Inteligencia Artificial

Tema 5: Razonamiento con Incertidumbre

1. Incertidumbre y conocimiento (Variables aleatorias)

A la hora de representar conocimiento existen reglas (reglas lógicas de descripción, reglas de primer orden, redes bayesianas, etc). Cada formalismo de representación emplea un método de **inferencia** específico (razonamiento hacia adelante, razonamiento hacia atrás, tableros, etc).

Este tema se centra en la teoría y cálculo de probabilidades para representar el conocimiento y razonar en base al mismo.

Ejemplo de conocimiento expresado mediante reglas:

- Si SINTOMA = DOLOR DE MUELAS, entonces ENFERMEDAD = CARIES

Podemos añadir más condicionales cómo:

- Si SINTOMA = DOLOR DE MUELAS, entonces, ENFERMEDAD = CARIES Ó SINUSITIS Ó MUELA DEL JUICIO, ETC….

Pero este conocimiento no es exacto ni preciso dado que no tenemos información completa, n es determinista. Podemos expresar el conocimiento mediante un grado de creencia:

- Creemos, en base a percepciones, que un paciente con DOLOR DE MUELAS, tendrá CARIES con un 80% de probabilidades.

Esto NO expresa el grado de verdad sino de creencia. La probabilidad puede cambiar según surgen nuevas evidencias. La probabilidad sirve por tanto para representar el conocimiento incierto.

Las variables aleatorias serán una parte de este conocimiento cuyo estado podemos desconocer:

- Variable aleatoria CARIES que expresa que el paciente puede tener o no tener caries.

Nuestra descripción del entorno vendrá dada por muchas variables aleatorias que pueden tomar valores dentro de un dominio; en este caso, CARIES podrá ser True o False. Pueden ser:

- Booleanas (cómo CARIES, que expresaremos cómo True/False o cómo CARIES/¬CARIES.
- Discretas (que incluye a las booleanas) ← Nos centramos en estas.
- Continuas.

Usando proposicionales junto con variables discretas podemos formas proposiciones (una proposición es una expresión que puede ser verdadera o falsa):

- CARIES ∧ ¬DOLOR

A cada proposición asignaremos probabilidades de forma que expresen el grado de creencia en las mismas.

2. Probabilidad incondicional

Dada una proposición *a*, su probabilidad incondicional *P(a)*, cuantifica el grado de creencia en que *a* ocurra en ausencia de cualquier otra información. Llamamos aproximación frecuentista al nº de casos favorables (veces que se cumple *a*) entre el nº de casos totales.

Una función de probabilidad es una función definida en el conjunto de proposiciones que verifica:

- 0 <= P(a) <= 1, para toda proposición a.
- P(true) = 1 y P(false) = 0.
- P(a ∨ b) = P(a) + P(b) - P(a ∧ b) para cualquier a y b.
- P(¬a) = 1 – P(a)

Partiendo de lo anterior, la **distribución de probabilidad** de una variable aleatoria indica las probabilidades que la variable puede tomar en casa uno de sus valores:

- P(Tiempo=sol) = 0.7; P(Tiempo=lluvia) = 0.2, P(Tiempo=nublado) = 0.08, P(Tiempo=nieve) = 0.02

Denotamos las distribuciones de probabilidad de forma compacta cómo:

- **P**(Tiempo) = {0.7;0.2;0.08;0.02}

Así mismo, llamamos **distribución de probabilidad conjunta** a la probabilidad de cada combinación de valores de dos o más variables aleatorias:

- **P**(X,Y); por ejemplo: **P**(Tiempo,Caries) sería una tabla de 4x2

Dado un conjunto de variables aleatorias que describen nuestro entorno, llamamos **evento atómico** a un tipo particular de proposición: conjunción de proposicionales elementales que expresan un valor CONCRETO para TODAS y cada una de las variables.

Ejemplo: si todas las variables aleatorias que describen nuestro entorno son CARIES y DOLOR, los posibles eventos atómicos son:

- CARIES ∧ DOLOR
- CARIES ∧ ¬DOLOR
- ¬CARIES ∧ DOLOR
- ¬CARIES ∧ ¬DOLOR

Los eventos atómicos son excluyentes, exhaustivos (alguno TIENE que ocurrir), implica la verdad o falsedad de toda la proposición y puede expresarse cómo la disyunción de un conjunto de eventos atómicos: CARIES = (CARIES ∧ DOLOR) ∨ (CARIES ∧ ¬DOLOR).

A la tabla que antes mencionamos, la que representa una distribución de probabilidad conjunta y completa, la llamamos DCC (especificación completa); por ejemplo:

	dolor	*dolor*	¬*dolor*	¬*dolor*
	hueco	¬*hueco*	*hueco*	¬*hueco*
caries	0.108	0.012	0.072	0.008
¬*caries*	0.016	0.064	0.144	0.576

De esta DCC podemos extraer que:

P(Caries) = P(Caries,Dolor,Hueco) + P(Caries, Dolor, ¬Hueco) + P(Caries, ¬Dolor,Hueco) + P(Caries, ¬Dolor, ¬Hueco) = 0.108 + 0.012 + 0.072 + 0.008 = 0.2

P(Caries ∨ Dolor) = P(Caries,Dolor,Hueco) + P(Caries, Dolor, ¬Hueco) + P(Caries, ¬Dolor,Hueco) + P(Caries, ¬Dolor, ¬Hueco) + P(¬Caries,Dolor,Hueco) + P(¬Caries,Dolor, ¬Hueco) = 0.108 + 0.012 + 0.072 + 0.008 + 0.016 + 0.064 = 0.28

etc…

Generalizando este cálculo: $\mathbf{P(Y) = \sum_z P(Y, z)}$ (regla de *marginalización*)

Que denotamos cómo:

- **Y** es un vector de variables aleatorias que simboliza cualquier combinación de valores de estas.
- **z** representa una combinación de valores concretos para un conjunto **Z** de variables aleatorias (las restantes).
- Hay un sumando **P(Y,z)** por cada posible , y cada sumando es una celda de la tabla DCC.

