1.

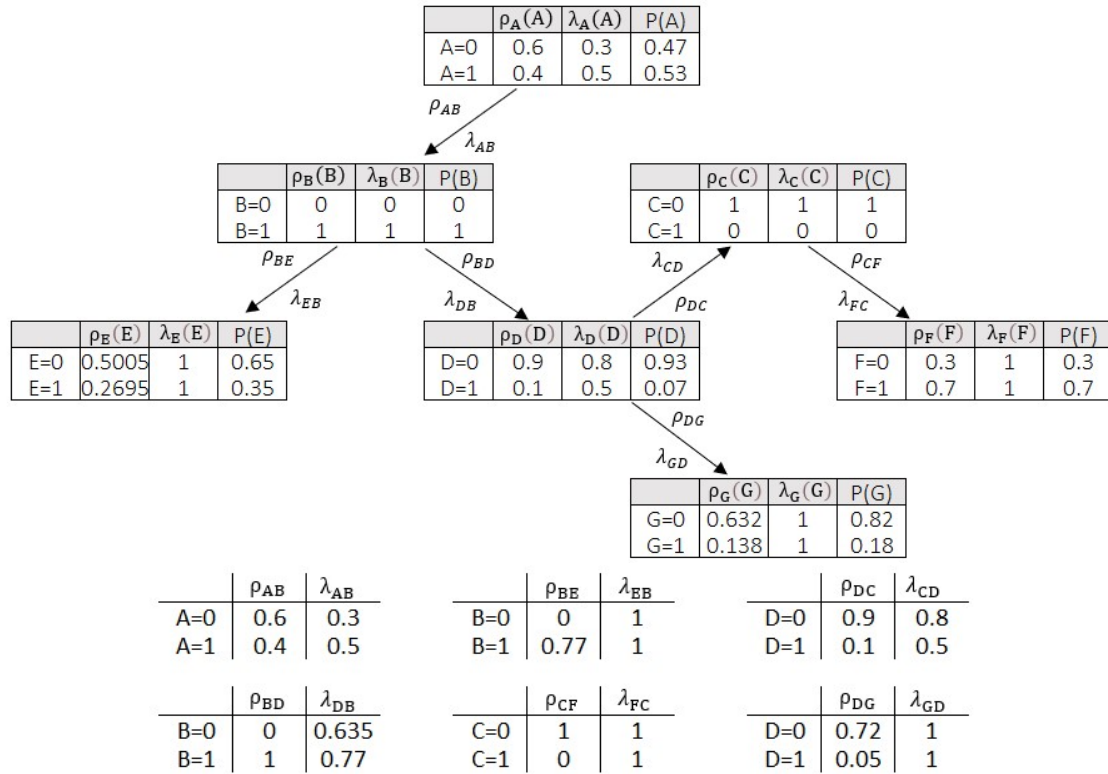


Figura 3.2: Funciones y mensajes obtenidos aplicando el algoritmo de propagación de evidencia en poliárboles (Algoritmo 1) a la red Bayesiana definida por la Figura 3.1.

Si por ejemplo consideramos X_i como nodo evidencial, se pueden eliminar del grafo todas las aristas de la forma $X_i \rightarrow X_j$, y para que la red Bayesiana resultante sea equivalente a la inicial, se sustituye la función de probabilidad condicionada del nodo X_j por una función definida sobre un conjunto más reducido de variables (donde se excluyen las variables que hemos considerado evidenciales):

$$P_1(X_j | \Pi_j \setminus X_i) = P(X_j | \Pi_j \setminus X_i, X_i = x_i).$$

La función de probabilidad $P(X_i | E_v)$, se obtiene promediando las probabilidades ponderadas obtenidas para cada valor posible de las variables del conjunto de corte:

$$\begin{aligned} P(X_i | E_v) &= \sum_{C_1, \dots, C_m} P(X_i | E_v, C_1, \dots, C_m) P(C_1, \dots, C_m | E_v) \\ &= k \sum_{C_1, \dots, C_m} P(X_i | E_v, C_1, \dots, C_m) P(E_v | C_1, \dots, C_m) P(C_1, \dots, C_m), \end{aligned}$$

siendo $k = \frac{1}{P(E_v)}$ la constante de normalización, y donde los dos primeros productos se pueden calcular mediante el Algoritmo 1 y el tercero asignando valores secuencialmente a los nodos de forma que solo sea necesaria una parte del grafo que tenga estructura de poliárbol para calcularlo.

Ejemplo 21. La función de probabilidad conjunta de la red Bayesiana de la Figura 3.3 es:

$$P(A, B, C, D, E, F, G) = P(A)P(B|A)P(E|B)P(D|B)P(C|A, D)P(F|C)P(G|D).$$

El bucle formado por los nodos A, B, C y D se puede romper considerando A como nodo evidencial, y eliminando o la arista $A \rightarrow B$ o la arista $A \rightarrow C$, aunque no es la única elección posible. Se dice que A absorbe la evidencia $A = a$. Sin pérdida de generalidad, se elimina la arista $A \rightarrow C$ (ya que así obtenemos

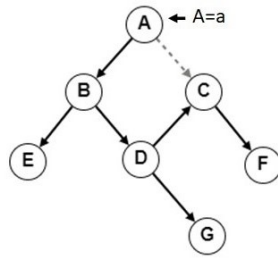


Figura 3.3: Ejemplo de red Bayesiana múltiplemente conexa. En este caso se toma como nodo evidencial la variable A y se elimina la arista $A \rightarrow C$.

el poliárbol sobre el que hemos aplicado el Algoritmo 1). Y se sustituye $P(C|D, A = a) = P_1(C|D)$. Entonces, la función de probabilidad conjunta resultante es:

$$P(A, B, C, D, E, F, G | A = a) = P(A = a)P(B|A = a)P(E|B)P(D|B)P_1(C|D)P(F|C)P(G|D).$$

Suponiendo como en el apartado anterior que se conoce $E_v = \{B = 1, C = 0\}$, la propagación consiste en calcular las probabilidades $P(X_i | A = a, B = 1, C = 0)$, para todos los posibles valores de A (0 y 1, en este caso). Este cálculo viene dado por:

$$P(X_i | B = 1, C = 0) = \sum_{a=0}^1 P(X_i | A = a, B = 1, C = 0)P(B = 1, C = 0 | A = a)P(A = a),$$

donde el producto $P(X_i | A = a, B = 1, C = 0)$ se puede calcular aplicando el Algoritmo 1 considerando la evidencia $E_v = \{A = a, B = 1, C = 0\}$, $P(A = a)$ se puede obtener directamente de la Figura 3.1, y por último el producto $P(B = 1, C = 0 | A = a)$ al no ser una función de un solo nodo no puede obtenerse directamente aplicando el Algoritmo 1, pero se puede descomponer mediante la regla de la cadena de la forma:

$$P(B = 1, C = 0 | A = a) = P(C = 0 | B = 1, A = a)P(B = 1 | A = a),$$

donde el primer producto se puede obtener aplicando el Algoritmo 1, para los dos posibles valores de A y $P(B = 1 | A = a)$ se se obtiene directamente de la Figura 3.1.

El principal inconveniente de este método es que el número de valores posibles que puede tomar el conjunto de corte aumenta exponencialmente con el número de nodos que lo forman. Por esto es importante seleccionar el menor conjunto de corte posible.

3.3. Métodos de agrupamiento

Estos métodos son los más comunes de propagación de evidencia en redes múltiplemente conexas. Además, a diferencia de los anteriores, son aplicables tanto a redes de Markov como a redes Bayesianas, aunque para redes Bayesianas se necesitarán algunas transformaciones previas. Consisten en transformar la estructura de la red para obtener un poliárbol. Para esta transformación, el algoritmo se basa en la construcción de cliques (Definición 7) que mantienen las estructuras locales del MGP. El proceso de propagación calcula las probabilidades locales en los cliques (que dependen de un número reducido de variables), evitando así calcular las probabilidades globales (que dependen de todas las variables).

3.3.1. Métodos de agrupamiento en redes de Markov

Estos métodos se aplican sobre redes de Markov descomponibles (Definición 25), además, en caso de aplicarse a un grafo no triangulado (Definición 10), este se puede triangular sin pérdida de generalidad, ya que los cliques del grafo triangulado contendrán a los del grafo inicial.

