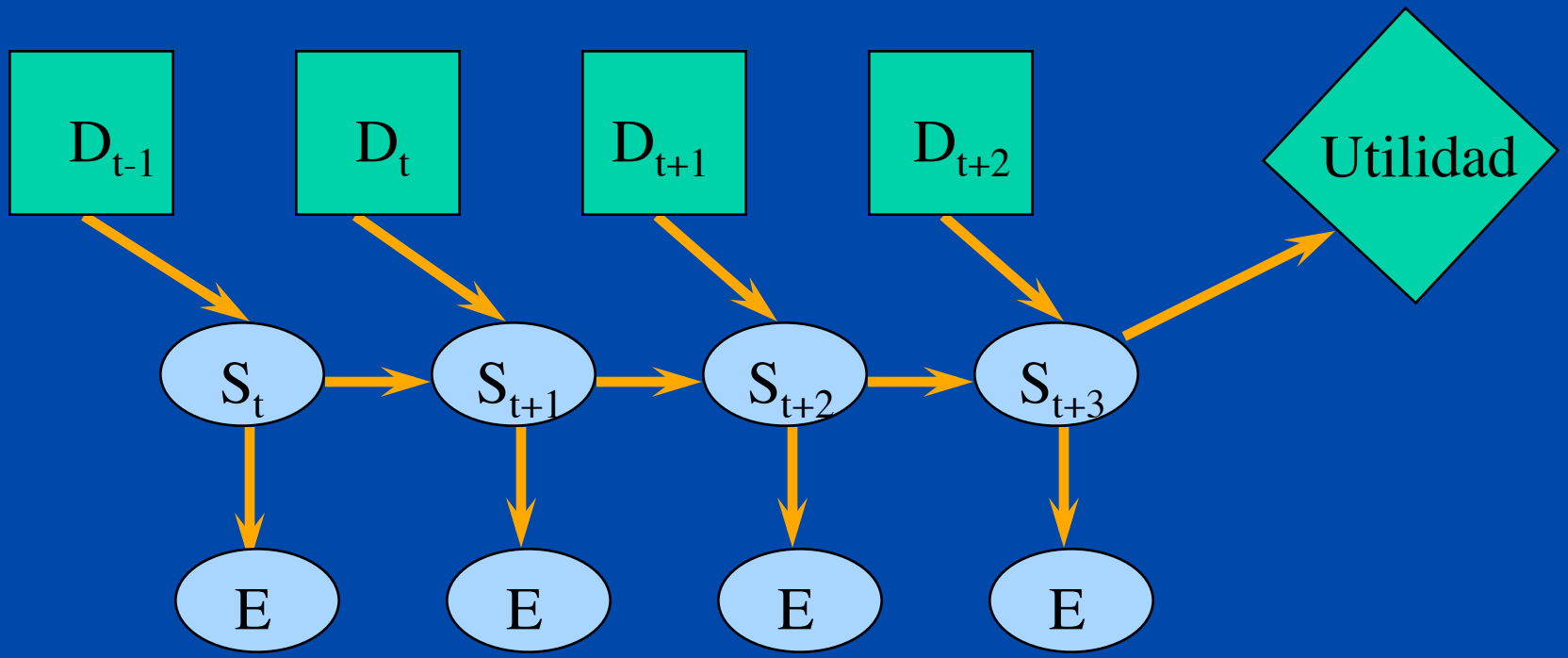
Modelos Gráficos Probabilistas



L. Enrique Sucar

INAOE

De datos … a información … a decisiones

De millones de pixeles a pocos conceptos (autos, personas, …)

De conceptos a decisiones (frenar, dar vuelta, seguir, …)

¿Porqué el enfoque bayesiano?

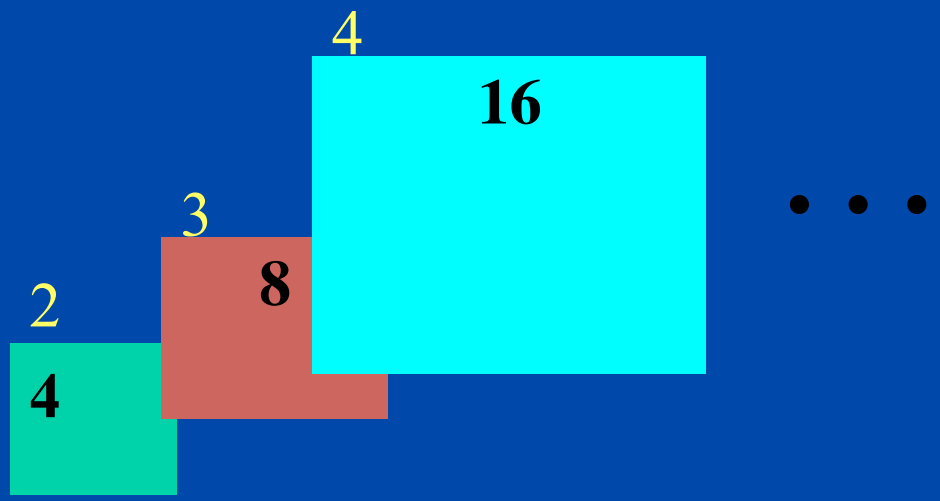
Combinar la información (datos) con la experiencia (conocimiento) para reducir la incertidumbre

¿Porqué modelos gráficos?

Los requerimientos de espacio y de cómputo (para estimar probabilidades) crecen exponencialmente con el número de variables

100

10^{30}



Al considerar las relaciones de dependencia e independencia entre variables se tiene un reducción sustancial en requerimiento de cómputo (memoria y operaciones)

Contenido

1. Introducción
2. Clasificadores Bayesianos
3. Redes Bayesianas
4. Inferencia en Redes Bayesianas
5. Aprendizaje de Redes Bayesianas
6. Redes bayesianas dinámicas
7. Ejemplos de aplicaciones

Información General

- Página del curso en INAOE:

http://ccc.inaoep.mx/~esucar/Clases-pgm/pgm.html

- Correo:

esucar@inaoep.mx

1: Introducción

"Vivir es el arte de derivar conclusiones suficientes de premisas insuficientes"

[S. Butler]

Modelos Probabilistas

- Los modelos probabilistas tienen varias varias ventajas:
  - Tienen una clara interpretación y sólidos fundamentos teóricos
  - Pueden hacer uso de las técnicas y herramientas desarrolladas en el área de probabilidad y estadística
  - Permiten un representación declarativa - separando el conocimiento del razonamiento

Modelos Probabilistas

- Sin embargo, si construimos un modelo probabilista en forma “directa” nos enfrentamos al problema de complejidad computacional - el modelo crece exponencialmente con el número de variables
- Los modelos gráficos probabilistas hacen uso de las independencias entre las variables para poder resolver problemas complejos en una forma mucho más eficiente

Modelos Gráficos Probabilistas

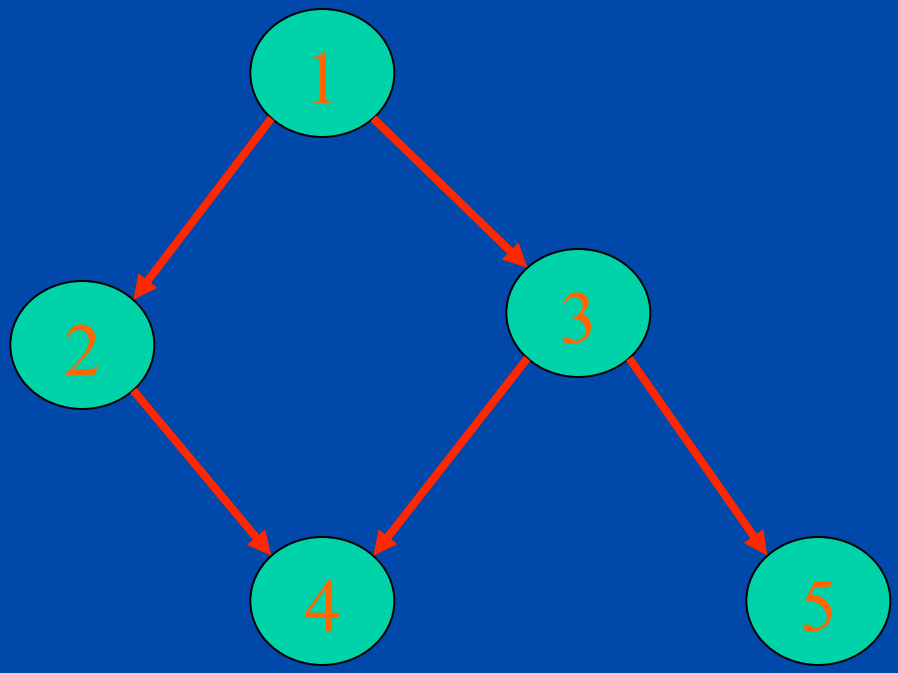
- Representaciones basadas en modelar la incertidumbre con variables aleatorias relacionadas mediante un modelo gráfico (dependencias)
- Muchas técnicas se pueden englobar dentro de este tipo de modelos

Modelos Gráficos Probabilistas

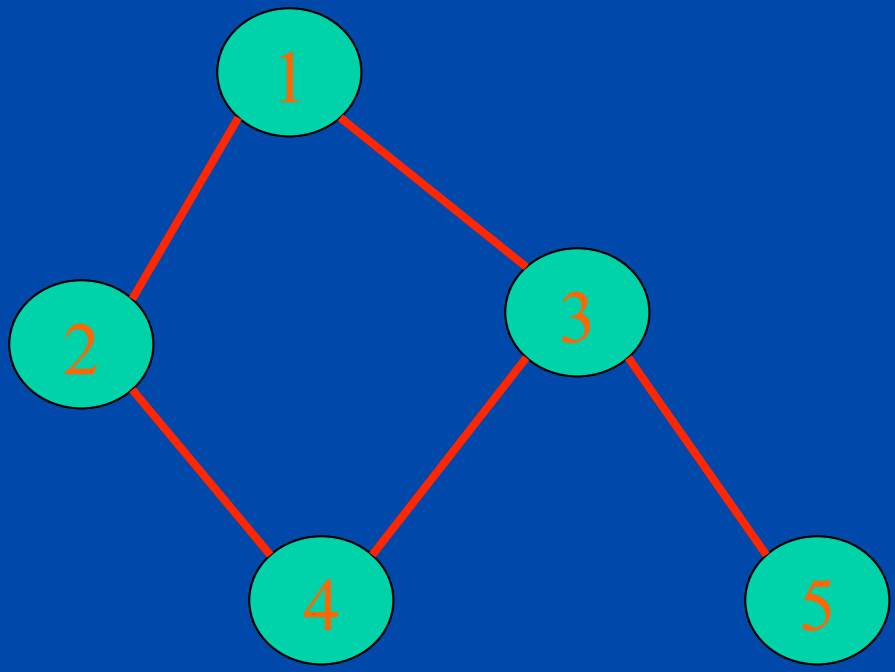
- Podemos clasificar los modelos gráficos probabilistas en 3 dimensiones principales:
  - Dirigidos vs. No-dirigidos
  - Estáticos vs. Dinámicos
  - Probabilista vs. Decisiones