A su vez, la DCC presenta problemas:

- Tamaño exponencial.
- Rara vez se conocen las probabilidades de TODOS los eventos atómicos.

En su lugar, el conocimiento del dominio se expresa mejor con probabilidades condicionales.

3. <u>Probabilidad condicional: inferencia probabilística</u>

Denotamos la probabilidad condicional asociada a *a* dado *b* (donde *a* y *b* son proposiciones) cómo P(a|b) y representa el grado de creencia sobre *a*, dad que todo lo que sabemos es que *b* ocurre.

- P(Caries|Dolor) = 0.8 implica que una vez sabemos que un paciente tiene dolor de muelas (y sólo sabemos eso), nuestra creencia es que el paciente tendrá caries con una probabilidad del 0.8 ~ 80%.

Existe una relación entre probabilidad condicional e incondicional:

- P(a|b) = P(a ∧ b) / P(b) aunque también puede denotarse cómo P(a ∧ b) = P(a|b) * P(b)

Las probabilidades condicionales son el reflejo de que las creencias varían según se descubren nuevas evidencias, pero NO es lo mismo que una implicación lógica, es decir, P(a|b) = 0.8 no es lo mismo que decir que SIEMPRE que *b* sea verdad, entonces P(a) será 0.8. Esto es falso, ya que lo único conocido es *b*.

Entendemos por **inferencia probabilística** al cálculo de la probabilidad de una proposición dada condicionada por la observación de varias evidencias. El conocimiento base vendrá dado por una DCC. Los algoritmos de inferencia probabilística son lo más importante de este tema.

Inferencia probabilística a partir de una DCC:

- P(Caries|Dolor) = P(Caries ∧ Dolor) / P(Dolor) = (0.108 + 0.012) / (0.108 + 0.012 + 0.016 + 0.064) = 0.6 (mirando la tabla de la página anterior).

Normalización: consiste en evitar el cálculo de P(Dolor); en su lugar, P(Caries|Dolor) = α P(Caries,Dolor) = α[P(Caries,Dolor,Hueco) + P(Caries,Dolor,Hueco)] = α[{0.108;0.016} + {0.012;0.064}] = α{0.12;0.08} = {0.6;0.4}; lo que hacemos es calcular P(Caries,Dolor) y P(¬ Caries,Dolor) y luego multiplicar por una constante α que haga que ambos sumen 1.

Generalizando el cálculo de una inferencia probabilística a partir de una DCC, tenemos:

$$\mathbf{P}(X|\mathbf{e}) = \alpha\mathbf{P}(X,\mathbf{e}) = \alpha\sum_{\mathbf{y}}\mathbf{P}(X,\mathbf{e},\mathbf{y})$$

- Existe un sumando por cada combinación **y** de valores de variables **Y** no observadas.
- Para cada valor de **X**, cada sumando **P(X,e,y)** es una celda de la DCC.
- **α** es una constante de normalización que hace que la suma de la distribución sume 1.
- Dada cualquier DCC, la fórmula anterior nos da un método para realizar una inferencia probabilística.

Presenta sus problemas:

- En la práctica resulta exponencial; con *n* variables, se necesita un tiempo 0(2^n).
- Un problema real puede tener del orden de cientos o miles de variables.

Ahora introducimos la idea de independencia: veremos que la posible independencia existente entre los eventos que describen las distintas variables aleatorias reduce esta complejidad.

4. Independencia

En casos prácticos, vemos que algunas de las variables son independientes entre sí:

- P(Tiempo=nublado|Dolor,Caries,Hueco) = P(Tiempo=nublado)
- Si la variable Tiempo (que puede tomar 4 valores) formara parte de una descripción que incluye a Dolor,Caries,Hueco, no necesitaríamos una tabla de 32 entradas sino dos tablas independientes de 8+4 entradas.

De forma intuitiva, dos variables son independientes si conocer el valor de una no nos actualiza el grado de creencia sobre el valor que tomará la otra. Con carácter general, asumir que dos variables son independientes está basado en el conocimiento previo sobre el entorno que se está modelando.

De forma formal, dos variables aleatorias **X** e **Y** son independientes si P(X|Y) = P(X) y de forma equivalente P(Y|X) = P(Y) ó P(X,Y) = P(X) * P(Y).

Asumir independencia sobre variables reduce la exponencialidad del problema.

5. Independencia condicional

En el ejemplo anterior, Dolor y Hueco no son independientes, dado que ambas dependen de Caries. Sólo una vez conocido el valor de Caries, pasan a ser independientes.