El algoritmo consta de los siguientes pasos: en la primera etapa se obtiene una factorización del modelo de la forma $P(X_1, \dots, X_n) = \prod_{i=1}^m P(R_i|S_i)$ dada en la Definición 25. En la segunda etapa, se calculan las funciones de probabilidad conjunta $P(R_i, S_i)$ asociadas a cada clique. Y en la última etapa se calculan las probabilidades de cada nodo $P(X_i|E_v)$ dada la evidencia. El método se aplica mediante el Algoritmo 2.

Para la absorción de evidencia en caso de conocerla, se modifican las funciones potenciales que contengan nodos evidenciales. Es decir, para cada clique que contenga algún nodo evidencial, se define su función potencial como:

$$\psi_i^*(C_i) = \begin{cases} 0, & \text{si algún valor de } C_i \text{ no es consistente con } E_v. \\ \psi_i(C_i), & \text{en otro caso.} \end{cases} \quad (3.2)$$

Para los cliques tales que no se conoce evidencia de ninguno de sus nodos, no se hace ningún cambio. Por lo tanto, la función de probabilidad conjunta dada la evidencia es equivalente a la siguiente expresión:

$$P(X_1, \dots, X_n|E_v) \propto \psi_1^*(C_1) \cdots \psi_m^*(C_m).$$

Este algoritmo se ilustrará con un ejemplo en el siguiente apartado.

3.3.2. Métodos de agrupamiento en redes Bayesianas

El algoritmo anterior a priori solamente es aplicable a redes de Markov, pero nos sirve de base para su implementación en redes Bayesianas, ya que en el caso de tratarse de modelos dirigidos, el procedimiento es análogo pero se requiere de algunas modificaciones previas.

Estas modificaciones consisten en transformar el grafo dirigido en un grafo no dirigido moralizado (Definición 18) y triangulado, ya que esto no altera la estructura de independencia de la red (pues cada familia del grafo dirigido estará contenida en algún clique del grafo triangulado) y asociar a la factorización de la red Bayesiana mediante funciones de probabilidad $P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i|\Pi_i)$, una factorización mediante funciones potenciales $P(X_1, \dots, X_n) \propto \prod_{i=1}^m \psi_i(C_i)$.

En resumen, los pasos a llevar a cabo para la transformación de la red Bayesiana en una red de Markov descomponible con representación potencial ψ son:

1. Moralizar y triangular el grafo dirigido.
2. Obtener una cadena de cliques $C = \{C_1, \dots, C_m\}$ que cumpla la propiedad de intersección dinámica.
3. Asignar a cada nodo X_i un único clique que contenga a sus familias. Sea A_i el conjunto de nodos asociados al clique C_i .
4. Para cada clique C_i definir $\psi_i(C_i) = \prod_{X_i \in A_i} P(X_i|\Pi_i)$. Si $A_i = \emptyset$, definir $\psi_i(C_i) = 1$.

Y a la red resultante se le puede aplicar el Algoritmo 2.

Ejemplo 22 (Método de agrupamiento en redes Bayesianas). *Consideramos la red Bayesiana definida en el Ejemplo 17 y vamos a calcular el riesgo de incendio en el norte en verano, es decir, se tiene la evidencia $E_v = \{V = 1, N = 1\}$. La factorización de la función de probabilidad es:*

$$P(V, N, C, L, I) = P(V)P(N)P(C|N, V)P(L|N)P(I|C, L).$$

Moralizando y triangulando el grafo dirigido se obtiene el grafo de la Figura 3.4, con los cliques C_1, C_2 y C_3 , que cumplen la propiedad de intersección dinámica. Por lo tanto, lo siguiente es obtener una

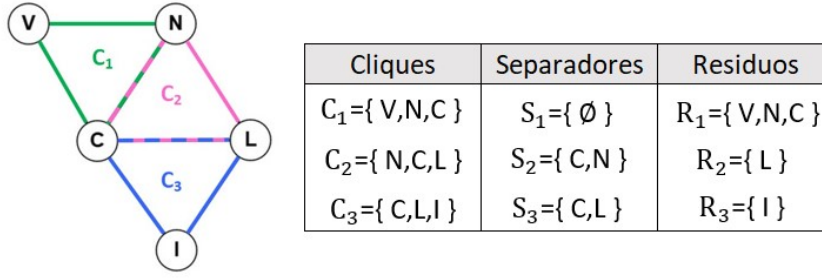


Figura 3.4: Grafo no dirigido moralizado y triangulado asociado a la red Bayesiana del Ejemplo 17, con los separadores y residuos asociados a cada clique.

representación mediante funciones potenciales tal que:

$$P(V, N, C, L, I) = \psi_1(C_1) \psi_2(C_2) \psi_3(C_3) = P(R_1|S_1)P(R_2|S_2)P(R_3|S_3) \\ = P(V, N, C)P(L|C, N)P(I|C, L).$$

Las funciones potenciales en cada clique C_i se obtienen de la siguiente forma:

$$\begin{aligned} \psi_1(C_1) &= P(V) P(N) P(C|N, V), \\ \psi_2(C_2) &= P(L|C, N) \equiv P(L|C), \text{ ya que } I(L, C|N) \text{ en el grafo dirigido original,} \\ \psi_3(C_3) &= P(I|C, L). \end{aligned}$$

El cálculo de estas funciones potenciales se muestra en la Figura 3.5.

N	V	C	$\psi_1(V, N, C)$
0	0	0	0.09
0	0	1	0.21
0	1	0	0.015
0	1	1	0.285
1	0	0	0.18
1	0	1	0.02
1	1	0	0.15
1	1	1	0.05

N	L	$\psi_2(N, L)$
0	0	0.9
0	1	0.1
1	0	0.3
1	1	0.7

C	L	I	$\psi_3(C, L, I)$
0	0	0	0.8
0	0	1	0.2
0	1	0	0.9
0	1	1	0.1
1	0	0	0.7
1	0	1	0.3
1	1	0	0.9
1	1	1	0.1

Figura 3.5: Valores de las funciones potenciales que definen la red de Markov de la Figura 3.4.

Y ya tenemos la red de Markov descomponible equivalente y las funciones potenciales que definen el MGP donde llevar a cabo la inferencia aplicando el Algoritmo 2.

■ Etapa de iniciación

1. Absorción de la evidencia $E_v = \{V = 1, N = 1\}$ como se indica en la ecuación (3.2), por lo tanto, la probabilidad conjunta dado que $N = 1$ y $V = 1$, viene dada por:

$$P(V, N, C, L, I) \propto \psi_1^*(V, N, C) \psi_2^*(N, L) \psi_3^*(C, L, I), \quad (3.3)$$

y las funciones $\psi_i^*(C_i)$ se muestran en la Figura 3.6.

2. Los cliques C_1 , C_2 y C_3 forman una cadena de cliques.
3. Se eligen vecinos para cada clique de manera que C_1 es vecino de C_2 y C_2 es vecino de C_3 .

N	V	C	$\psi_1^*(V, N, C)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0.15
1	1	1	0.05

N	L	$\psi_2^*(N, L)$
0	0	0
0	1	0
1	0	0.3
1	1	0.7

C	L	I	$\psi_3^*(C, L, I)$
0	0	0	0.8
0	0	1	0.2
0	1	0	0.9
0	1	1	0.1
1	0	0	0.7
1	0	1	0.3
1	1	0	0.9
1	1	1	0.1

Figura 3.6: Valores de las funciones potenciales que definen la red dada la evidencia de la ecuación (3.3).

■ Etapa iterativa

Paso 1. Cálculo de $P(I|C, L)$, $P(L|N)$, $P(V, N, C)$.

- Para el cálculo de $P(R_3|S_3) = P(I|C, L)$ se tiene por definición, $P(I|C, L) = \psi_3^*(C, L, I)$.
- Lo mismo sucede para $P(R_2|S_2) = P(L|N) = \psi_2^*(N, L)$.
- Para $P(R_1|S_1) = P(V, N, C)$, también se tiene que $P(V, N, C) = P(V)P(N|V)P(C|N, V) \equiv P(V)P(N)P(C|N, V) = \psi_1^*(V, N, C)$.

Paso 2. Cálculo de las distribuciones de probabilidad en cada clique C_i .