Modelos Gráficos Probabilistas

- Dirigido

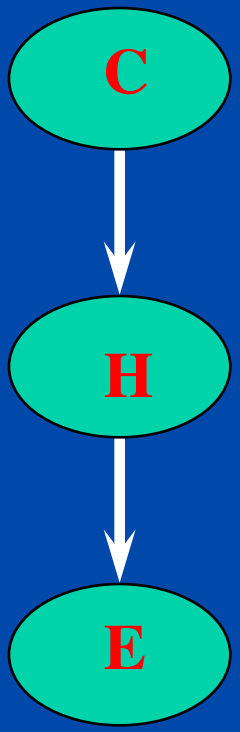


- No-dirigido

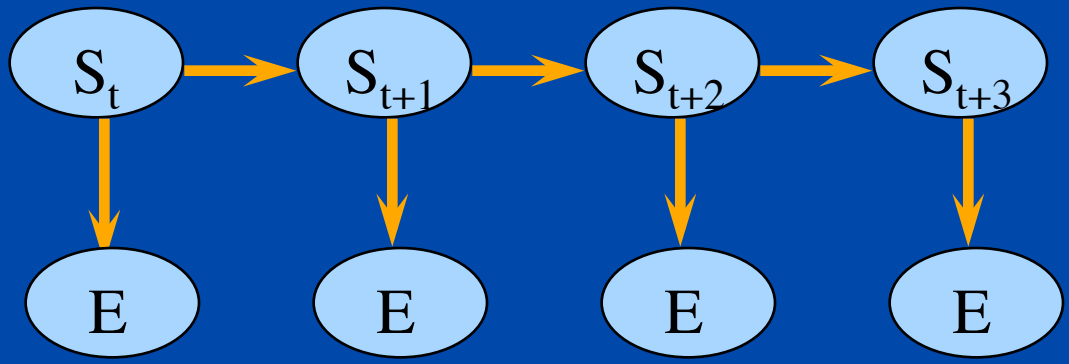


Modelos Gráficos Probabilistas

- Estático

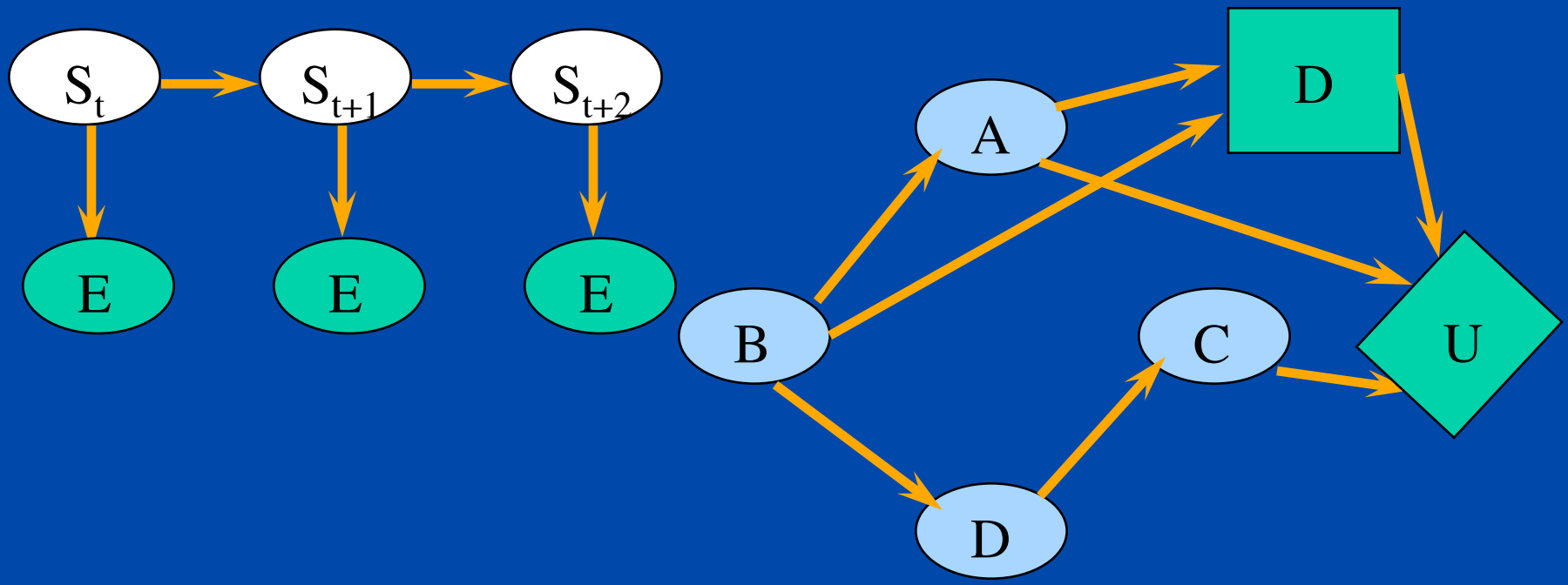


- Dinámico



Modelos Gráficos Probabilistas

- Probabilista
- Decisiones



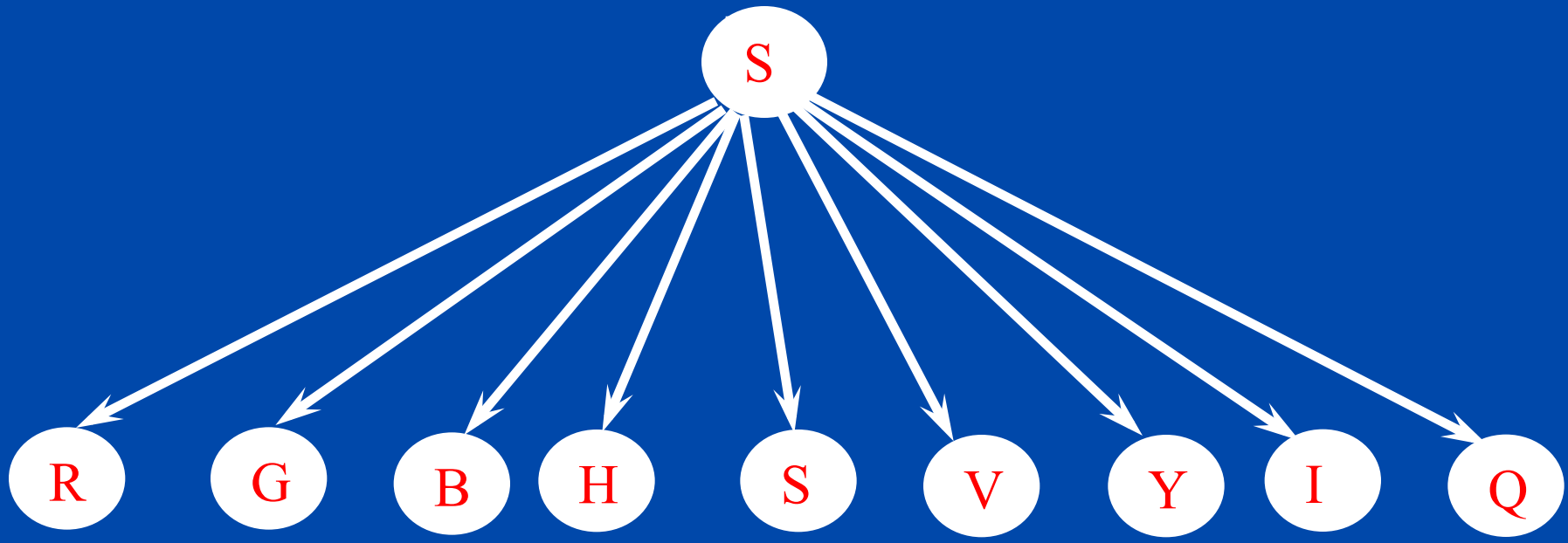
Modelos Gráficos Probabilistas

Modelo	D/N	E/D	P/D
Clasificador bayesiano	D/N	S	P
Cadenas ocultas de Markov	D	D	P
Campos / Redes de Markov	N	S	P
Redes bayesianas	D	S	P
Redes bayesianas dinámicas	D	D	P
Filtros de Kalman	D	D	P
Redes de Decisión	D	S	D
Procesos de decisión de Markov	D	D	D
MDPs parcialmente observables	D	D	D

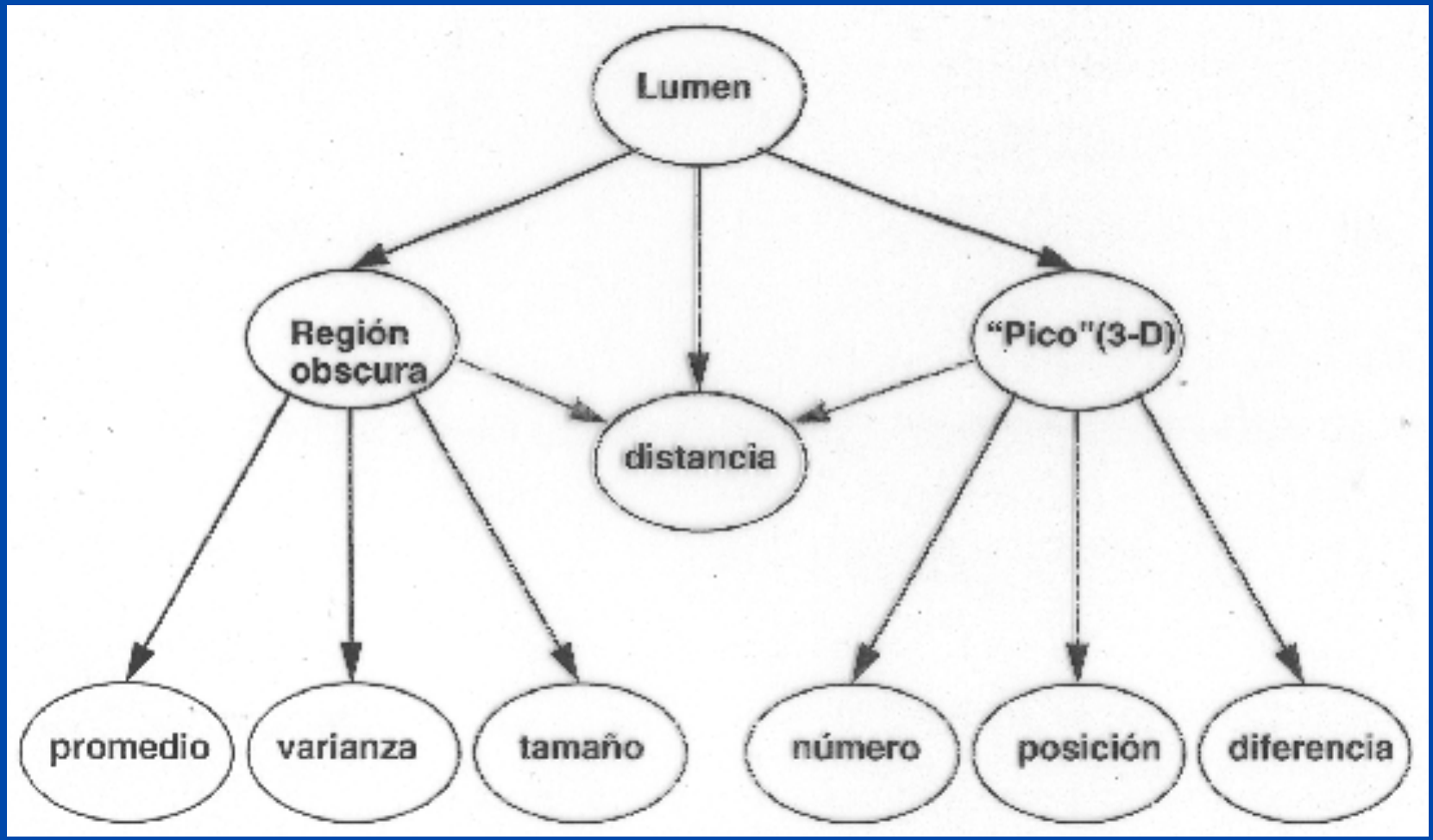
Aplicaciones

- Existen actualmente muchas aplicaciones de PGMs, como:
  - Reconocimiento de voz y ademanes
  - Diagnóstico industrial
  - Modelado del estudiante
  - Navegación robótica
  - Validación de sensores
  - Recuperación de información
  - …