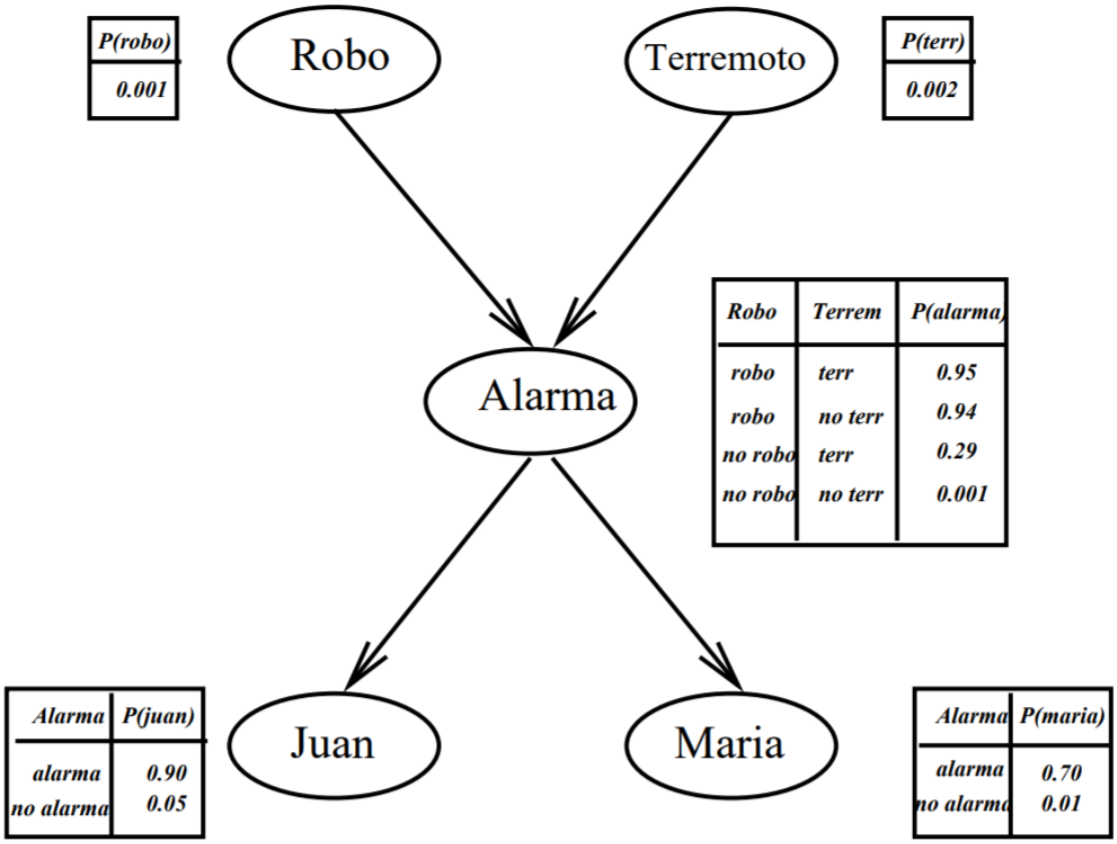
De forma intuitiva, **X** es condicionalmente independiente de **Y** dado un conjunto de variables **Z** si nuestro grado de creencia en que **X** tome un valor dado, sabiendo el valor que toman las variables de **Z**, no se vería actualizado si además supiéramos el valor que toma **Y**.

De forma formal, dos variables aleatorias **X** e **Y** son independientes dado un conjunto de variables aleatorias **Z** si P(X,Y|Z) = P(X|Z) * P(Y|Z) y de forma equivalente P(X|Y,Z) = P(X|Z) ó P(Y|X,Z) = P(Y|Z).

6. Redes bayesianas

Las redes bayesianas (también llamadas redes de creencia) constituyen una manera práctica y compacta de representar conocimiento incierto.

- Tenemos una alarma antirrobo instalada en una casa
- La alarma salta normalmente con la presencia de ladrones
- Pero también cuando ocurren pequeños temblores de tierra
- Tenemos dos vecinos en la casa, Juan y María, que han prometido llamar a la policía si oyen la alarma
  - Juan y María podrían no llamar aunque la alarma sonara: por tener música muy alta en su casa, por ejemplo
  - Incluso podrían llamar aunque no hubiera sonado: por confundirla con un teléfono, por ejemplo