- En el clique C_1 ya se tiene $P(C_1) = P(V, N, C)$.
- Para el cálculo de $P(C_2) = P(C, N, L)$, calcular primero $P(C, N)$ marginalizando sobre la función $P(V, N, C)$ tal que $P(C, N) = \sum_V P(V, N, C)$, y se tiene:

$$P(0, 0) = 0, \quad P(0, 1) = 0.15, \quad P(1, 0) = 0, \quad P(1, 1) = 0.05.$$

Entonces, $P(C_2) = P(N, C, L) = P(L|N)P(C, N)$, y normalizando, el resultado, se muestra en la Figura 3.7.

- Para el cálculo de $P(C_3) = P(C, L, I)$, calcular primero $P(C, L)$ marginalizando sobre la función $P(N, C, L)$ tal que $P(C, L) = \sum_N P(N, C, L)$:

$$P(0, 0) = 0.225, \quad P(0, 1) = 0.525, \quad P(1, 0) = 0.075, \quad P(1, 1) = 0.0175.$$

Entonces, $P(C_3) = P(C, L, I) = P(I|C, L)P(C, L)$, y normalizando, su resultado se muestra en la Figura 3.7

N	V	C	$P(N, V, C)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0.75
1	1	1	0.25

C	N	L	$P(N, C, L)$
0	0	0	0
0	0	1	0
0	1	0	0.225
0	1	1	0.525
1	0	0	0
1	0	1	0
1	1	0	0.075
1	1	1	0.175

C	L	I	$P(C, L, I)$
0	0	0	0.18
0	0	1	0.045
0	1	0	0.0675
0	1	1	0.0075
1	0	0	0.3675
1	0	1	0.1575
1	1	0	0.1575
1	1	1	0.0175

Figura 3.7: Funciones de probabilidad en cada clique.

Paso 3. Cálculo de la función de probabilidad de cada nodo. Una vez tenemos las distribuciones de cada clique, solo queda marginalizar las funciones en cada nodo y se obtienen las funciones de probabilidad de la Figura 3.8

V	P(V)	N	P(N)	C	P(C)	L	P(L)	I	P(I)
0	0	0	0	0	0.75	0	0.3	0	0.77
1	1	1	1	1	0.25	1	0.7	1	0.23

Figura 3.8: Funciones de probabilidad en cada nodo.

En conclusión, el riesgo de que haya un incendio en el norte en verano, es decir $P(I = 1 | V = 1, N = 1)$, es del 23%. Notar además que al tratarse de una red Bayesiana, se simplifican muchos de los cálculos, como los del Paso 1, lo que no ocurriría en una red de Markov. Aquí otro ejemplo de las ventajas de un modelo dirigido frente a uno no dirigido.

Una modificación de estos métodos de agrupamiento que mejora la eficacia de la inferencia son los métodos de agrupamiento basados en *árboles de unión*.

3.3.3. Propagación en árboles de unión

Este método de agrupamiento consiste en la transformación de la estructura de la red para obtener una estructura de árbol de unión donde llevar a cabo la inferencia de manera más efectiva. La propagación se realiza mediante el envío de mensajes sobre el árbol obtenido.

Definición 29 (Árbol de unión o árbol de cliques). En un árbol de unión o árbol de cliques, cada nodo representa un clique del grafo sobre el que está construido, y los nodos están unidos por un camino si los cliques que representan tienen algún nodo en común.

Para construir un árbol de unión se parte de un grafo no dirigido triangulado G , el primer paso es obtener una cadena de cliques C_1, \dots, C_m (Definición 12). Después, para cada clique C_i , se selecciona el clique C_k con el máximo número de nodos comunes a C_i y se añade la arista $C_i - C_k$. En caso de que existan varios cliques con el mismo número de nodos en común, la elección se realiza de manera arbitraria.

La principal ventaja de utilizar un árbol de unión frente a un grafo de cliques, es que el árbol contiene un menor número de aristas, por lo que el número de mensajes a calcular y propagar es menor, lo que hace que el proceso de inferencia sea más eficiente.

Al igual que los métodos de agrupamiento anterior, sirve tanto para redes de Markov como para redes Bayesianas, y en el caso de tener una red Bayesiana, se deben hacer las mismas transformaciones previas para obtener la estructura de grafo no dirigido triangulado y la representación potencial ψ . Luego sin pérdida de generalidad podemos suponer que se tiene un grafo no dirigido triangulado con representación potencial $\psi = \{\psi_1, \dots, \psi_m\}$. La propagación se lleva a cabo en forma de mensajes y se propaga por el árbol mediante el envío de estos entre los cliques vecinos. Este método se aplica mediante el Algoritmo 3 y los detalles de este algoritmo se encuentran en el Anexo A.3.

Ejemplo 23 (Propagación mediante árboles de unión en RB). Consideramos la red del Ejemplo 17 para contrastar este algoritmo con el anterior, ya que ambos son aplicables a redes con la misma estructura. Para poder aplicar el Algoritmo 3, el primer paso es transformar la RB en un árbol de unión, elegimos los mismos cliques que en la Figura 3.4, por lo que las funciones potenciales del árbol serán las ya calculadas en la Figura 3.5, y tomando la misma evidencia que antes, nos lleva a los valores de la Figura 3.6. Existe más de un árbol de unión asociado a esta red, pero vamos a utilizar el que se muestra en la Figura 3.9. Por lo tanto ya tenemos el árbol de unión de la red de Markov y sus funciones potenciales.

■ Etapa de iniciación

1. Se absorbe de nuevo la evidencia $E_v = \{V = 1, N = 1\}$ como se muestra en la Figura 3.6.

2. Se obtiene el árbol de unión de la Figura 3.9, que define una red de Markov con las funciones potenciales ψ_i^* de la Figura 3.5.

■ **Etapa iterativa**

Paso 1. Cálculo de los mensajes $M_{ij}(S_{ij})$.

• *Primera iteración.*

- 1) C_1 solo tiene un vecino C_2 , por lo que puede calcular y enviar el mensaje $M_{12}(S_{12}) = M_{12}(N, C) = \sum_V \psi_1^*(V, N, C)$:

$$\begin{aligned} M_{12}(0, 0) &= 0, & M_{12}(1, 0) &= 0.15, \\ M_{12}(0, 1) &= 0, & M_{12}(1, 1) &= 0.05. \end{aligned}$$

- 2) C_2 ha recibido mensaje del resto de nodos (en este caso solo C_1), excepto de C_3 , por lo tanto, puede calcular y enviar

$$M_{23}(C, L) = \sum_{C_2 \setminus S_{23}} \psi_2^*(C_2) \prod_{k \neq 3} M_{k2}(S_{k2}) = \sum_N \psi_2^*(N, C, L) M_{12}(N, C).$$

$$\begin{aligned} M_{23}(0, 0) &= 0.025, & M_{23}(1, 0) &= 0.075, \\ M_{23}(0, 1) &= 0.525, & M_{23}(1, 1) &= 0.175. \end{aligned}$$

- 3) C_3 solo tiene como vecino a C_2 , luego puede calcular y enviar $M_{32}(C, L) = \sum_I \psi_3^*(I, C, L)$, y se tiene:

$$M_{32}(0, 0) = M_{32}(0, 1) = M_{32}(1, 0) = M_{32}(1, 1) = 1.$$

• *Segunda iteración.*

- 1) El clique C_2 ha recibido los mensajes de todos sus cliques vecinos, luego puede enviar el único mensaje que falta, $M_{21}(N, C) = \sum_L \psi_2^*(N, C, L) M_{32}(C, L)$, y también se tiene:

$$\begin{aligned} M_{21}(0, 0) &= 0, & M_{21}(1, 0) &= 1, \\ M_{21}(0, 1) &= 0, & M_{21}(1, 1) &= 1. \end{aligned}$$

Paso 2. Cálculo de la función de probabilidad de cada clique.

- 1) $P(C_1) = P(V, N, C) = \psi_1^*(V, N, C) M_{21}(N, C)$.
 2) $P(C_2) = P(N, C, L) = \psi_2^*(N, C, L) M_{12}(N, C) M_{32}(C, L)$.
 3) $P(C_3) = P(C, L, I) = \psi_3^*(C, L, I) M_{23}(C, L)$.