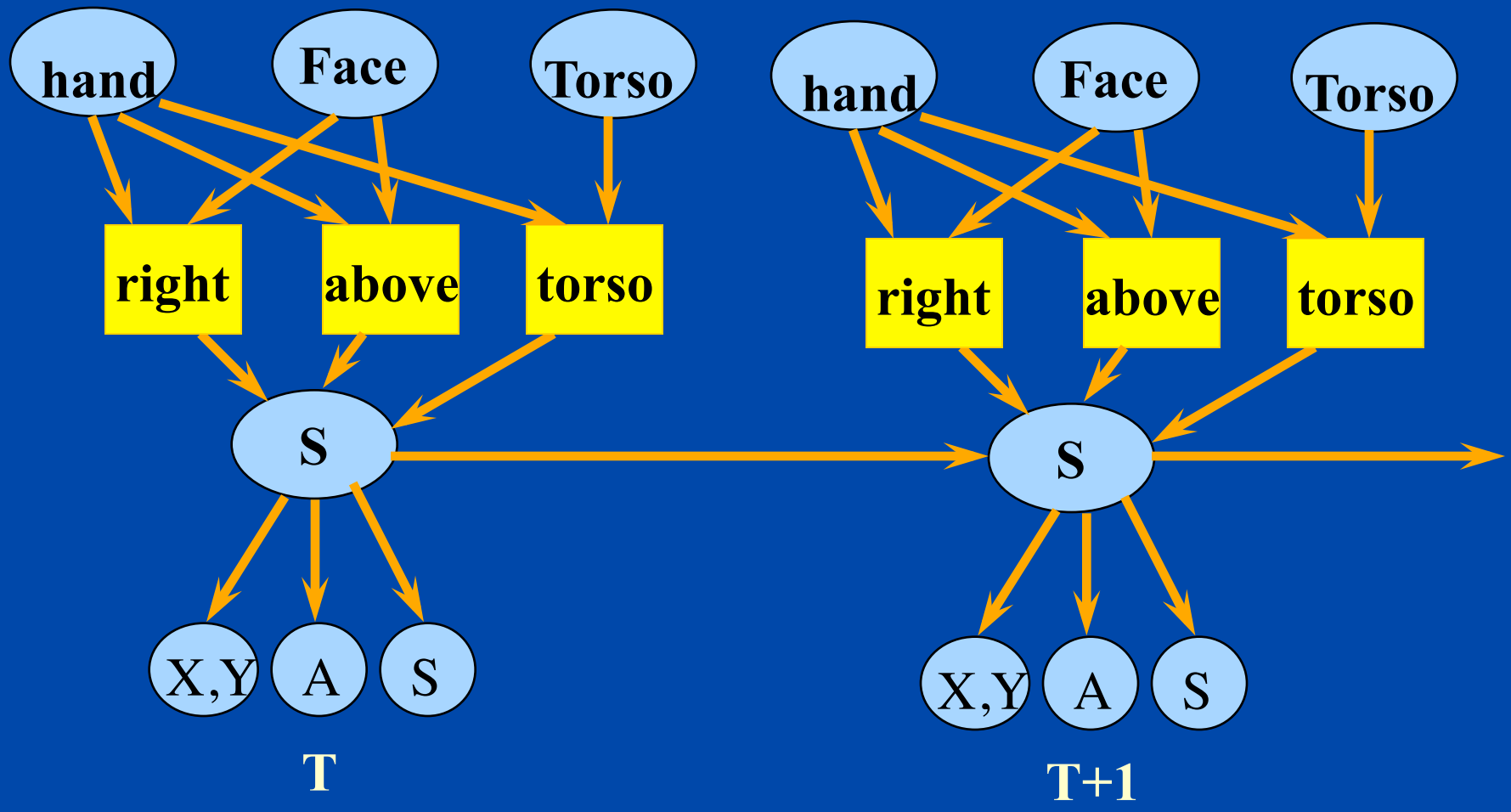
Clasificación de piel en imágenes



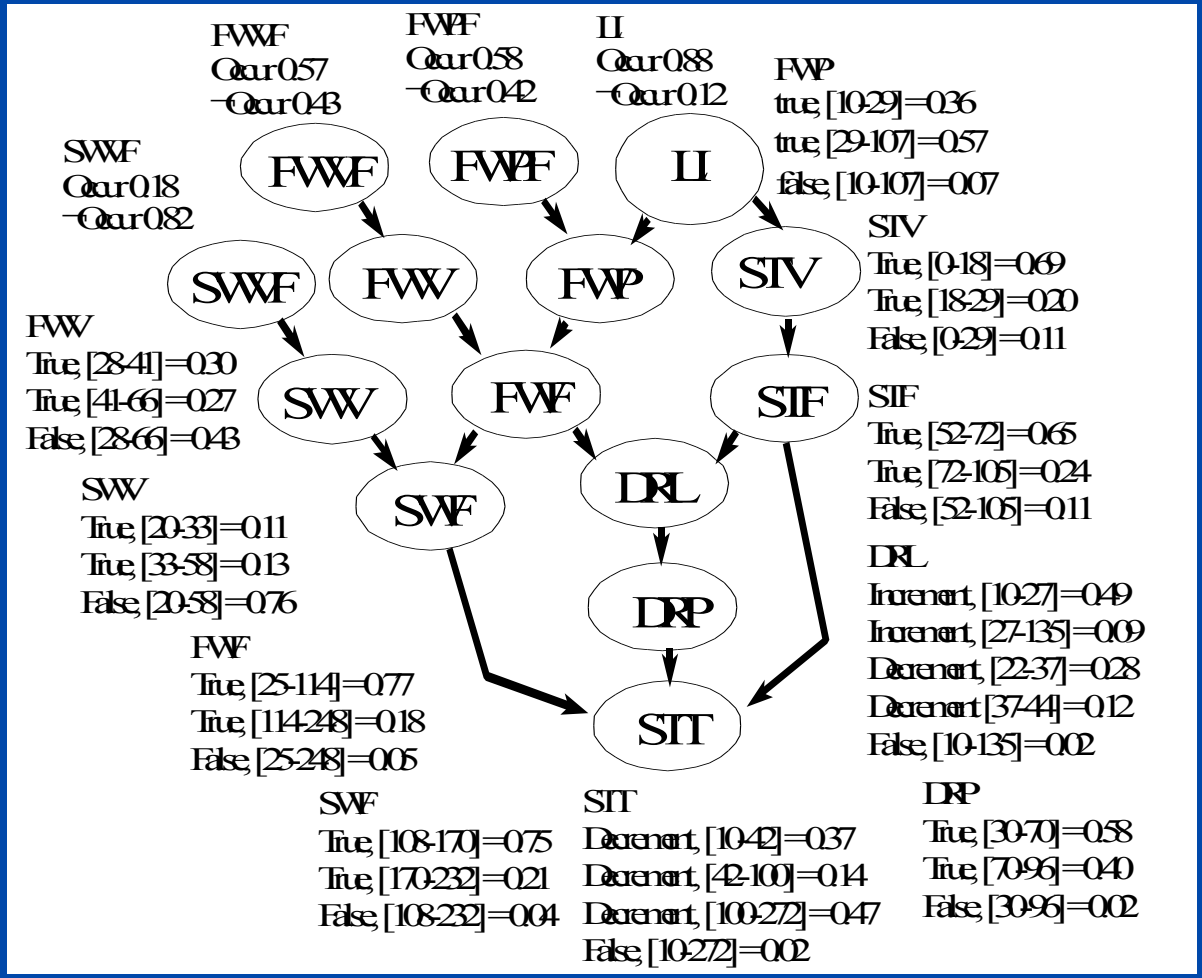
Medicina - endoscopía



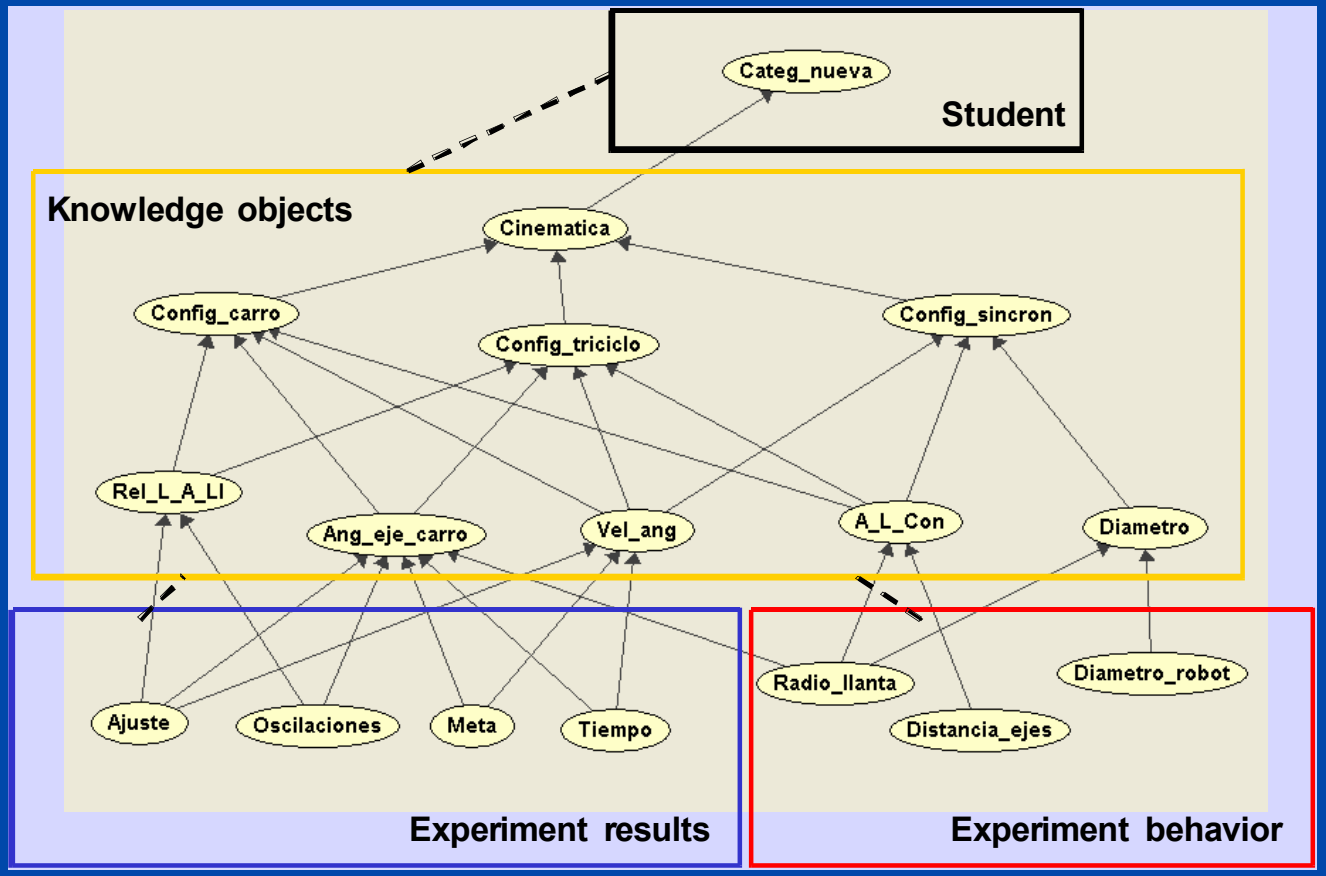
Reconocimiento de ademanes



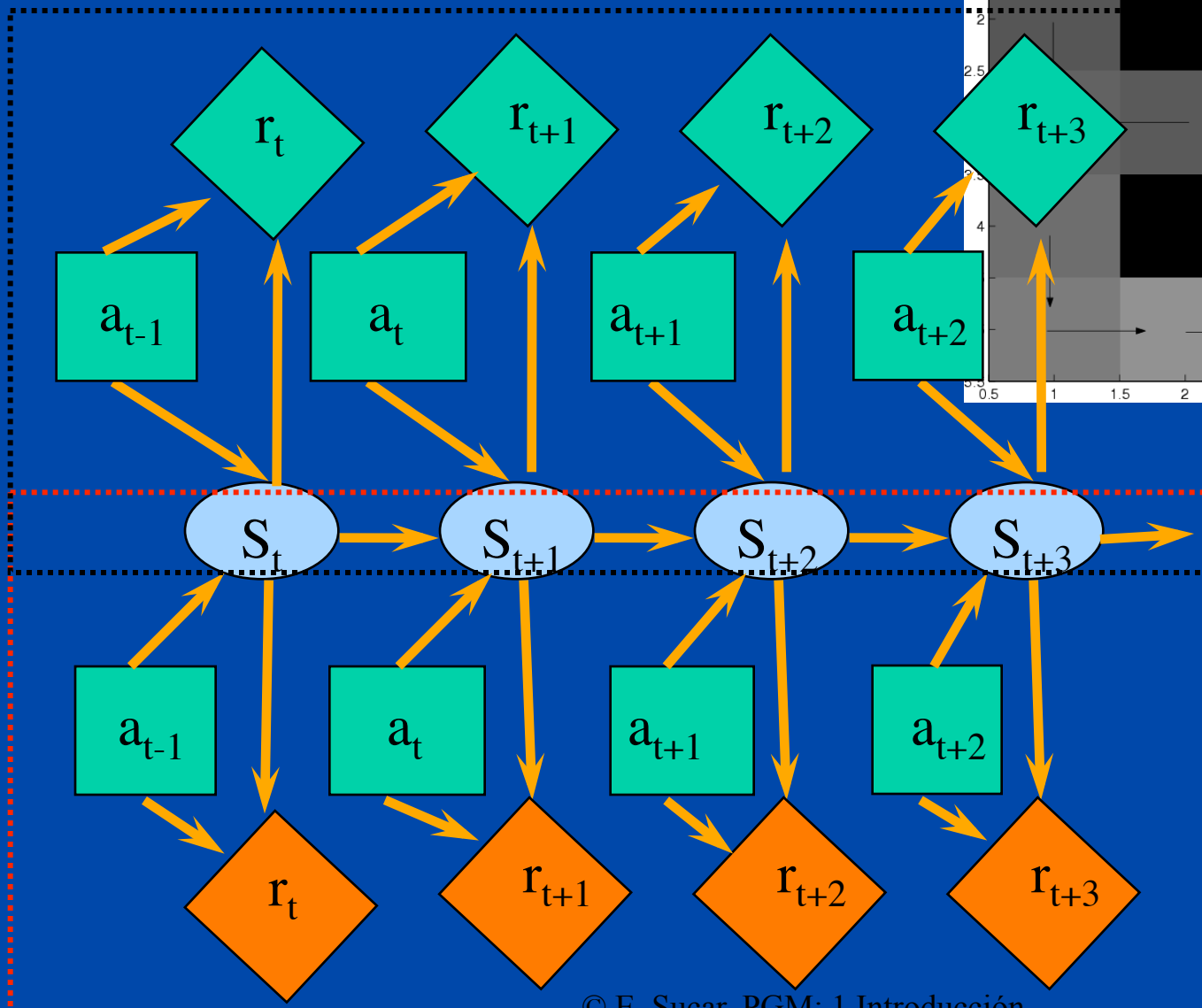
Diagnóstico industrial

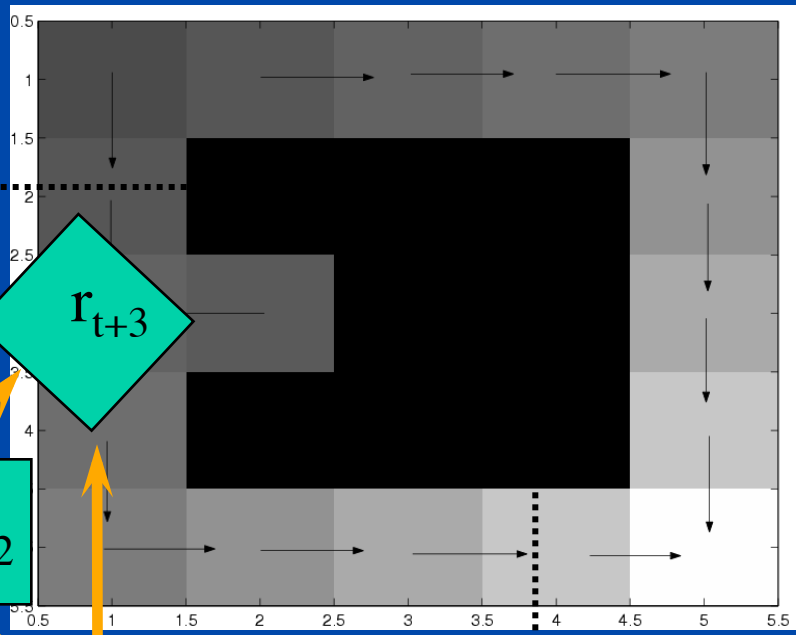


Modelado del estudiante

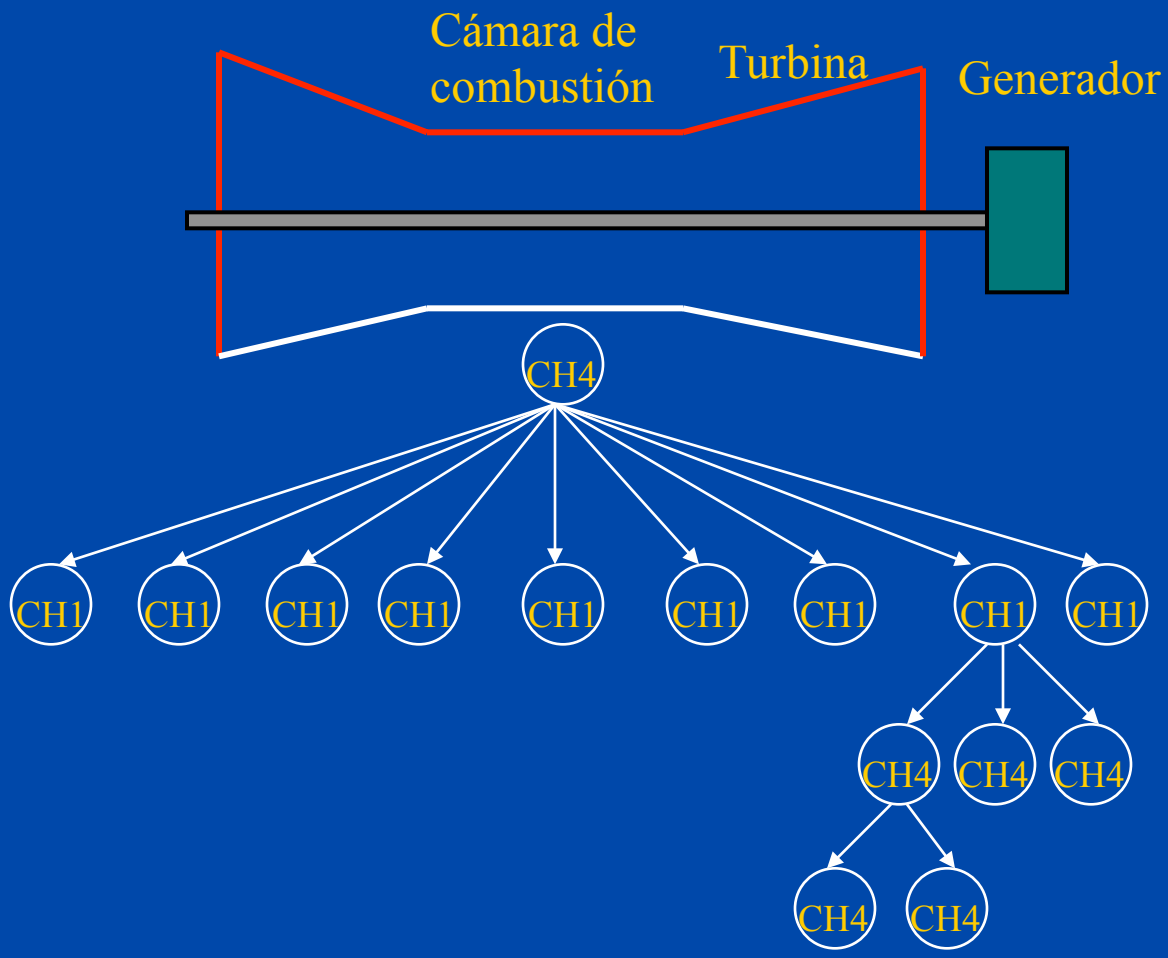


Navegación robótica

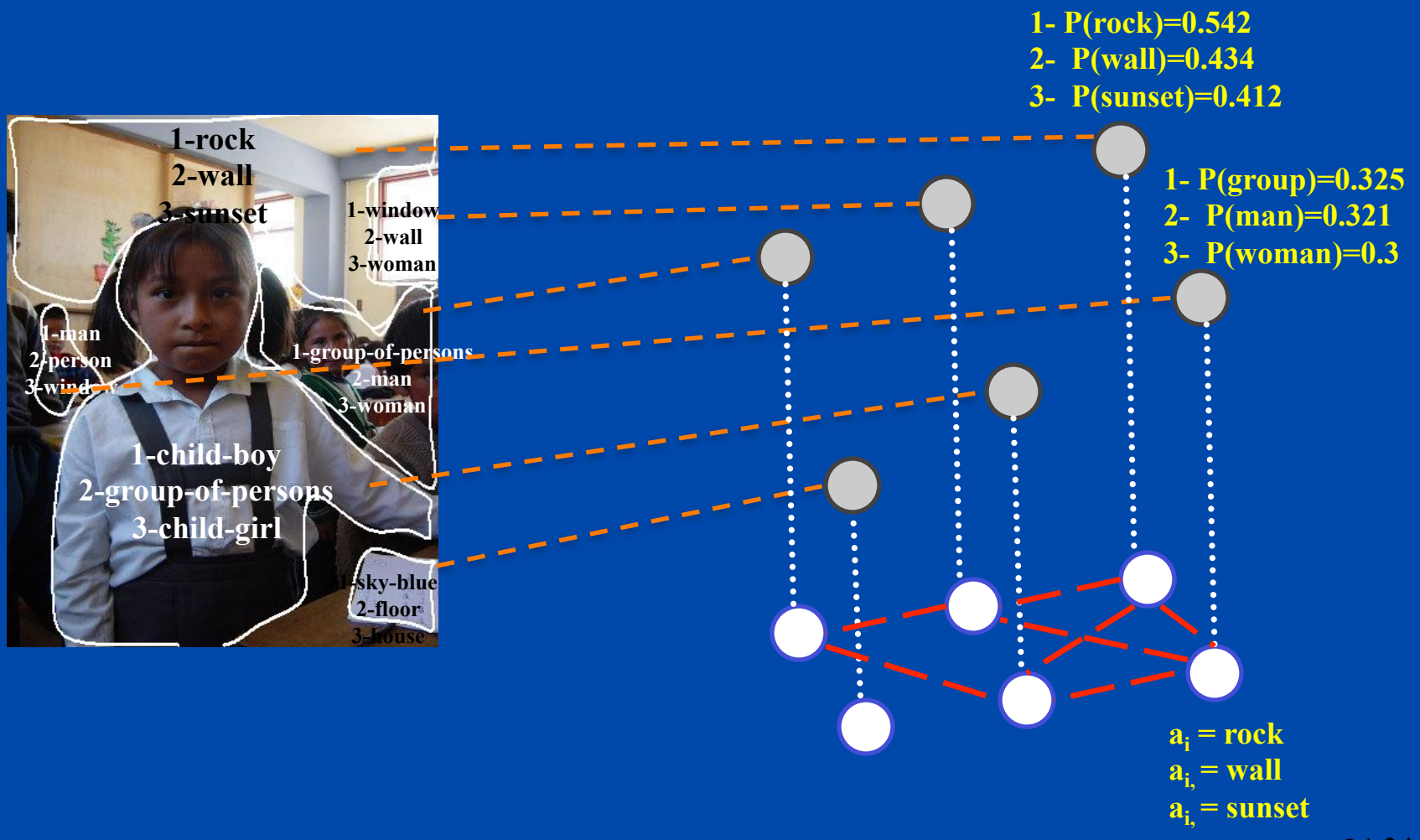




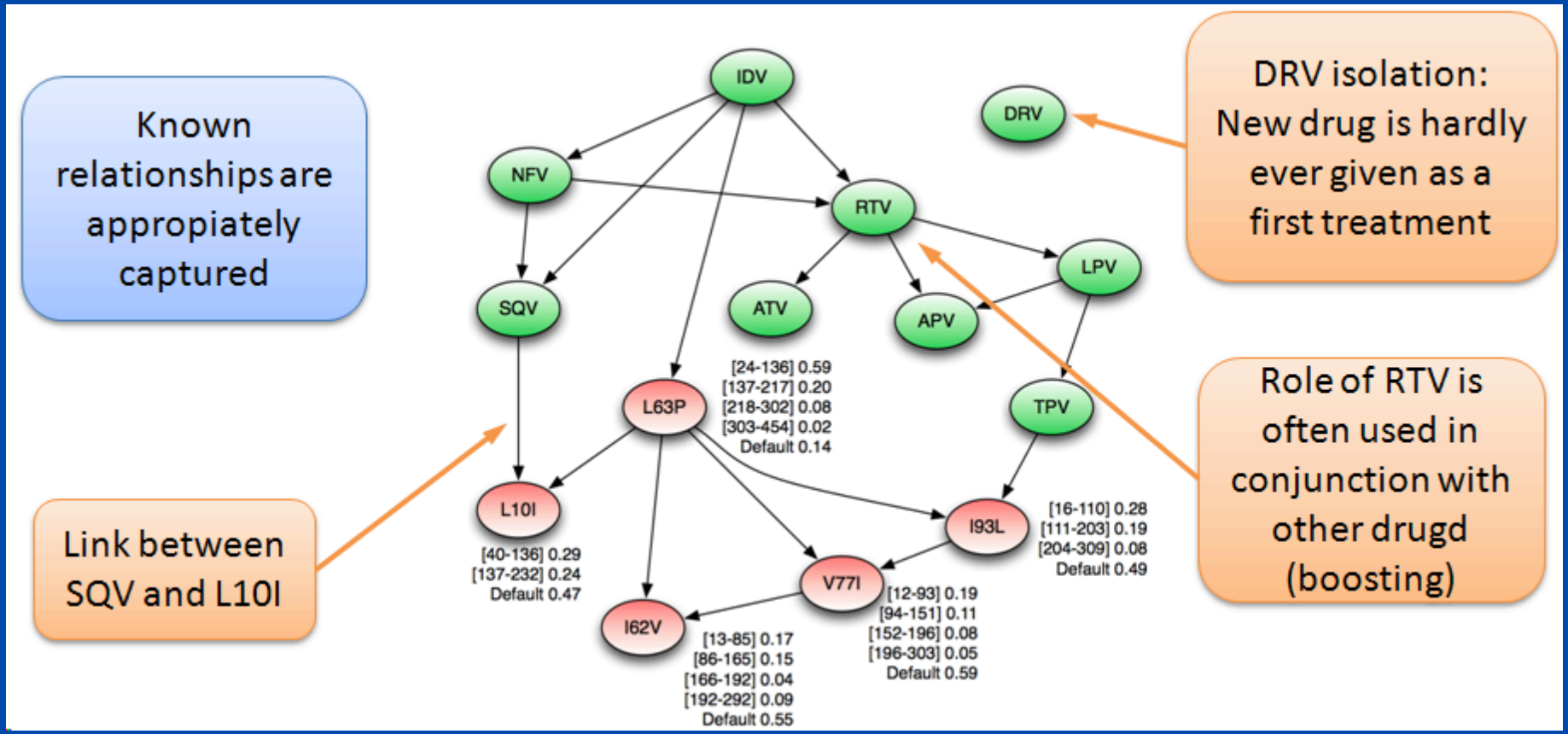
Validación de sensores



Anotación de Imágenes



Predicción de mutaciones del virus del VIH



2: Clasificadores Bayesianos

" La teoría de probabilidades en el fondo no es nada más que sentido común reducido a cálculos ..."

[Pierre Simon Laplace, 1819]

Clasificación