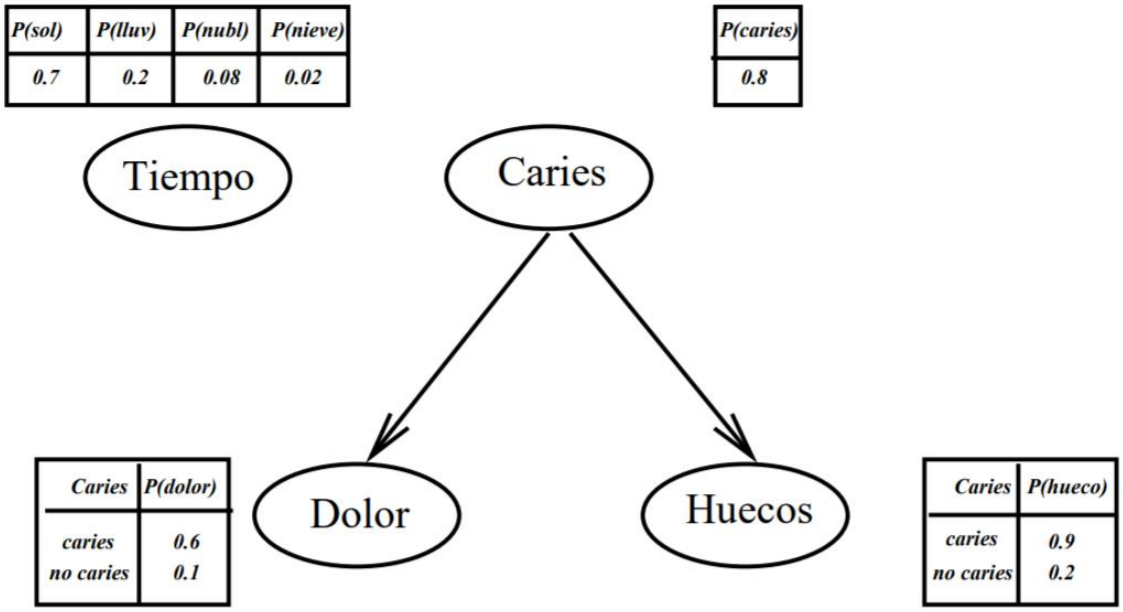
Una red bayesiana es un grafo dirigido acíclico que consta de:

- Un conjunto de nodos, uno por cada variable aleatoria del entorno a representar.
- Un conjunto de aristas que conectan los nodos:
  - Si hay una arista de **X** a **Y** decimos que **X** es un padre de **Y**.
  - A partir del concepto de padre, definimos también el concepto de antecesores y descendientes de una variable.
- Cada nodo **Xi** contiene la distribución de probabilidad condicional P(Xi|padres(Xi)).
  - Si Xi es booleana, se suele omitir la probabilidad de la negación.

De forma intuitiva, si se conoce el valor que toman las variables padres de una variable aleatoria, entonces el grado de creencia en que la variable tome un valor determinado, no se ve actualizado si además conocemos el valor que toma otra variable no descendiente.

Es tarea del experto o de un modelo de aprendizaje automático, el decidir las relaciones de independencia condicional (ósea, la topología de la red bayesiana).

Si expresamos el ejemplo de las Caries, Dolor, Huecos cómo red bayesiana queda:



Este modelo nos está expresando qué:

- Caries es una causa directa de Dolor y Huecos
- Dolor y Huecos son condicionalmente independientes dada Caries
- Tiempo es independiente de las restantes variables
- No es necesario dar la probabilidad de las negaciones de caries, dolor, etc.

De forma formal, las redes bayesianas representan DCCs; si consideramos una red bayesiana con *n* variables aleatorias y un orden entre estas: X1, … , Xn; suponemos que:

- padres(Xi) ⊆ {Xi-1, … ,X1} → El orden escogido debe ser consistente con el orden parcial que induce el grafo.
- P(Xi|Xi-1, … ,X1) = P(Xi|padres(Xi)) → cada variable es condicionalmente independiente de todos los que son sus descendientes, dados sus padres.

Hemos dicho que las redes bayesianas son compactas; para ello introducimos el concepto de **dominios localmente estructurados**: las relaciones de independencia que existen entre las variables de un dominio hacen que las redes bayesianas sean una representación mucho más compacta frente a una tabla con todas las posibles combinaciones de valores.

Con *n* variables, si cada variable está directamente influenciada por *k* variables (a lo sumo), entonces una red bayesiana necesita n*2^k números frente a los 2^n números que se necesitarían en una DCC.

Puntualmente, una variable influye directamente en otra, pero esta dependencia es muy tenue, por lo que suele compensar no considerar esa dependencia, perdiendo precisión a cambio de ganar manejabilidad.

Lógicamente, las redes bayesianas también presentan problemas:

- Crear la red de manera que modele bien la realidad representada a la par que sea compacta.
- Deducir las independencias condicionales entre grupos de variables.
- Inferencia probabilística:
    - Inferencia exacta → algoritmos de enumeración y eliminación de variables.
    - Inferencia aproximada → algoritmos de muestreo con rechazo y ponderación por verosimilitud.

Algoritmo de construcción de redes bayesianas

FUNCION CONSTRUYE_RED(VARIABLES)
1. Sea $(X_1, \ldots, X_n)$ una ordenación de VARIABLES
2. Sea RED una red bayesiana ''vacía''
3. PARA $i = 1, \ldots, n$ HACER
3.1 Añadir un nodo etiquetado con X_i a RED
3.2 Sea padres(X_i) un subconjunto minimal de $\{X_{i-1}, \ldots, X_1\}$ tal que existe una independencia condicional entre X_i y cada elemento de $\{X_{i-1}, \ldots, X_1\}$ dado padres(X_i)
3.3 Añadir en RED un arco dirigido entre cada elemento de padres(X_i) y X_i
3.4 Asignar al nodo X_i la tabla de probabilidad P(X_i|padres(X_i))
4. Devolver RED

7. Construcción de redes bayesianas: Orden, Inferencia y Algoritmos

El problema del orden: elegir el orden entre variables puede ser un problema, dado que un mal orden puede llevar a representaciones poco eficientes. En general comenzamos por las "causas originales" siguiendo aquellas a las que influencian directamente hasta llegar a las que no influyen directamente en ninguna.