En el Paso 2 se obtienen las mismas funciones de probabilidad que se han calculado con el algoritmo de agrupamiento y que se muestran en la Figura 3.7, por lo tanto, la distribución de probabilidad de cada nodo será igual a la que se muestra en la Figura 3.8, y se llega a la misma solución.

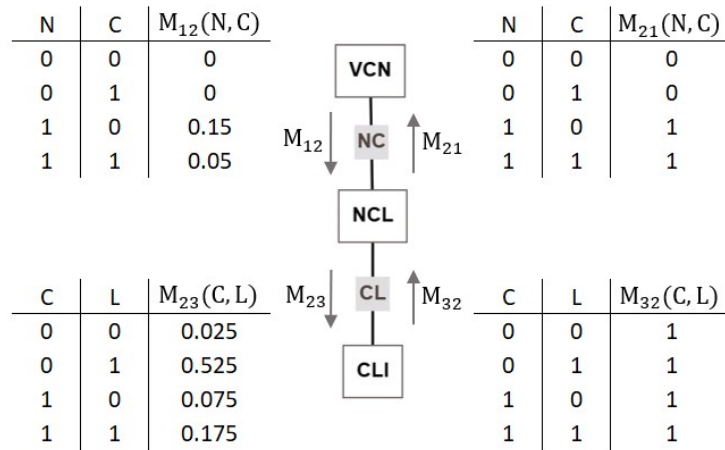


Figura 3.9: Árbol de unión y los mensajes enviados.

Algoritmo 1: Propagación en poliárboles

■ Etapa de iniciación:

1. Asignar a todos los nodos evidenciales las funciones:
 - $\rho_i(X_i) = 1$ si $x_i = e_i$ y $\rho_i(X_i) = 0$ si $x_i \neq e_i$.
 - $\lambda_i(X_i) = 1$ si $x_i = e_i$ y $\lambda_i(X_i) = 0$ si $x_i \neq e_i$.
2. Asignar a todos los nodos $X_i \notin E_v$ que no tengan padres la función $\rho_i(X_i) = P(X_i)$.
3. Asignar a todos los nodos $X_i \notin E_v$ que no tengan hijos la función $\lambda_i(X_i) = 1$, para todos los valores posibles de X_i .

■ Etapa iterativa

Paso 1 (Iteraciones). Para cada nodo $X_i \notin E_v$:

- a) Si X_i ha recibido los mensajes $\rho_{U_j X_i}(U_j)$ de todos sus padres, calcular $\rho_i(X_i)$:

$$\rho_i(X_i) = \sum_U P(X_i | U \cup E_{vi}^+) \prod_{j=1}^p \rho_{U_j X_i}(U_j).$$

- b) Si X_i ha recibido los mensajes $\lambda_{Y_j X_i}(X_i)$ de todos sus hijos, calcular $\lambda_i(X_i)$:

$$\lambda_i(X_i) = \prod_{j=1}^c \lambda_{Y_j X_i}(X_i).$$

- c) Si ya se ha calculado $\rho_i(X_i)$, entonces, para cada hijo Y_j de X_i tal que X_i haya recibido los mensajes λ del resto de sus hijos, calcular y enviar $\rho_{X_i Y_j}(X_i)$:

$$\rho_{X_i Y_j}(X_i) \propto \rho_i(X_i) \prod_{k \neq j} \lambda_{Y_k X_i}(X_i).$$

Entonces, si X_i ha recibido los mensajes λ de todos sus hijos, puede enviar todos los mensajes ρ correspondientes.

- d) Si ya se ha calculado $\lambda_i(X_i)$, entonces, para cada padre U_j de X_i tal que X_i haya recibido los mensajes ρ del resto de sus padres, calcular y enviar $\lambda_{X_i U_j}(U_j)$:

$$\lambda_{Y_j X_i}(X_i) = \sum_{Y_j} \lambda_{Y_j}(Y_j) \sum_{S_1, \dots, S_q} P(Y_j | S_i) \prod_{k=1}^q \rho_{S_k Y_j}(S_k).$$

Entonces, si X_i ha recibido los mensajes ρ de todos sus padres, ya puede enviar todos los mensajes λ correspondientes.

Paso 2. Repetir el paso 1 hasta que se calculen las funciones ρ y λ de todos los nodos no evidenciales.

Paso 3. Para cada nodo $X_i \notin E_v$, calcular $P(X_i | E_v)$ mediante la ecuación (3.1) donde

$$k = \frac{1}{\sum_{X_i} \lambda_i(X_i) \rho_i(X_i)}.$$

Algoritmo 2: Algoritmo de agrupamiento en redes de Markov

■ Etapa de iniciación

1. Absorber la evidencia $E_v = e_v$ en las funciones potenciales.
2. Obtener una cadena de cliques C_1, \dots, C_m que cumpla la propiedad de intersección dinámica.
3. Para cada clique C_i elegir como vecino cualquier otro clique C_j , con $j < i$, tal que $S_i \subset C_j$.

■ Etapa iterativa

Paso 1. Para $i = m$ hasta 1:

- a) Calcular $m_i(S_i) = \sum_{r_i} \psi_i(C_i)$.
- b) Asignar $P(R_i|S_i) = \frac{\psi_i(C_i)}{m_i(S_i)}$.
- c) Reemplazar la función $\psi_j(C_j)$, con C_j vecino de C_i por: $\psi_j(C_j) \leftarrow \psi_j(C_j)m_i(S_i)$.

Paso 2. Asignar $P(C_1) = P(R_1|S_1) = P(R_1)$, y para $i = 2$ hasta m hacer:

- a) Calcular $P(S_i)$ marginalizando $P(C_j)$ del clique C_j , vecino de C_i .
- b) Asignar $P(C_i) = P(R_i|S_i) = P(R_i)$.

Paso 3. Para $i = 1$ hasta n hacer:

- a) Elegir el clique de menor tamaño C_j que contenga al nodo X_i .
 - b) Calcular las probabilidades de cada nodo marginalizando $P(X_i|E_v) \propto \sum_{C_j \ni X_i} P(C_j)$.
 - c) Normalizar los valores obtenidos en b).
-

Algoritmo 3: Propagación mediante árboles de unión en redes de Markov

■ Etapa de iniciación

1. Absorber la evidencia $E_v = e_v$ en las funciones potenciales ψ como en (3.2).
2. Obtener un árbol de unión de la red de Markov.

■ Etapa iterativa

Paso 1 (Iteraciones). Para $i = 1, \dots, m$ hacer: Para cada vecino B_j de C_i , si C_i ha recibido los mensajes del resto de sus vecinos, calcular y enviar el mensaje $M_{ij}(S_{ij})$ a B_j :

$$M_{ij}(S_{ij}) = \sum_{C_i \setminus S_{ij}} \psi_i(C_i) \prod_{k \neq j} M_{ki}(S_{ki}).$$

Y repetir este paso hasta que no se obtenga ningún nuevo mensaje.

Paso 2. Calcular la función de probabilidad de cada clique C_i :

$$P(C_i) = \psi_i(C_i) \prod_{j=1}^q M_{ki}(S_{ki}).$$

Paso 3. Calcular $P(X_i|E_v)$ de cada nodo marginalizando la función de probabilidad del menor clique C_k que contenga el nodo X_i .

Apéndice A

Material complementario

A.1. Ejemplo de la definición alternativa a la D-separación

Ejemplo 24 (Separación en grafos dirigidos). En la Figura A.1 se incluye el grafo dirigido del que queremos estudiar las relaciones de independencia y su grafo moral. Y a continuación, en la Figura A.2 se incluyen los grafos resultantes de eliminar en cada uno de ellos los nodos que no forman parte del menor subconjunto ancestral que contenga los nodos X , Y y Z , que son los grafos en los que se aplica el criterio de U -separación.

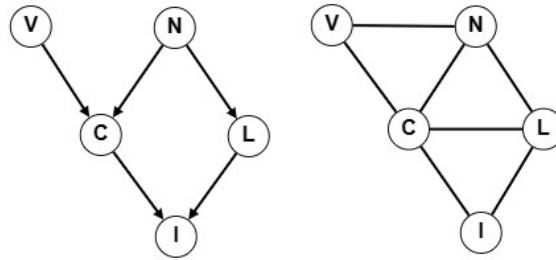


Figura A.1: Grafo dirigido(izquierda) y grafo moral (derecha).