- El problema de clasificación (supervisada) consiste en obtener el valor más probable de una variable (hipótesis) dados los valores de otras variables (evidencia, atributos)

$$\text{Arg}_H \left[\text{Max } P(H \mid E_1, E_2, \ldots E_N) \right]$$
$$\text{Arg}_H \left[\text{Max } P(H \mid \mathbf{E}) \right]$$

$$\mathbf{E} = \{E_1, E_2, \ldots E_N\}$$

Regla de Bayes

- La probabilidad posterior se puede obtener en base a la regla de Bayes:

$$P(H \mid \mathbf{E}) = P(H)\, P(\mathbf{E} \mid H) / P(\mathbf{E})$$

$$P(H \mid \mathbf{E}) = P(H)\, P(\mathbf{E} \mid H) / \Sigma_i\, P(\mathbf{E} \mid H_i)\, P(H_i)$$

- Normalmente no se requiere saber el valor de probabilidad, solamente el valor más probable de H

Clasificador bayesiano simple

- Estimar la probabilidad: $P(\mathbf{E} \mid H)$ es complejo, pero se simplifica si se considera que los atributos son independientes dada la hipotesis:
  $P(E_1, E_2, ...E_N \mid H) = P(E_1 \mid H)\, P(E_2 \mid H) \ldots P(E_N \mid H)$
- Por lo que la probabilidad de la hipótesis dada la evidencia puede estimarse como:

$$P(H \mid E_1, E_2, ...E_N) = \frac{P(H)\, P(E_1 \mid H)\, P(E_2 \mid H) \ldots P(E_N \mid H)}{P(\mathbf{E})}$$

- Esto se conoce como el clasificador bayesiano simple

Clasificador bayesiano simple

- Como veíamos, no es necesario calcular el denominador:

$$P(H \mid E_1, E_2, \ldots E_N) \sim$$

$$P(H)\, P(E_1 \mid H)\, P(E_2 \mid H) \ldots P(E_N \mid H)$$

- P(H) se conoce como la *probabilidad a priori*, $P(E_i \mid H)$ es la probabilidad de los atributos dada la hipótesis (*verosimilitud*), y $P(H \mid E_1, E_2, \ldots E_N)$ es la *probabilidad posterior*