El problema de la inferencia: calcular la probabilidad a posteriori para un conjunto de variables es complejo. **X** denotará la variable de consulta, **E** el conjunto de variables de evidencia y **e** una observación concreta para esas variables e **Y** el conjunto de las restantes variables de la red así cómo **y** representa un conjunto de cualquier valor para esas variables.

Inferencia por enumeración

Recordamos la fórmula de la inferencia probabilística a partir de una DCC de la página 4:

$$\mathbf{P}(X|\mathbf{e}) = \alpha\mathbf{P}(X,\mathbf{e}) = \alpha\sum_{\mathbf{y}}\mathbf{P}(X,\mathbf{e},\mathbf{y})$$

- Entrada: una v.a. X de consulta, un conjunto de valores observados **e** para la variables de evidencia y una red bayesiana
- Salida: $\mathbf{P}(X|\mathbf{e})$

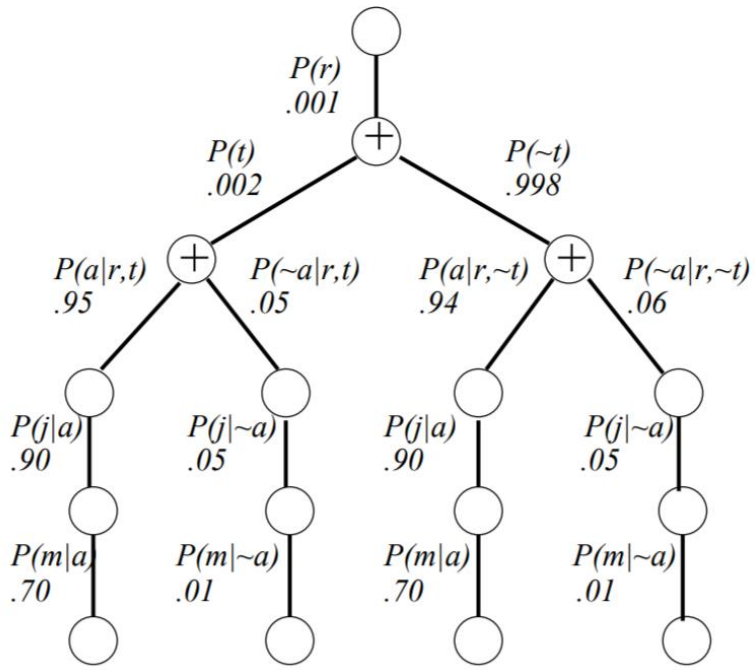
Algoritmo de inferencia por enumeración

```
FUNCION INFERENCIA_ENUMERACION(X,e,RED)
1. Sea Q(X) una distribución de probabilidad sobre X,
   inicialmente vacía
2. PARA cada valor x_i de X HACER
   2.1 Extender e con el valor x_i para X
   2.2 Hacer Q(x_i) el resultado de
       ENUM_AUX(VARIABLES(RED),e,RED)
3. Devolver NORMALIZA(Q(X))
```

Algoritmo de inferencia por enumeración

```
FUNCION ENUM_AUX(VARS,e,RED)
1. Si VARS es vacío devolver 1
2. Si no,
   2.1 Hacer Y igual a PRIMERO(VARS)
   2.2 Si Y tiene un valor y en e,
       2.2.1 devolver P(y|padres(Y,e))·ENUM_AUX(RESTO(VARS),e)
       2.2.2 Si no, devolver
       SUMATORIO(y,P(y|padres(Y,e))·ENUM_AUX(RESTO(VARS),e_y))
       (donde: padres(Y,e) es el conjunto de valores que toman en e
        los padres de Y en la RED, y e_y extiende e con el valor y
        para Y)
```

, cuya salida puede expresarse cómo un árbol:



VARIABLES(RED) debe devolver las variables en un orden consistente. El recorrido en profundidad garantiza que el árbol se genere de abajo a arriba con un coste lineal. Plante un problema: los cálculos repetidos.

El problema de los cálculos redundantes: para evitar la repetición de cálculos se puede:

- Realizar las operaciones correspondientes a cada sumatorio UNA SOLA VEZ, para TODOS los posibles valores de las variables que intervienen en dicho sumatorio.
- En lugar de multiplicar números, multiplicar tablas de probabilidades que denominamos factores.
- Esta idea da lugar al algoritmo de eliminación de variables.

Algoritmo de eliminación de variables

Tomar, al comienzo, todos los factores (tablas de probabilidad) que intervienen en el cálculo. A partir de estas:

- Cada variable aporta su propio factor.
- En dichas tablas, los valores de las variables de evidencia ya están fijados.

El factor correspondiente a M se obtiene a partir de la distribución condicional P(M|A):

- Cómo M es una variable de evidencia y su valor está fijado a true, el factor correspondiente, denotado fM(A), es P(m|A):
A	$\mathbf{f_M}(A) = \mathbf{P}(m\|A)$
a	0,70
$\neg a$	0,01

El factor correspondiente a J se obtiene a partir de la distribución condicional P(J|A):

- Cómo J es una variable de evidencia y su valor está fijado a true, el factor correspondiente fJ(A) es P(j|A):
A	$\mathbf{f_J}(A) = \mathbf{P}(j\|A)$
a	0,90
$\neg a$	0,05

Una vez tenemos todos los factores iniciales vamos “eliminando” las variables que no son ni de consulta ni de evidencia:

- De forma intuitiva, el proceso de “eliminación” de una variable se corresponde a al sumatorio de la correspondiente fórmula, pero en forma de tablas.
- Cuando eliminemos todas las variables, solo quedan factores dependientes de la variable de consulta, que habrán de ser multiplicados y normalizados.