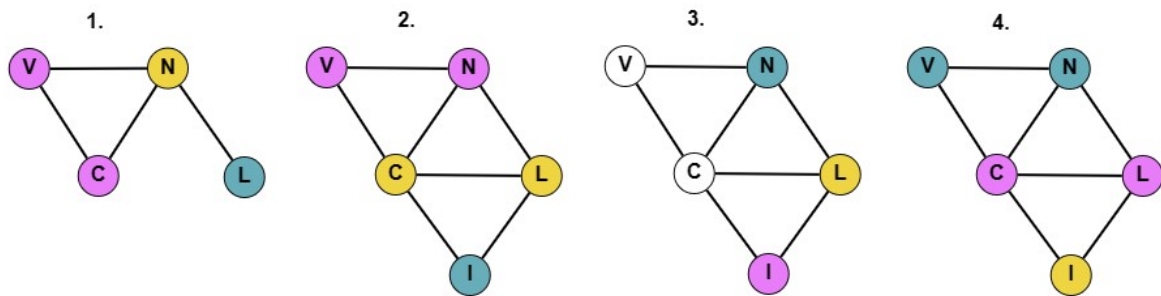


Figura A.2: Comprobación mediante los grafos moralizados resultantes de eliminar los nodos que no forman parte del conjunto ancestral. Los colores azul y rosa corresponden a los nodos pertenecientes a los conjuntos X e Y y el amarillo a los nodos del conjunto separador Z en la notación $I(X, Y|Z)$ y $D(X, Y|Z)$. Se necesita comprobar la U -separación en estos grafos para verificar las independencias condicionales.

En los grafos de la Figura A.2 se muestran las siguientes relaciones:

1. Como el menor subgrafo que contiene al conjunto ancestral está formado por los nodos V , N , C y L el grafo moral donde se aplica la U -separación en este caso es el que se muestra en 1, por lo que $I(L, \{C, V\} | N)$, ya que L y $\{C, V\}$ están U -separados por N , porque no existe un camino entre ellos que no esté cortado por C .
2. Se verifica $I(I, \{V, N\} | \{C, L\})$, ya que todos los caminos entre el nodo I y los nodos $\{V, N\}$ están cortados o por el nodo C o por el nodo L .
3. Se comprueba que $D(N, I | I)$, ya que existe un camino entre N e I , $(N - C - I)$ que no contiene al nodo L .
4. Vemos que $D(\{V, N\}, \{C, L\} | I)$, ya que los caminos entre $\{V, N\}$ y $\{C, L\}$ no pasan por el nodo I .

Por último, los nodos N y V son marginalmente independientes, ya que el menor subgrafo que los contiene es el grafo sin aristas formado por los nodos N y V , que evidentemente están U -separados.

A.2. Propagación en poliárboles. Cálculo de λ_i y ρ_i

Para calcular las funciones $\rho_i(X_i)$ se tiene que:

$$\begin{aligned} \rho_i(X_i) &= P(X_i, E_{vi}^+) = \sum_U P(X_i, U \cup E_{vi}^+) = \sum_U P(X_i | U \cup E_{vi}^+) P(U \cup E_{vi}^+) \\ &= \sum_U P(X_i | U \cup E_{vi}^+) P(U \cup E_{U_1 X_i}^+ \cup \dots \cup E_{U_p X_i}^+). \end{aligned}$$

Como se trata de un poliárbol, los conjuntos $\{U_j, E_{U_j X_i}^+\}$ y $\{U_k, E_{U_k X_i}^+\}$ son condicionalmente independientes para $k \neq j$, entonces se puede descomponer la ecuación en factores asociados al padre U_j :

$$\rho_i(X_i) = \sum_U P(X_i | U \cup E_{vi}^+) \prod_{j=1}^p P(U_j, E_{U_j X_i}^+) = \sum_U P(X_i | U \cup E_{vi}^+) \prod_{j=1}^p \rho_{U_j X_i}(U_j), \quad (\text{A.1})$$

donde $P(U_j, E_{U_j X_i}^+)$ son los mensajes que envía un nodo padre U_j a su hijo X_i y se denotan por $\rho_{U_j X_i}(U_j)$.

Análogamente para el cálculo de las funciones $\lambda_i(X_i)$ se descompone el subconjunto de evidencia E_{vi}^- , donde $E_{X_i Y_j}$ y $E_{X_i Y_k}$ son condicionalmente independientes para $k \neq j$, por lo que se tiene:

$$\lambda_i(X_i) = P(E_{vi}^- | X_i) = P(E_{X_i Y_1}^- \cup \dots \cup E_{X_i Y_c}^- | X_i) = \prod_{j=1}^c P(E_{X_i Y_j}^- | X_i) = \prod_{j=1}^c \lambda_{Y_j X_i}(X_i), \quad (\text{A.2})$$

donde $P(E_{X_i Y_j}^- | X_i)$ son los mensajes que un nodo Y_j envía a su padre X_i y se denotan por $\lambda_{Y_j X_i}(X_i)$.

Para calcular λ y ρ es necesario conocer el valor de los mensajes $\rho_{X_i Y_j}(X_i)$ y $\lambda_{Y_j X_i}(X_i)$.

- Para el cálculo de $\rho_{X_i Y_j}(X_i)$ partimos de la expresión $E_{X_i Y_j}^+ = E_{vi}^+ \cup_{k \neq j} E_{X_i Y_k}^-$ y se tiene:

$$\begin{aligned} \rho_{X_i Y_j}(X_i) &= P(X_i, E_{X_i Y_j}^+) = P(X_i, E_{vi}^+ \cup_{k \neq j} E_{X_i Y_k}^-) = P(E_{vi}^+ | X_i \cup_{k \neq j} E_{X_i Y_k}^-) P(X_i \cup_{k \neq j} E_{X_i Y_k}^-) \\ &= P(E_{vi}^+ | X_i) P(\bigcup_{k \neq j} E_{X_i Y_k}^- | X_i) P(X_i) \propto P(X_i | E_{vi}^+) \prod_{k \neq j} P(E_{X_i Y_k}^- | X_i) P(X_i) \\ &\propto \rho_i(X_i) \prod_{k \neq j} \lambda_{Y_k X_i}(X_i). \end{aligned} \quad (\text{A.3})$$

- Para el cálculo de $\lambda_{Y_j X_i}(X_i)$ se considera que X_i es el padre de Y_j y este a su vez tiene un conjunto de q padres más $S = \{S_1, \dots, S_q\}$, por lo tanto se obtiene la expresión de la evidencia dada los hijos de X_i de la forma $E_{X_i Y_j}^- = E_{Y_j}^- \cup E_{SY_j}^+$, donde $E_{Y_j}^-$ representa la evidencia a través del hijo Y_j y $E_{SY_j}^+$ representa la evidencia obtenida a través de los padres $S = \{S_1, \dots, S_q\}$ de Y_j . Luego se puede calcular:

$$\begin{aligned}
 \lambda_{Y_j X_i}(X_i) &= P(E_{X_i Y_j}^- | X_i) = \sum_{Y_j, S} P(Y_j, S, E_{X_i Y_j}^- | X_i) = \sum_{Y_j, S} P(Y_j, S, E_{Y_j}^- \cup E_{SY_j}^+ | X_i) \\
 &= \sum_{Y_j, S} P(E_{Y_j}^- | Y_j, S, E_{SY_j}^+, X_i) P(Y_j | S, E_{SY_j}^+, X_i) P(S, E_{SY_j}^+ | X_i) \\
 &= \sum_{Y_j} P(E_{Y_j}^- | Y_j) \sum_S P(Y_j | S, X_i) P(S, E_{SY_j}^+),
 \end{aligned}$$

donde la penúltima igualdad se ha obtenido mediante la regla de la cadena y la última teniendo en cuenta las relaciones de independencia condicional entre las variables. Y se tiene:

$$\lambda_{Y_j X_i}(X_i) = \sum_{Y_j} \lambda_{Y_j}(Y_j) \sum_{S_1, \dots, S_q} P(Y_j | S_i) \prod_{k=1}^q \rho_{S_k Y_j}(S_k). \quad (\text{A.4})$$

De las ecuaciones (A.1), (A.2), (A.3) y (A.4), se deduce la forma en la que se envían los mensajes $\lambda_{Y_j X_i}(X_i)$ y $\rho_{U_j X_i}(U_j)$, así como cómo se calculan las funciones $\lambda_i(X_i)$ y $\rho_i(X_i)$.