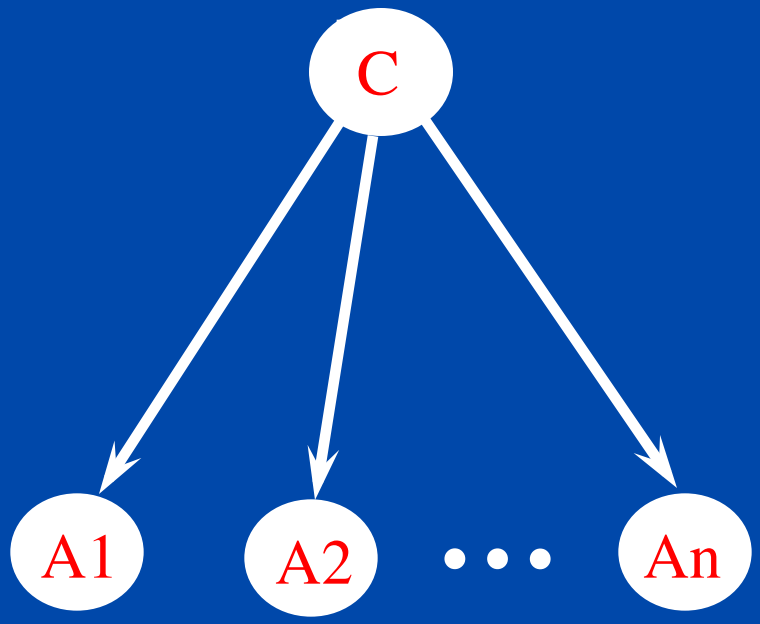
Ventajas

- Bajo tiempo de clasificación
- Bajo tiempo de aprendizaje
- Bajos requerimientos de memoria
- “Sencillez”
- Buenos resultados en muchos dominios

Limitaciones

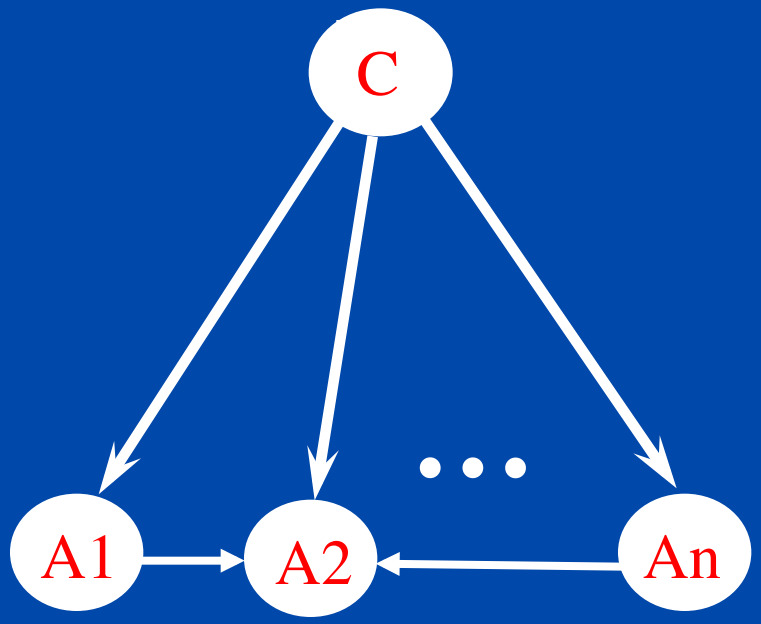
- En muchas ocasiones la suposición de independencia condicional no es válida
- Para variables continuas, existe el problema de discretización
- Alternativas – dependencias:
  - Estructuras que consideran dependencias
  - Mejora estructural del clasificador
- Alternativas – variables continuas:
  - Descriminador lineal (variables gaussianas)
  - Técnicas de discretización

CBS – modelo gráfico



Extensiones

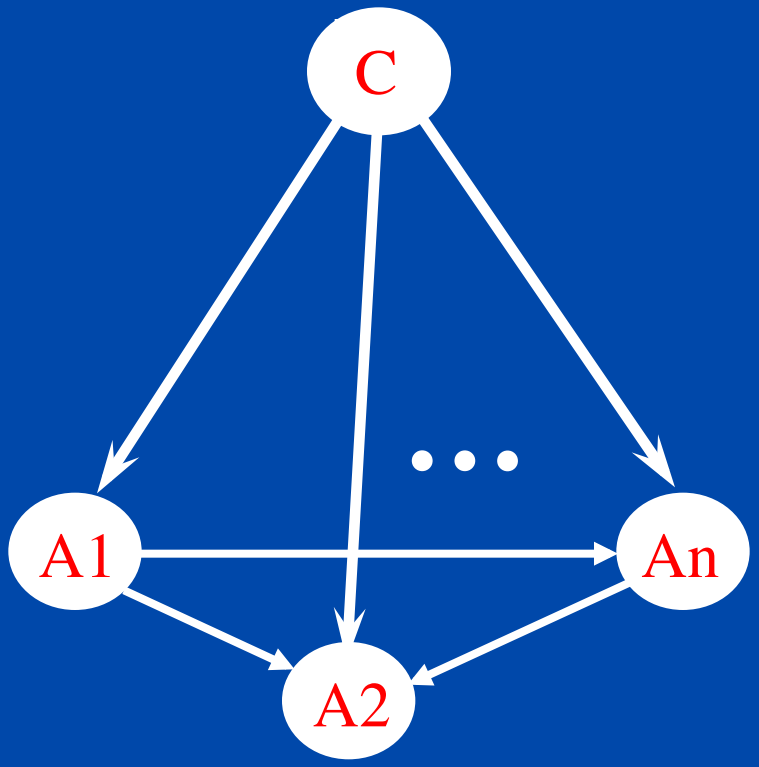
- TAN



Se incorpora algunas dependencias entre atributos mediante la construcción de un "árbol" entre ellos

Extensiones

- BAN



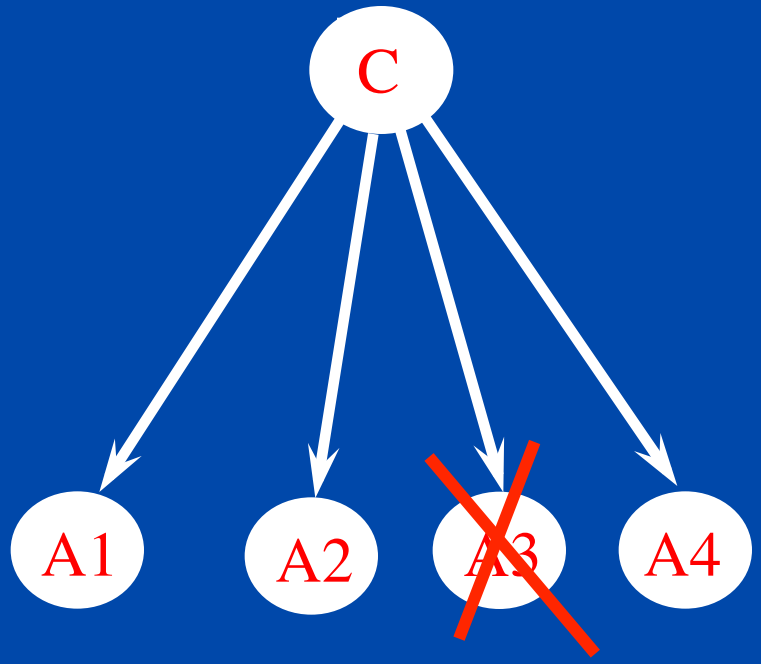
Se incorpora una “red” para modelar las dependencias entre atributos

Mejora estructural

- Otra alternativa para mejorar el CBS es partir de una estructura “simple” y modificarla mediante:
  - Eliminación de atributos irrelevantes (selección de atributos)
  - Verificación de las relaciones de independencia entre atributos y alterando la estructura:
    - Eliminar nodos
    - Combinar nodos
    - Insertar nodos

Eliminación de atributos

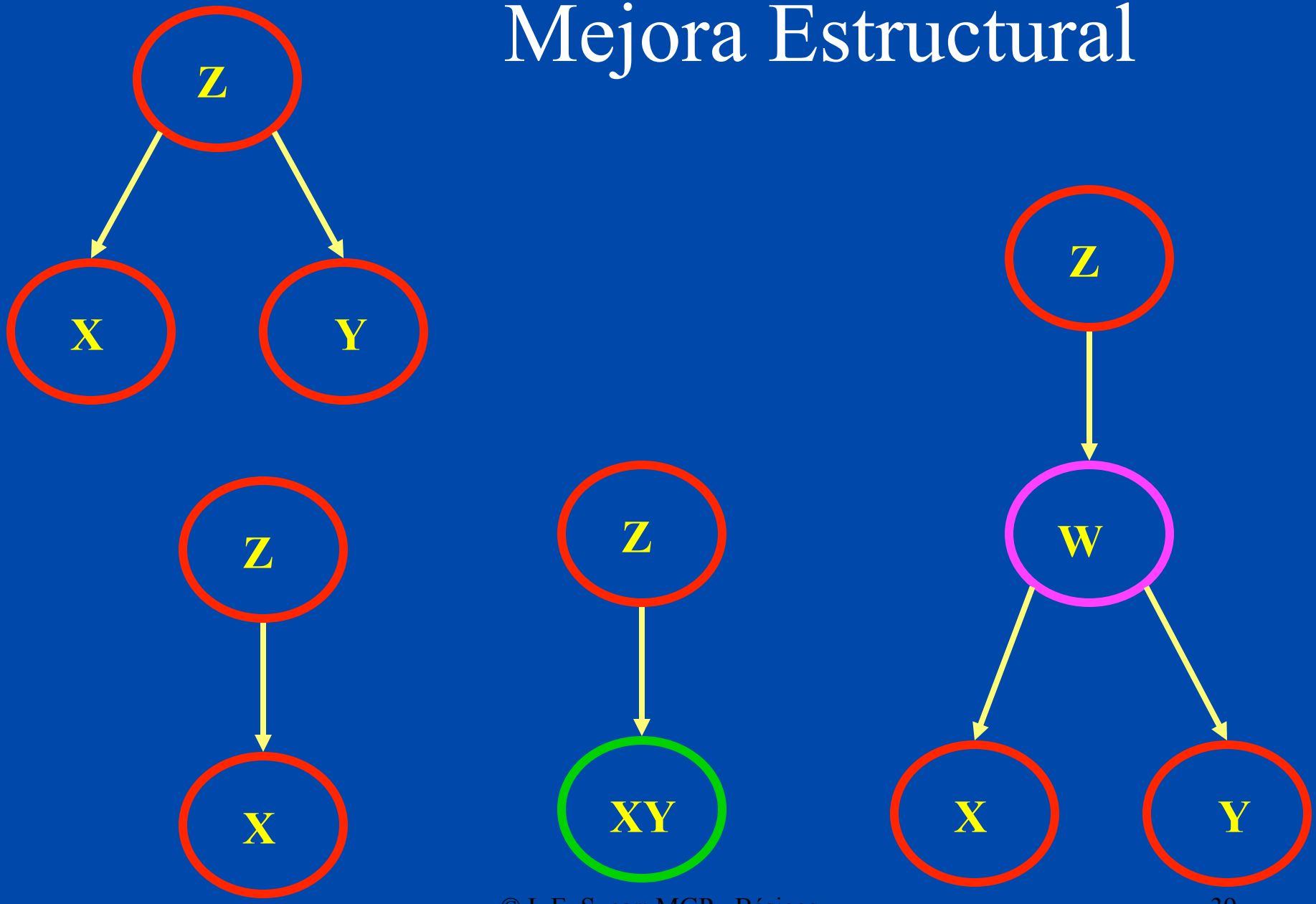
- Medir la "dependencia" entre la clase y atributos (por ejemplo con la información mutua), y eliminar aquellos con "poca" aportación



Mejora estructural

- Medir la dependencia entre pares de atributos dada la clase (por ejemplo mediante la información mutua condicional), alterar la estructura si hay 2 dependientes:
  1. Eliminación: quitar uno de los dos (redundantes)
  2. Unión: juntar los 2 atributos en uno, combinando sus valores
  3. Inserción: insertar un atributo "virtual" entre la clase y los dos atributos que los haga independientes.