El orden de “eliminación” no altera al resultado final, pero si influye en la eficiencia del algoritmo.

Por ejemplo, eliminamos en primer lugar la variable A

- Se correspondería con realizar $\sum_a \mathbf{P}(A|R,T)P(j|A)P(m|A)$, o lo que es lo mismo, hacer $\sum_a \mathbf{f_A}(A,R,T)\mathbf{f_J}(A)\mathbf{f_M}(A)$
- La multiplicación de $\mathbf{f_M}$, $\mathbf{f_J}$ y $\mathbf{f_A}$, notada $\mathbf{f}_{\times\mathbf{A}}(A,R,T)$ se obtiene multiplicando las entradas correspondientes a los mismos valores de A, R y T
- Es decir, para cada valor v_1 de A, v_2 de R y v_3 de T se tiene $\mathbf{f}_{\times\mathbf{A}}(v_1,v_2,v_3) = \mathbf{f_M}(v_1)\mathbf{f_J}(v_1)\mathbf{f_A}(v_1,v_2,v_3)$. Por ejemplo: $\mathbf{f}_{\times\mathbf{A}}(\mathit{true},\mathit{false},\mathit{true}) = \mathbf{f_M}(\mathit{true})\mathbf{f_J}(\mathit{true})\mathbf{f_A}(\mathit{true},\mathit{false},\mathit{true}) = 0{,}70 \times 0{,}90 \times 0{,}29 = 0{,}1827$

Ahora hay que *agrupar* el valor de A en $\mathbf{f}_{\times\mathbf{A}}$ (realizar el sumatorio $\sum_a$)

- Así, obtenemos una tabla $\mathbf{f}_{\overline{\mathbf{A}}}(R,T)$ haciendo $\mathbf{f}_{\overline{\mathbf{A}}}(v_1,v_2) = \sum_a \mathbf{f}_{\times\mathbf{A}}(a,v_1,v_2)$ para cada valor v_1 de R y v_2 de T, y variando a en los posibles valores de A
- Llamaremos a esta operación *agrupamiento*
- Hemos *eliminado* la variable A
- Una vez realizada la agrupación, guardamos $\mathbf{f}_{\overline{\mathbf{A}}}$ y nos olvidamos de $\mathbf{f_M}$, $\mathbf{f_J}$ y $\mathbf{f_A}$

R	T	a	$\neg a$	$\mathbf{f}_{\overline{\mathbf{A}}}(R,T)$
r	t	$0{,}70 \times 0{,}90 \times 0{,}95 = 0{,}5985$	$0{,}01 \times 0{,}05 \times 0{,}05 = 0{,}00003$	0,59853
r	$\neg t$	$0{,}70 \times 0{,}90 \times 0{,}94 = 0{,}5922$	$0{,}01 \times 0{,}05 \times 0{,}06 = 0{,}00003$	0,59223
$\neg r$	t	$0{,}70 \times 0{,}90 \times 0{,}29 = 0{,}1827$	$0{,}01 \times 0{,}05 \times 0{,}71 = 0{,}00036$	0,18306
$\neg r$	$\neg t$	$0{,}70 \times 0{,}90 \times 0{,}001 = 0{,}00063$	$0{,}01 \times 0{,}05 \times 0{,}999 = 0{,}0005$	0,00113

Eliminación de T:

- Notemos por $\mathbf{f}_{\overline{\mathbf{T}}}(R)$ al resultado de multiplicar $\mathbf{f}_{\overline{\mathbf{A}}}(R,T)$ por $\mathbf{f_T}(T)$ y agrupar por T

R	t	$\neg t$	$\mathbf{f}_{\overline{\mathbf{T}}}(R)$
r	$0{,}59853 \times 0{,}002 = 0{,}001197$	$0{,}59223 \times 0{,}998 = 0{,}591046$	0,59224
$\neg r$	$0{,}18306 \times 0{,}002 = 0{,}000366$	$0{,}00113 \times 0{,}998 = 0{,}001128$	0,00149

- Podemos olvidarnos de $\mathbf{f}_{\overline{\mathbf{A}}}$ y $\mathbf{f_T}$

Queda la variable R:

- Al ser la variable de consulta, no se agrupa, sólo multiplicamos los factores en los que aparece
- Multiplicamos $\mathbf{f}_{\overline{\mathbf{T}}}(R)$ y $\mathbf{f}_{\mathbf{R}}(R))$ para obtener $\mathbf{f}_{\times\mathbf{R}}(R)$

R	$\mathbf{f}_{\overline{\mathbf{T}}}(R) \times \mathbf{f}_{\mathbf{R}}(R)$
r	$0{,}59224 \times 0{,}001 = 0{,}00059$
$\neg r$	$0{,}00149 \times 0{,}999 = 0{,}00149$

- Finalmente, normalizamos la tabla (para que sus componentes sumen 1): $\mathbf{P}(R|j,m) = \langle 0{,}28417; 0{,}71583\rangle$

La tabla finalmente devuelta es la distribución $\mathbf{P}(R|j,m)$. Es decir, $P(r|j,m) = 0{,}28417$ y $P(\neg r|j,m) = 0{,}71583$

Optimización: Variables irrelevantes

Previo al algoritmo de eliminación de variables, se suele realizar un paso previo para descartar las variables irrelevantes para consulta.