1. La función $\lambda_i(X_i)$ se puede calcular en cuanto X_i haya recibido los mensajes $\lambda_{Y_j X_i}(X_i)$ de todos sus hijos Y_j . Además, una vez calculado $\lambda_i(X_i)$, X_i puede enviar el mensaje $\lambda_{X_i U_j}(U_j)$ a su padre U_j cuando haya recibido los mensajes $\rho_{U_k X_i}(U_k)$, con $k \neq p$, del resto de sus padres.
2. La función $\rho_i(X_i)$ se puede calcular en cuanto X_i haya recibido los mensajes $\rho_{U_j X_i}(U_j)$ de todos sus padres U_j . Además, una vez calculado $\rho_i(X_i)$, X_i puede enviar el mensaje $\rho_{X_i Y_j}(X_i)$ a su hijo Y_j cuando haya recibido los mensajes $\lambda_{Y_k X_i}(X_i)$, con $k \neq p$, del resto de sus hijos.

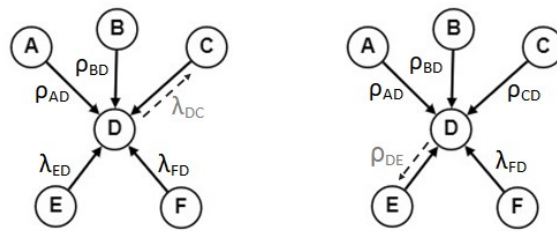


Figura A.3: El poliárbol de la izquierda representa el punto 1, y el de la derecha el punto 2. Se muestra con líneas discontinuas los mensajes que se pueden enviar a partir de los que han sido recibidos (línea continua).

Ejemplo 25. La Figura A.3 ilustra los cálculos y mensajes que se pueden enviar en ambos grafos. En el grafo de la izquierda, se tiene que el nodo D ha recibido los mensajes λ de todos sus hijos E y F por lo que se puede calcular $\lambda_D(D)$, y además, como ha recibido los mensajes de sus padres A y B , puede enviarle el mensaje λ_{DC} a su padre restante C . En el de la derecha se tiene que D ha recibido los mensajes ρ de todos sus padres, por lo que se puede calcular $\rho_D(D)$. Además, puede enviar el mensaje ρ_{DE} a su hijo E ya que ha recibido el mensaje λ del resto de sus hijos (en este caso solo tiene un hijo más, que es F).

A.3. Algoritmo de propagación mediante árboles de unión. Detalles

Partiendo de la estructura de un árbol de cliques, suponer que se tiene un clique C_i arbitrario que tiene q cliques vecinos $\{B_1, \dots, B_q\}$. Sean C_{ij} el conjunto de cliques y X_{ij} el conjunto de nodos contenidos en el subárbol correspondiente al clique C_i cuando se elimina la arista $C_i - B_j$. Análogamente los cliques y los nodos asociados al subárbol correspondiente a B_j son C_{ji} y X_{ji} , respectivamente. Por lo que $X = C_{ij} \cup C_{ji} = X_{ij} \cup X_{ji}$.

Los separadores S_{ij} correspondientes a la cadena de cliques se pueden representar mediante la intersección de cliques vecinos, es decir, $C_i \cap C_j = S_{ij}$. Para calcular la distribución de probabilidad del conjunto separador S_{ij} primero se descompone:

$$X \setminus S_{ij} = (X_{ij} \cup X_{ji}) \setminus S_{ij} = (X_{ij} \setminus S_{ij}) \cup (X_{ji} \setminus S_{ij}) = R_{ij} \cup R_{ji},$$

donde $R_{ij} = X_{ij} \setminus S_{ij}$ es el conjunto de nodos contenidos en el subárbol asociado a C_i pero no en el asociado a B_j , procedentes de eliminar $C_i - B_j$.

Por la definición de árbol de unión, todo nodo contenido en dos cliques distintos tiene que estar también contenido en todos los cliques que pasan por el camino que los une. Luego los únicos nodos comunes a X_{ij} y X_{ji} estarán contenidos en su conjunto separador S_{ij} . Por lo tanto R_{ij} y R_{ji} son disjuntos. Y se puede calcular la función de probabilidad de S_{ij} de la forma:

$$P(S_{ij}) = \sum_{X \setminus S_{ij}} \prod_{k=1}^m \psi_k(C_k) = \sum_{R_{ij} \cup R_{ji}} \prod_{k=1}^m \psi_k(C_k) = \left(\sum_{R_{ij}} \prod_{C_k \in C_{ij}} \psi_k(C_k) \right) \left(\sum_{R_{ji}} \prod_{C_k \in C_{ji}} \psi_k(C_k) \right) = M_{ij}(S_{ij}) M_{ji}(S_{ji}).$$

Siendo

$$M_{ij}(S_{ij}) = \sum_{R_{ij}} \prod_{C_k \in C_{ij}} \psi_k(C_k) \quad (\text{A.5})$$

el mensaje que el clique C_i envía al clique vecino B_j . Es decir, la función de probabilidad del conjunto separador S_{ij} , es el producto de los mensajes M_{ij} y M_{ji} .

Para calcular la función de probabilidad de un clique C_i , se descompone el conjunto $X \setminus C_i$:

$$X \setminus C_i = \left(\bigcup_{k=1}^q X_{ki} \right) \setminus C_i = \bigcup_{k=1}^q (X_{ki} \setminus C_i) = \bigcup_{k=1}^q R_{ki}.$$

La última igualdad se obtiene de que debido a la estructura de árbol de clique, cada nodo contenido en X_{ki} y C_i estará también contenido en el separador S_{ki} , por lo que: $X_{ki} \setminus C_i = X_{ki} \setminus S_{ki} = R_{ki}$. Luego la función de probabilidad del clique C_i se calcula mediante:

$$\begin{aligned} P(C_i) &= \sum_{X \setminus C_i} \prod_{j=1}^m \psi_j(C_j) = \psi_i(C_i) \sum_{X \setminus C_i} \prod_{j \neq i} \psi_j(C_j) = \psi_i(C_i) \sum_{R_{1i} \cup \dots \cup R_{qi}} \prod_{j \neq i} \psi_j(C_j) \\ &= \psi_i(C_i) \left(\sum_{R_{1i}} \prod_{C_k \in C_{1i}} \psi_k(C_k) \right) \cdots \left(\sum_{R_{qi}} \prod_{C_k \in C_{qi}} \psi_k(C_k) \right) = \psi_i(C_i) \prod_{j=1}^q M_{ki}(S_{ki}). \end{aligned} \quad (\text{A.6})$$

Para calcular los mensajes $M_{ij}(S_{ij})$, consideramos: $R_{ij} = X_{ij} \setminus S_{ij} = (C_i \setminus S_{ij}) \cup (\bigcup_{k \neq j} X_{ki} \setminus S_{ki})$, y se puede sustituir R_{ij} en la ecuación anterior (A.5).

$$\begin{aligned} M_{ij}(S_{ij}) &= \sum_{C_i \setminus S_{ij}} \sum_{X_{ki} \setminus S_{ki}: j \neq k} \prod_{C_s \in C_{ij}} \psi_s(C_s) = \sum_{C_i \setminus S_{ij}} \psi_i(C_i) \prod_{k \neq j} \sum_{X_{ki} \setminus S_{ki}} \prod_{C_s \in C_{ij}} \psi_s(C_s) \\ &= \sum_{C_i \setminus S_{ij}} \psi_i(C_i) \prod_{k \neq j} M_{ki}(S_{ki}). \end{aligned} \quad (\text{A.7})$$

Con esto ya tenemos cómo se calculan los mensajes entre cliques y por lo tanto, cómo se calculan las funciones de probabilidad $P(C_i)$, para calcular las probabilidades de cada nodo $P(X_i)$ basta con marginalizar la función en el menor clique que lo contenga. Para el cálculo de los mensajes $M_{ij}(S_{ij})$ pueden darse las siguientes situaciones:

1. Si C_i ha recibido los mensajes de todos sus vecinos, entonces C_i puede calcular y enviar los mensajes a todos sus vecinos.
2. Si C_i ha recibido los mensajes de todos sus vecinos excepto de B_j , entonces C_i solo puede calcular y enviar el mensaje $M_{ij}(S_{ij})$ al clique B_j .
3. Si C_i no ha recibido los mensajes de dos o más vecinos, entonces no se puede calcular ningún mensaje.