Mejora Estructural



Atributos redundantes

- Prueba de dependencia entre cada atributo y la clase
- Información mutua:

$$I(C, Ai) = \Sigma P(C, Ai) \log [P(C, Ai) / P(C) P(Ai)]$$

- Eliminar atributos que no provean información a la clase

Atributos dependientes

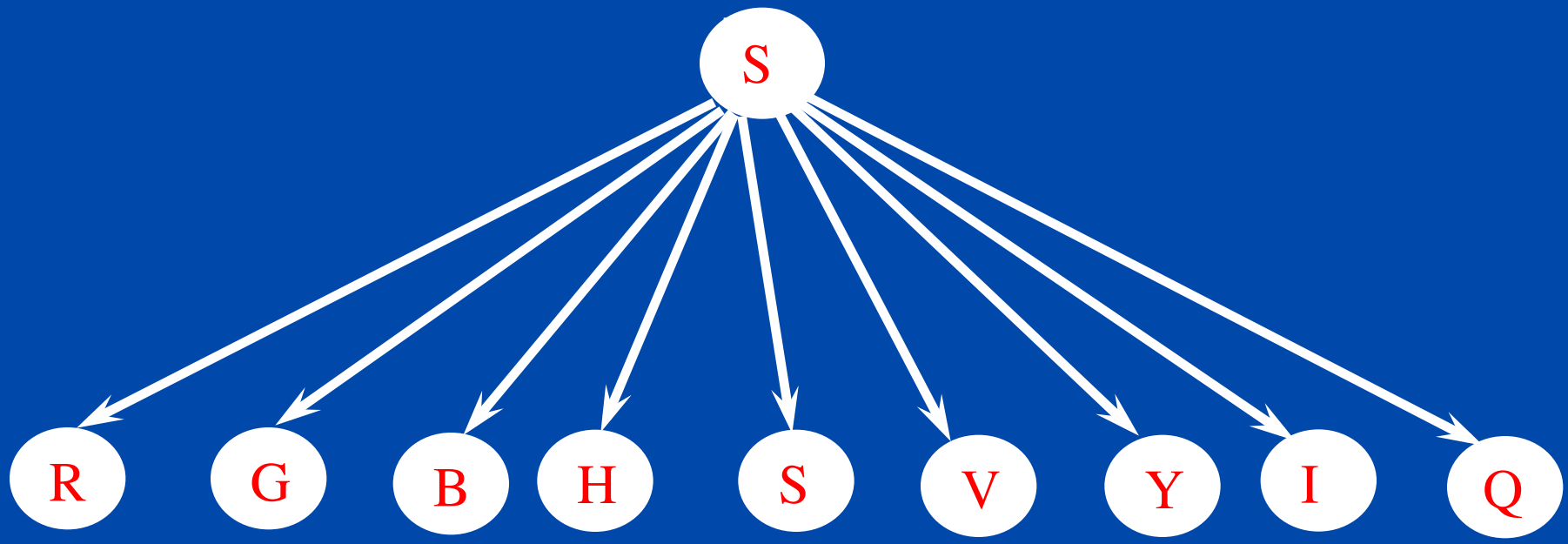
- Prueba de independencia de cada atributo dada la clase
- Información mutua condicional