Con carácter general, toda variable que no sea antecesor (en la red) de alguna de las variables de consulta o de evidencia, es irrelevante para la consulta y por tanto, puede ser eliminada.

Entrada: una v.a. X de consulta, un conjunto de valores observados $\mathbf{e}$ para la variables de evidencia y una red bayesiana

Salida: $\mathbf{P}(X|\mathbf{e})$

Algoritmo de eliminación de variables

```
FUNCION INFERENCIA_ELIMINACION_VARIABLES(X,e,RED)
1. Sea RED' el resultado de eliminar de RED las variables
   irrelevantes para la consulta realizada
2. Sea FACTORES igual a conjunto de los factores
   correspondientes a cada variable de RED'
4. Sea VARS_ORD el conjunto de las variables de RED' que
   no sean de evidencia, ordenado según un orden de
   eliminación
5. PARA cada VAR en VARS_ORD HACER
   5.1 Si VAR es de consulta, eliminar de FACTORES los
       factores en los que aparece VAR, e incluir el factor
       resultante de multiplicarlos
   5.3 Si VAR no es de consulta, eliminar de FACTORES los
       factores en los que aparece VAR, e incluir el factor
       resultante de  multiplicarlos y agruparlos por VAR
6. Devolver la NORMALIZACION del único factor
   que queda en FACTORES
```

PD: La complejidad del algoritmo de eliminación de variables depende del tamaño del mayor factor obtenido durante el proceso. En ello influye el orden en el que se consideran las variables; para ello, es posible usar un criterio heurístico para elegir el orden de eliminación.

Si la red está simplemente conectada (si a lo sumo hay un camino no dirigido entre cada dos nodos), se puede probar que la complejidad es lineal en el tamaño de la red (nº de entradas en sus tablas).

En general, el algoritmo tiene complejidad exponencial en el peor de los casos; cuando la inferencia exacta se hace inviable es necesario usar métodos aproximados de inferencia: Muestreo.

Muestreo

Entendemos por muestreo o "sampling" respecto de una distribución de probabilidad, a los métodos de generación de eventos, de manera que la probabilidad de generación de un evento nuevo coincide con la que indica la distribución.

Se generaliza esta idea para diseñar un procedimiento de muestreo respecto de una DCC que representa una red bayesiana.

El muestreo más sencillo: consideremos una v.a. booleana A tal que $\mathbf{P}(A) = \langle \theta, 1 - \theta \rangle$

- Basta con tener un método de generación aleatoria y uniforme de números $x \in [0, 1]$
- Si se genera $x < \theta$, se devuelve a; en caso contrario $\neg a$
- En el límite, el número de muestras generadas con valor a entre el número de muestras totales es θ

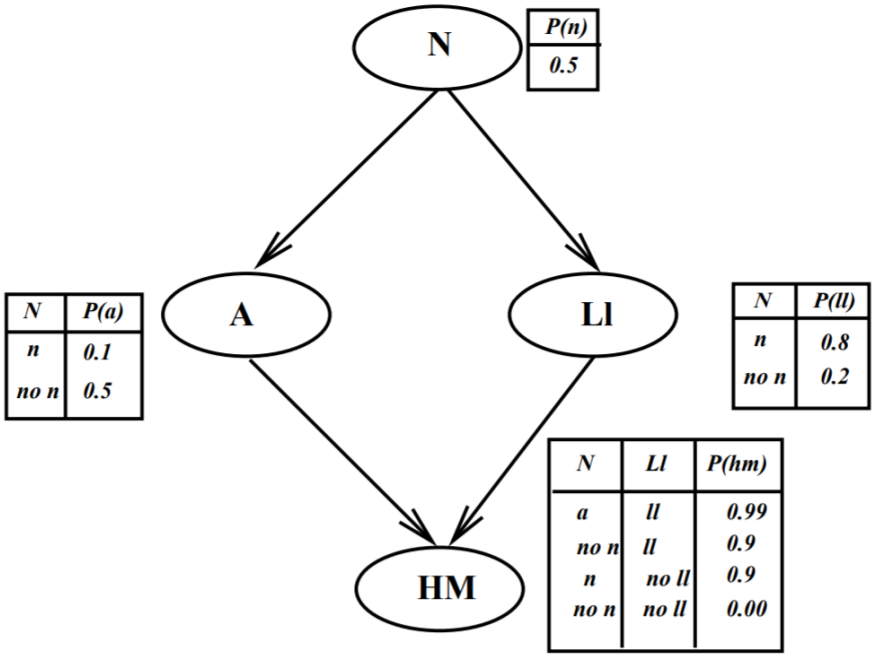
Ejemplo:

Consideremos las siguientes variables aleatorias booleanas:

- N: el cielo está nublado
- A: el aspersor se ha puesto en marcha
- LL: ha llovido
- HM: la hierba está mojada

Las relaciones causales y el conocimiento probabilístico asociado están reflejadas en la siguiente red bayesiana

- Nótese que es un ejemplo de red que no está simplemente conectada.