Apéndice B

Bnlearn

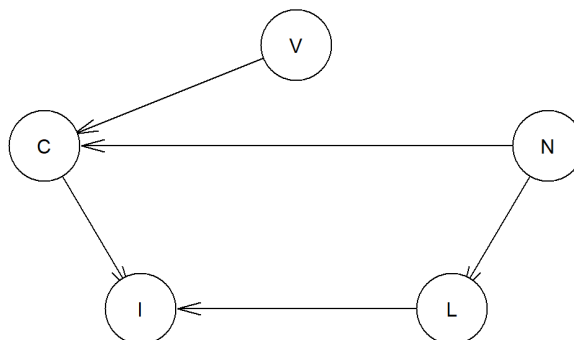
Bnlearn es una librería de [8] especializada en el manejo de redes Bayesianas. Proporciona gran variedad de funciones para el aprendizaje de la estructura gráfica de estas redes, su manipulación y análisis, así como para la representación visual de la red y sus propiedades. Además, permite la estimación de parámetros y distribuciones de probabilidad asociadas a las redes Bayesianas. En este capítulo, se utiliza esta librería para la comprobación de los conceptos y resultados mostrados a lo largo de este trabajo. Como no ofrece los algoritmos de inferencia exacta utilizados en el Capítulo 3, se han estimado las distribuciones de probabilidad de los ejemplos mediante algoritmos de inferencia aproximada.

B.1. Construcción y análisis de una red Bayesiana

En este apartado se muestran algunas funciones básicas para la creación, representación y análisis de una red Bayesiana, así como la manipulación de esta y su transformación en otras estructuras útiles como los *grafos esqueletos* o los *grafos morales*. Para esto se utiliza el grafo del Ejemplo 12 con el que se ha trabajado durante el trabajo.

Creación y representación del grafo dirigido

```
# Creación del grafo dirigido
Gdir <- model2network(" [V] [N] [C|V:N] [L|N] [I|C:L] ")
plot(Gdir)
```



Nodos y arcos del grafo

```
nodes(Gdir)
## [1] "C" "I" "L" "N" "V"
```

```
arcs(Gdir)
##      from to
## [1,] "V"  "C"
## [2,] "N"  "C"
## [3,] "N"  "L"
## [4,] "C"  "I"
## [5,] "L"  "I"
```

Comprobación que el grafo es acíclico y dirigido

```
acyclic(Gdir, directed = FALSE, debug = FALSE)
## [1] TRUE
directed(Gdir)
## [1] TRUE
```

Padres, hijos y vecinos de un nodo

```
parents(Gdir, "C")
## [1] "N" "V"
children(Gdir,node="C")
## [1] "I"
nbr(Gdir,node="C")
## [1] "I" "N" "V"
```

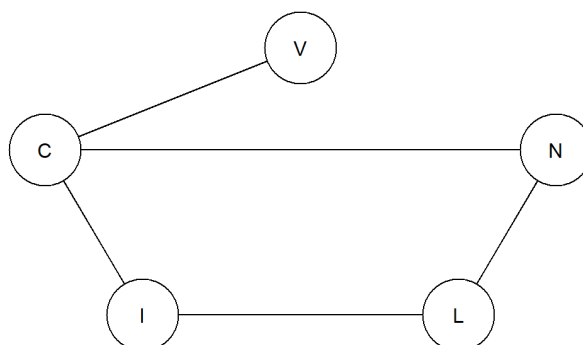
Caminos entre nodos

¿Existe un camino dirigido de *I* a *C*? ¿Y no dirigido?

```
path.exists(Gdir, from = "I", to = "C", direct = TRUE, underlying.graph =
FALSE, debug = FALSE)
## [1] FALSE
path.exists(Gdir, from = "I", to = "C", direct = TRUE, underlying.graph =
TRUE, debug = FALSE)
## [1] TRUE
```

Construcción del esqueleto

```
skeleton(Gdir)
plot(skeleton(Gdir))
```

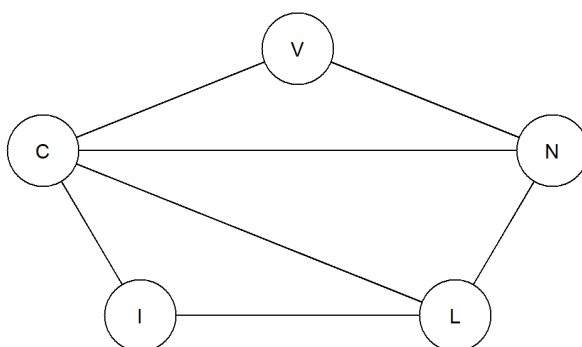


V-estructuras del grafo

```
vstructs(Gdir)
##      X   Z   Y
## [1,] "N" "C" "V"
## [2,] "C" "I" "L"
```

Grafo moral asociado

```
moral<-moral(Gdir)
plot(moral)
```



B.2. Comprobación de las independencias condicionales

Comprobamos el concepto de D-separación aplicado en el Ejemplo 14.

```
# 1.  $I(\{V, C\}, L | N)$ 
if (!dsep(Gdir, "C", "L", "N") &
    !dsep(Gdir, "V", "L", "N")) {
  print(FALSE)
} else {
  print(TRUE)}
## [1] TRUE

# 2.  $I(\{V, N\}, I | \{C, L\})$ 
dsep(Gdir, "V", "I", c("C", "L"))
## [1] TRUE

# 3.  $D(I, N | L)$ 
dsep(Gdir, "I", "N", "L")
## [1] FALSE

# 4.  $D(\{C, L\}, \{V, N\} | I)$ 
if (!dsep(Gdir, "C", "V", "I") &
    !dsep(Gdir, "C", "N", "I") &
    !dsep(Gdir, "L", "V", "I") &
    !dsep(Gdir, "L", "N", "I")) {
  print(FALSE)
}
```

```

} else {
  print(TRUE)}
## [1] FALSE

```

B.3. Ejemplo de propagación en poliárboles

Para la comprobación del Ejemplo 20 de propagación en poliárboles utilizaremos algoritmos de inferencia aproximada.

Descripción de las relaciones entre los nodos de la red Bayesiana y asignación las probabilidades asociadas a cada uno de ellos.

```

#Descripción de la red Bayesiana
Gpoliarbol= model2network(" [A] [B|A] [E|B] [D|B] [C|D] [F|C] [G|D] ")

#Asignación de probabilidades a cada nodo
pA=matrix(c(0.6,0.4),ncol=2,dimnames=list(NULL,c("P(A=0)", "P(A=1)")))
pB.A=matrix(c(0.7,0.3,0.5,0.5),ncol=2,nrow=2,dimnames=list("B"=c("P(B=0)",
"P(B=1)"), "A"=c("P(A=0)", "P(A=1)")))
pE.B=matrix(c(0.2,0.8,0.65,0.35),ncol=2,nrow=2,dimnames=list("E"=c("P(E=0)",
"P(E=1)"), "B"=c("P(B=0)", "P(B=1)")))
pD.B=matrix(c(0.45,0.55,0.9,0.1),ncol=2,nrow=2,dimnames=list("D"=c("P(D=0)",
"P(D=1)"), "B"=c("P(B=0)", "P(B=1)")))
pC.D=matrix(c(0.8,0.2,0.5,0.5),ncol=2,nrow=2,dimnames=list("C"=c("P(C=0)",
"P(C=1)"), "D"=c("P(D=0)", "P(D=1)")))
pF.C=matrix(c(0.3,0.7,0.1,0.9),ncol=2,nrow=2,dimnames=list("F"=c("P(F=0)",
"P(F=1)"), "C"=c("P(C=0)", "P(C=1)")))
pG.D=matrix(c(0.85,0.15,0.4,0.6),ncol=2,nrow=2,dimnames=list("G"=c("P(G=0)",
"P(G=1)"), "D"=c("P(D=0)", "P(D=1)")))

#Tabla de probabilidades
tprob1=custom.fit(Gpoliarbol, dist=list("A"=pA, "B"=pB.A, "C"=pC.D, "D"=pD.B,
"E"=pE.B, "F"=pF.C, "G"=pG.D))

```

Comprobación mediante inferencia aproximada de los resultados de la Figura 3.2.

```

t.countsA = table(cpdist(tprob1, nodes="A", evidence=((B == "P(B=1)") &
(C == "P(C=0)")), method = "ls"))
prop.table(t.countsA)
##
##      P(A=0)      P(A=1)
## 0.4990177 0.5009823
t.countsB = table(cpdist(tprob1, nodes="B", evidence=((B == "P(B=1)") &
(C == "P(C=0)")), method = "ls"))
prop.table(t.countsB)
##
## P(B=0) P(B=1)
##      0      1

```

```

t.countsC = table(cpdist(tprob1, nodes="C", evidence=((B == "P(B=1)") &
(C == "P(C=0)")), method = "ls"))
prop.table(t.countsC)
##
## P(C=0) P(C=1)
##      1      0
t.countsD = table(cpdist(tprob1, nodes="D", evidence=((B == "P(B=1)") &
(C == "P(C=0)")), method = "ls"))
prop.table(t.countsD)
##
##      P(D=0)      P(D=1)
## 0.94348725 0.05651275
t.countsE = table(cpdist(tprob1, nodes="E", evidence=((B == "P(B=1)") &
(C == "P(C=0)")), method = "ls"))
prop.table(t.countsE)
##
## P(E=0) P(E=1)
## 0.66577 0.33423
t.countsF = table(cpdist(tprob1, nodes="F", evidence=((B == "P(B=1)") &
(C == "P(C=0)")), method = "ls"))
prop.table(t.countsF)
##
##      P(F=0)      P(F=1)
## 0.2927702 0.7072298
t.countsG = table(cpdist(tprob1, nodes="G", evidence=((B == "P(B=1)") &
(C == "P(C=0)")), method = "ls"))
prop.table(t.countsG)
##
##      P(G=0)      P(G=1)
## 0.8174825 0.1825175

```

B.4. Ejemplo de métodos de agrupamiento

Por último, comprobamos los resultados procedentes de la inferencia en la red Bayesiana que modela el riesgo de incendios, para la que se ha tomado la evidencia de estar en el norte, $N = 1$, en la estación de primavera/verano, $V = 1$. A continuación, se asignan las probabilidades que definen el modelo.

```

#probabilidades marginales
pV=matrix(c(0.5,0.5),ncol=2,dimnames=list(NULL,c("P(V=0)","P(V=1)")))
pN=matrix(c(0.6,0.4),ncol=2,dimnames=list(NULL,c("P(N=0)","P(N=1)")))

#probabilidades condicionadas
#prob C dado V, N
pC.VN=c(0.3,0.7,0.05,0.95,0.9,0.1,0.75,0.25)
dim(pC.VN)=c(2,2,2)
dimnames(pC.VN)=list("C"=c("P(C=0)","P(C=1)"),"V"=c("P(V=0)","P(V=1)"),
"N"=c("P(N=0)","P(N=1)"))

```

```
#prob I dado C,L
pI.CL=c(0.8,0.2,0.9,0.1,0.7,0.3,0.9,0.1)
dim(pI.CL)=c(2,2,2)
dimnames(pI.CL)=list("I"=c("P(I=0)", "P(I=1)"), "C"=c("P(C=0)", "P(C=1)"),
"L"=c("P(L=0)", "P(L=1)"))

#prob L dado N
pL.N=matrix(c(0.9,0.1,0.3,0.7),ncol=2,nrow=2,dimnames=list("L"=c("P(L=0)",
"P(L=1)"), "N"=c("P(N=0)", "P(N=1)")))

#tabla de probabilidades
tprob=custom.fit(Gdir, dist=list("V"=pV, "N"=pN, "C"=pC.VN, "L"=pL.N, "I"=pI.CL))
```

Cálculo de la probabilidades en los cliques (estos resultados se muestran en la Figura 3.5):

(a) $P(N, V, C)$

```
set.seed(123)
t.counts = table(cpdist(tprob, nodes=c("N", "V", "C"), evidence=((V == "P(V=1)"))
&(N == "P(N=1)")), method = "ls"))
prop.table(t.counts)
## , , C = P(C=0)
##
##      V
## N      P(V=0)    P(V=1)
## P(N=0) 0.0000000 0.0000000
## P(N=1) 0.0000000 0.7512291
##
## , , C = P(C=1)
##
##      V
## N      P(V=0)    P(V=1)
## P(N=0) 0.0000000 0.0000000
## P(N=1) 0.0000000 0.2487709
```

(b) $P(N, C, L)$:

```
set.seed(123)
t.counts = table(cpdist(tprob, nodes=c("N", "C", "L"), evidence=((V == "P(V=1)"))
&(N == "P(N=1)")), method = "ls"))
prop.table(t.counts)
## , , L = P(L=0)
##
##      C
## N      P(C=0)    P(C=1)
## P(N=0) 0.0000000 0.0000000
## P(N=1) 0.22025565 0.07669617
##
## , , L = P(L=1)
##
```

```
##           C
## N           P(C=0)      P(C=1)
## P(N=0) 0.00000000 0.00000000
## P(N=1) 0.53097345 0.17207473
```

(c) $P(C, L, I)$

```
set.seed(123)
t.counts = table(cpdist(tprob, nodes=c("C", "L", "I"), evidence=((V == "P(V=1)")
&(N == "P(N=1)")), method = "ls"))
prop.table(t.counts)
## , , I = P(I=0)
##
##           L
## C           P(L=0)      P(L=1)
## P(C=0) 0.174041298 0.363815143
## P(C=1) 0.067846608 0.151425762
##
## , , I = P(I=1)
##
##           L
## C           P(L=0)      P(L=1)
## P(C=0) 0.048180924 0.168141593
## P(C=1) 0.005899705 0.020648968
```

Cálculo de las probabilidades marginales:

(a) La probabilidad de que haga calor, $P(C)$ es:

```
t.counts = table(cpdist(tprob, nodes="C", evidence=((V ==
"P(V=1)")&(N == "P(N=1)")), method = "ls"))
prop.table(t.counts)
##
## P(C=0) P(C=1)
## 0.7634298 0.2365702
```

(b) La probabilidad de tener un día lluvioso, $P(L)$:

```
t.counts = table(cpdist(tprob, nodes="L", evidence=((V ==
"P(V=1)")&(N == "P(N=1)")), method = "ls"))
prop.table(t.counts)
##
## P(L=0) P(L=1)
## 0.3021654 0.6978346
```

(c) La probabilidad de incendio, $P(I = 1)$ es:

```
cpquery(tprob, event = (I == "P(I=1)"), evidence = ((V ==
"P(V=1)")&(N == "P(N=1)")), method = "ls")
## [1] 0.2412371
```


Bibliografía

- [1] E. CASTILLO, J.M. GUTIÉRREZ, Y A.S. HADI, *Sistemas expertos y modelos de redes probabilísticas*, Monografías de la Academia de Ingeniería, (1996), URL: <https://personales.unican.es/gutierjm/papers/bookcgh.pdf>.
- [2] P.R. KUMAR, J.M. WAINWRINGHT, R. ZECCHINA, *Graphical models and message-passing algorithms: Some introductory lectures*, Mathematical Foundations of Complex Networked Information Systems: Politecnico di Torino, Verrès, Italy (2009).
- [3] I. LAYO, *Análisis e implementación del operador de Bayes multivariante en redes Bayesianas, marginalización aproximada vía simulación*, Trabajo Fin de Grado, Universidad Politécnica de Madrid (2022).
- [4] L. E. SUCAR, *Probabilistic Graphical Models: Principles and Applications*, 2nd edition, Springer (2021).
- [5] L. E. SUCAR, *Métodos de inteligencia artificial*, Universidad Popular Autónoma del Estado de Puebla, (2014).
- [6] S.L. LAURITZEN, *Graphical Models*, Oxford University Press, (1996).
- [7] E. CHARNIAK, *Bayesian networks without tears*, AI magazine, vol. 12, no 4, p. 50-50 (1991).
- [8] R CORE TEAM , *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, (2021). URL: <https://www.R-project.org/>.
- [9] M. SCUTARI, *Learning Bayesian Networks with the bnlearn R Package.*, Journal of Statistical Software, 35(3), 1–22, (2010).URL: <http://www.jstatsoft.org/v35/i03/>.