Veamos cómo generar un evento aleatorio completo a partir de la DCC que especifica la red anterior

- Muestreo de $\mathbf{P}(N) = \langle 0{,}5; 0{,}5\rangle$; aplicando el método anterior obtenemos n
- Muestreo de $\mathbf{P}(A|n) = \langle 0{,}1; 0{,}9\rangle$; obtenemos $\neg a$
- Muestreo de $\mathbf{P}(LL|n) = \langle 0{,}8; 0{,}2\rangle$; obtenemos ll
- Muestreo de $\mathbf{P}(HM|\neg a, ll) = \langle 0{,}9; 0{,}1\rangle$; obtenemos hm

El evento generado por muestreo ha sido $\langle n, \neg a, ll, hm\rangle$

La probabilidad de generar ese evento es $0{,}5 \times 0{,}9 \times 0{,}8 \times 0{,}9 = 0{,}324$, ya que cada muestreo individual se realiza independientemente

Muestreo con rechazo

Entrada: una v.a. X de consulta, un conjunto de valores observados **e** para la variables de evidencia, una *RED* bayesiana (con n variables) y número N de muestras totales a generar

Algoritmo de muestreo con rechazo

```
FUNCION MUESTREO-CON-RECHAZO(X,e,RED,N)
1. Sea N[X] un vector con una componente por cada posible
   valor de la variable de consulta X, inicialmente todas a 0
2. PARA k = 1,...,N HACER
   2.1 Sea (y1,...,yn) igual a MUESTREO-A-PRIORI(RED)
   2.2 SI y = (y1,...,yn) es consistente con e entonces HACER
       N[x] igual a N[x] + 1, donde en el evento y la v.a. X
       toma el valor x
3. Devolver NORMALIZA(N[X])
```

Este algoritmo devuelve una estimación consistente de P(X|e) pero se rechazan demasiadas muestras sobre todo si el nº de variables de evidencia es grande.

Ponderación con verosimilitud

Es posible diseñar un algoritmo que sólo genere muestras consistentes con la observación **e**.

Los valores de las variables de evidencia no se generan (son fijados de antemano), pero no todos los eventos generados "pesan" lo mismo: aquellos en los que la evidencia es más improbable deben "pesar" menos. Cada evento generado va acompañado de un peso igual al producto de las probabilidades condicionadas de cada valor que aparezca en **e**.

Supongamos que queremos calcular $\mathbf{P}(LL|a, hm)$; para generar cada muestra con su correspondiente peso w, hacemos lo siguiente ($w = 1{,}0$ inicialmente):

- Muestreo de $\mathbf{P}(N) = \langle 0{,}5; 0{,}5\rangle$; obtenemos n
- Como A es una variable de evidencia (cuyo valor es a) hacemos w igual a $w \times P(a|n)$ (es decir, $w = 0{,}1$)
- Muestreo de $\mathbf{P}(LL|n) = \langle 0{,}8; 0{,}2\rangle$; obtenemos ll
- HM es una variable de evidencia (con valor hm); por tanto, hacemos w igual a $w \times P(hm|a, ll)$ (es decir, $w = 0{,}099$)

Por tanto, el muestreo devolvería $\langle n, a, ll, hm\rangle$ con un peso igual a 0,099

Entrada: una v.a. X de consulta, un conjunto de valores observados **e** para la variables de evidencia, una *RED* bayesiana (con n variables) y un número N de muestras totales a generar

Algoritmo de ponderación por verosimilitud

FUNCION PONDERACION-POR-VEROSIMILITUD(X,e,RED,N)
1. **Sea** $W[X]$ **un vector con una componente para cada posible valor de la variable de consulta** X, **inicialmente todas a 0**
2. **PARA** $k = 1, \ldots, N$ **HACER**
   - **2.1 Sea** $[(y1, \ldots, yn), w]$ **igual a MUESTRA-PONDERADA(RED,e)**
   - **2.2 Hacer** $W[x]$ **igual a** $W[x] + w$, **donde en el evento** y **la v.a.** X **toma el valor** x
3. **Devolver NORMALIZA(**$W[X]$**)**

Obtenición de muestras ponderadas

FUNCION MUESTRA-PONDERADA(RED,e)
1. **Hacer** $w = 1{,}0$
2. **PARA** $i = 1, \ldots, n$ **HACER**
   - **2.1 SI la variable** X_i **tiene valor** x_i **en** e **ENTONCES** $w = w \times p(X_i = x_i | padres(X_i))$
   - **2.2 SI NO, sea** x_i **el resultado de un muestreo de** $P(X_i | padres(X_i))$
3. **Devolver** $[(x_1, \ldots, x_n), w]$

Este algoritmo devuelve una estimación consistente de la probabilidad buscada; en el caso de que haya muchas variables de evidencia, el algoritmo podría degradarse ya que la mayoría de las muestras tendrán un peso infinitesimal.

PD: Existen otros algoritmos de inferencia aproximada en redes bayesianas más eficientes y sofisticados cómo el algoritmo de Monte Carlo de Cadenas de Markov.