$$I(Ai, Aj \mid C) = \sum P(Ai,Aj|C) \log [P(Ai,Aj|C) / P(Ai|C)\, P(Aj|C)]$$

- Eliminar, unir o (insertar) atributos

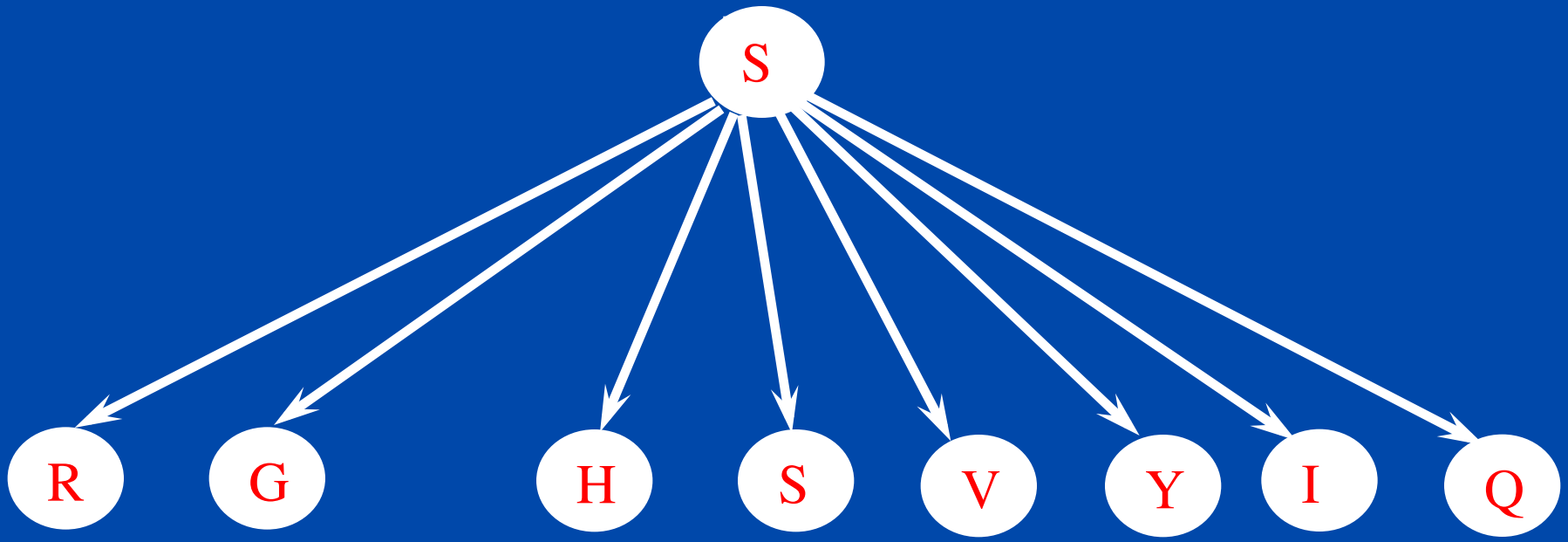
Ejemplo: clasificación de piel

- 9 atributos - 3 modelos de color: RGB, HSV, YIQ

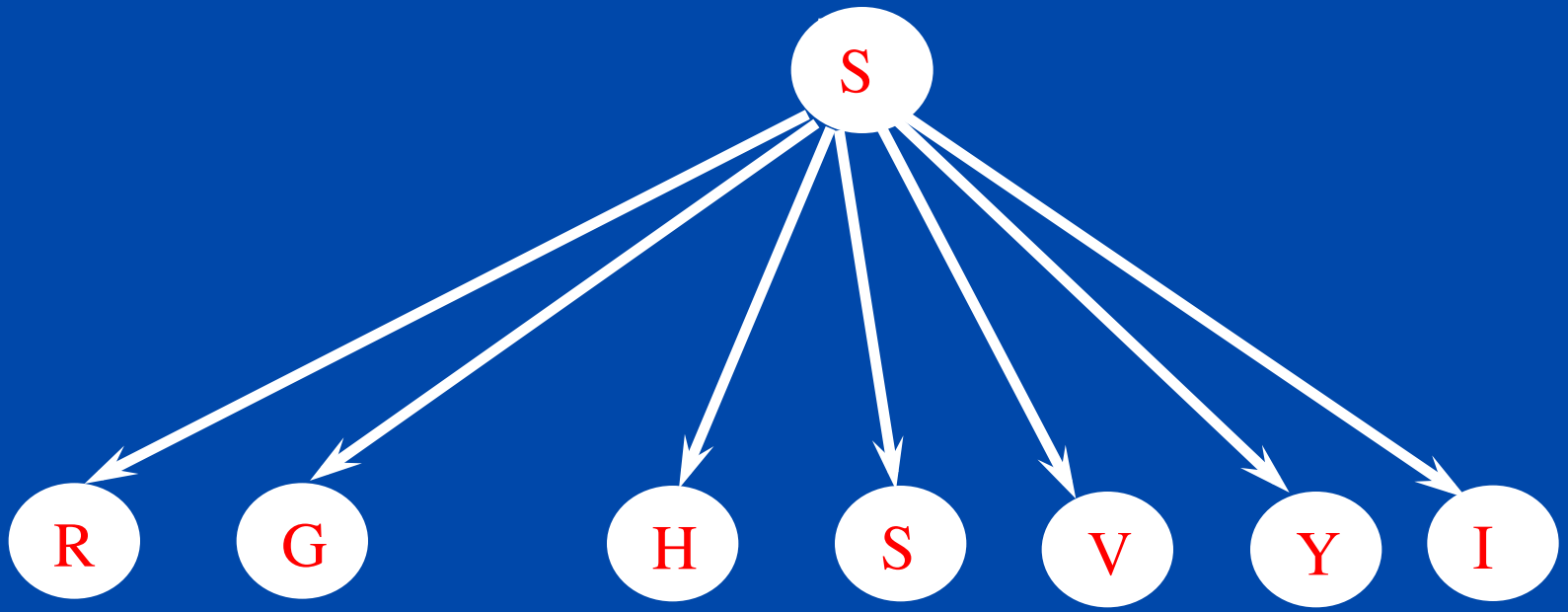


Mejora estructural

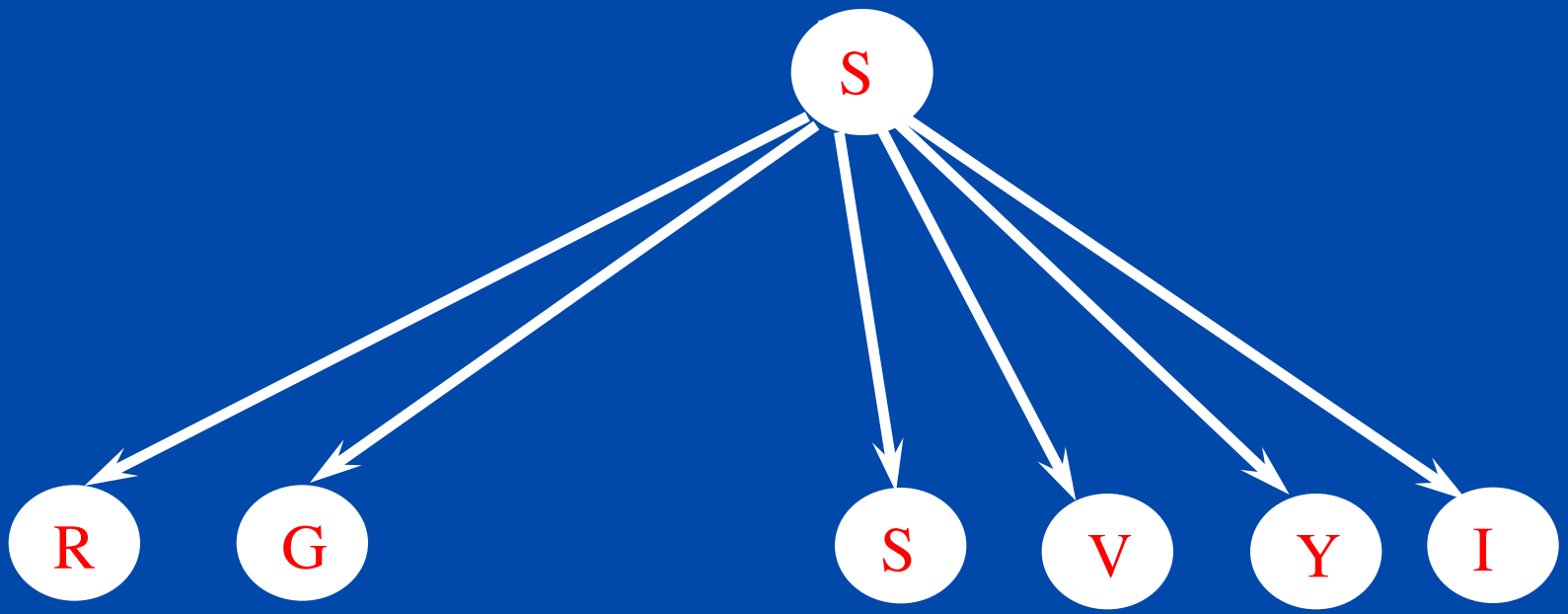
Elimina B



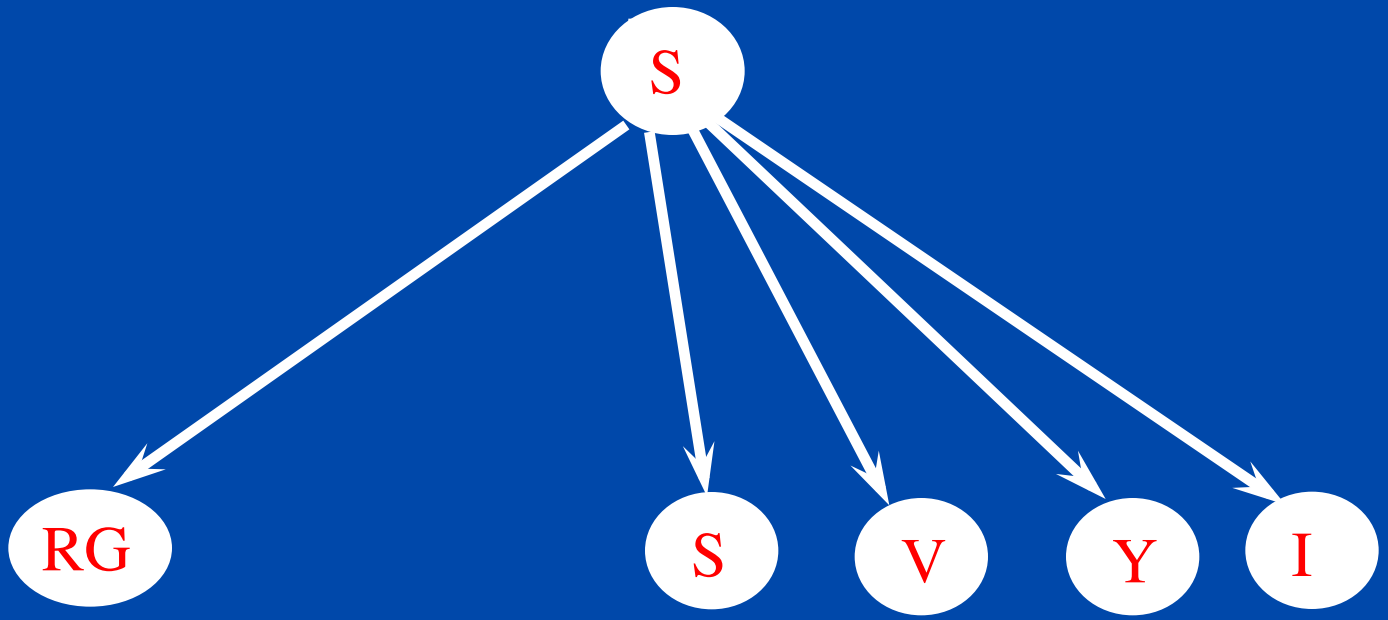
Elimina Q



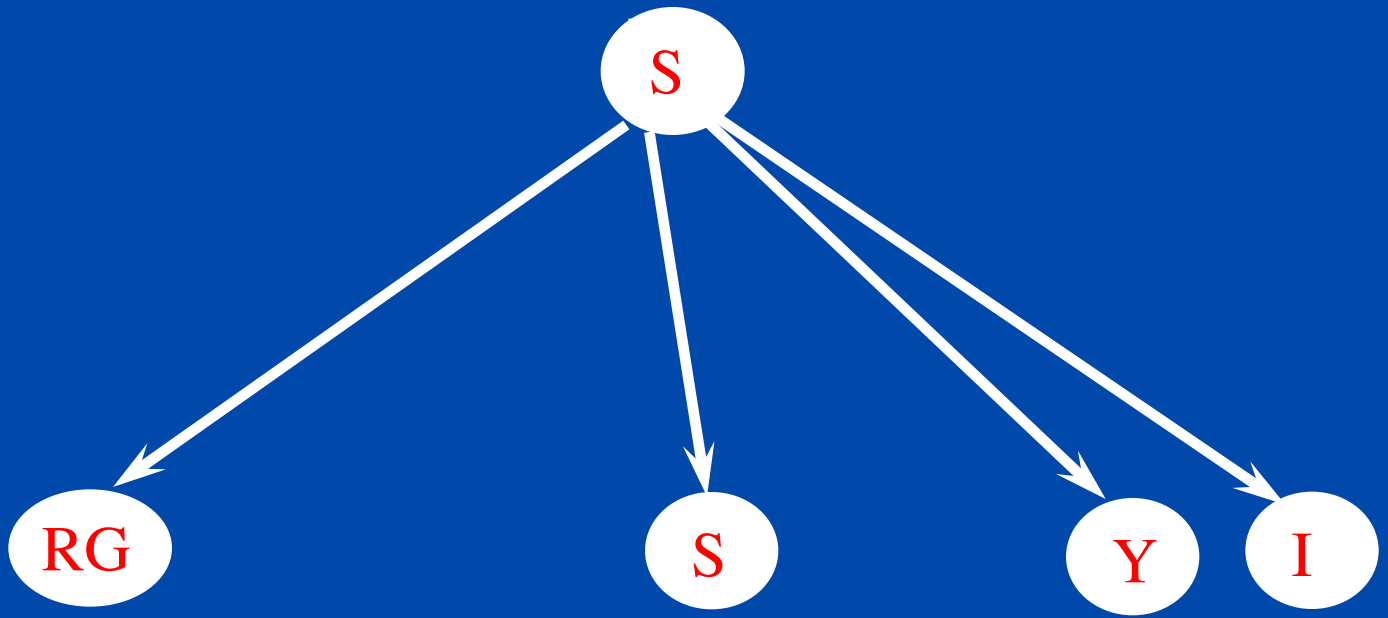
Elimina H



Unir RG

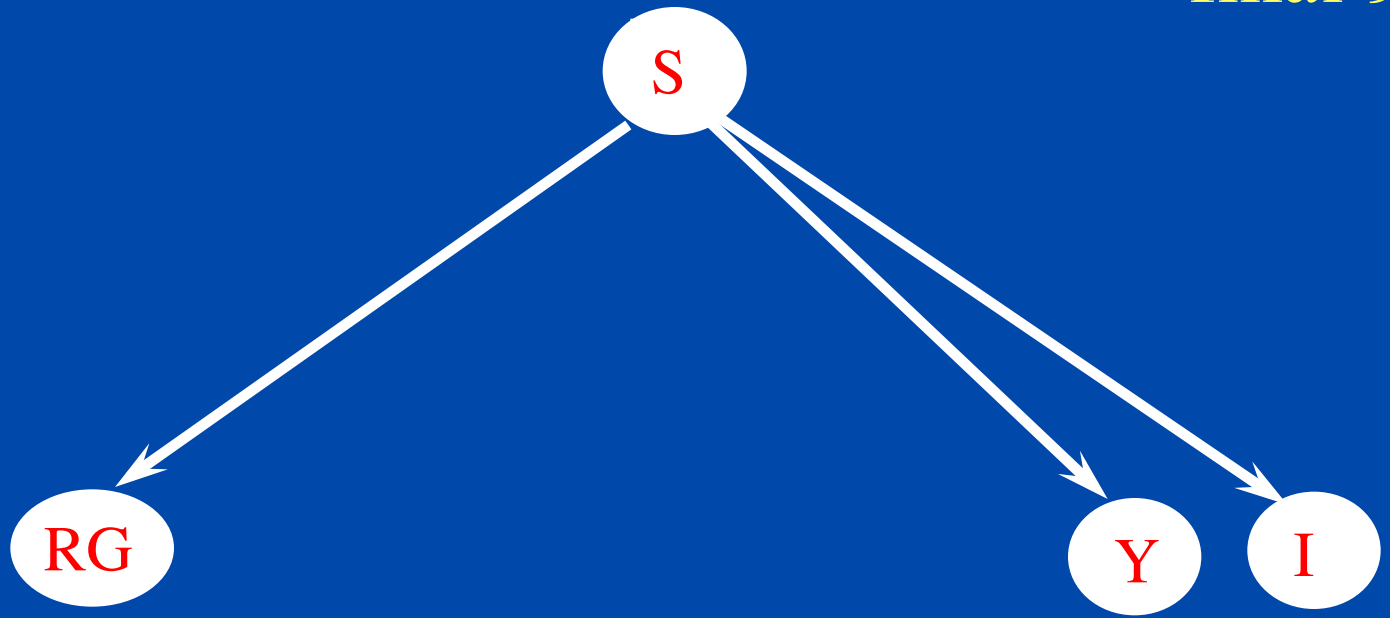


Elimina V



Elimina S

Exactitud: inicial 94%
final 98%



Clasificadores Multidimensionales

- En un clasificador multidimensional se desea maximizar la probabilidad del conjunto posible de clases dados los atributos:

$$\text{Arg}_{\mathbf{H}} [\text{Max } P(H_1, H_2, \ldots H_m \mid E_1, E_2, \ldots E_N)]$$

$$\text{Arg}_{\mathbf{H}} [\text{Max } P(H_1, H_2, \ldots H_m \mid \mathbf{E})]$$

$$\mathbf{E} = \{E_1, E_2, \ldots E_N\}$$

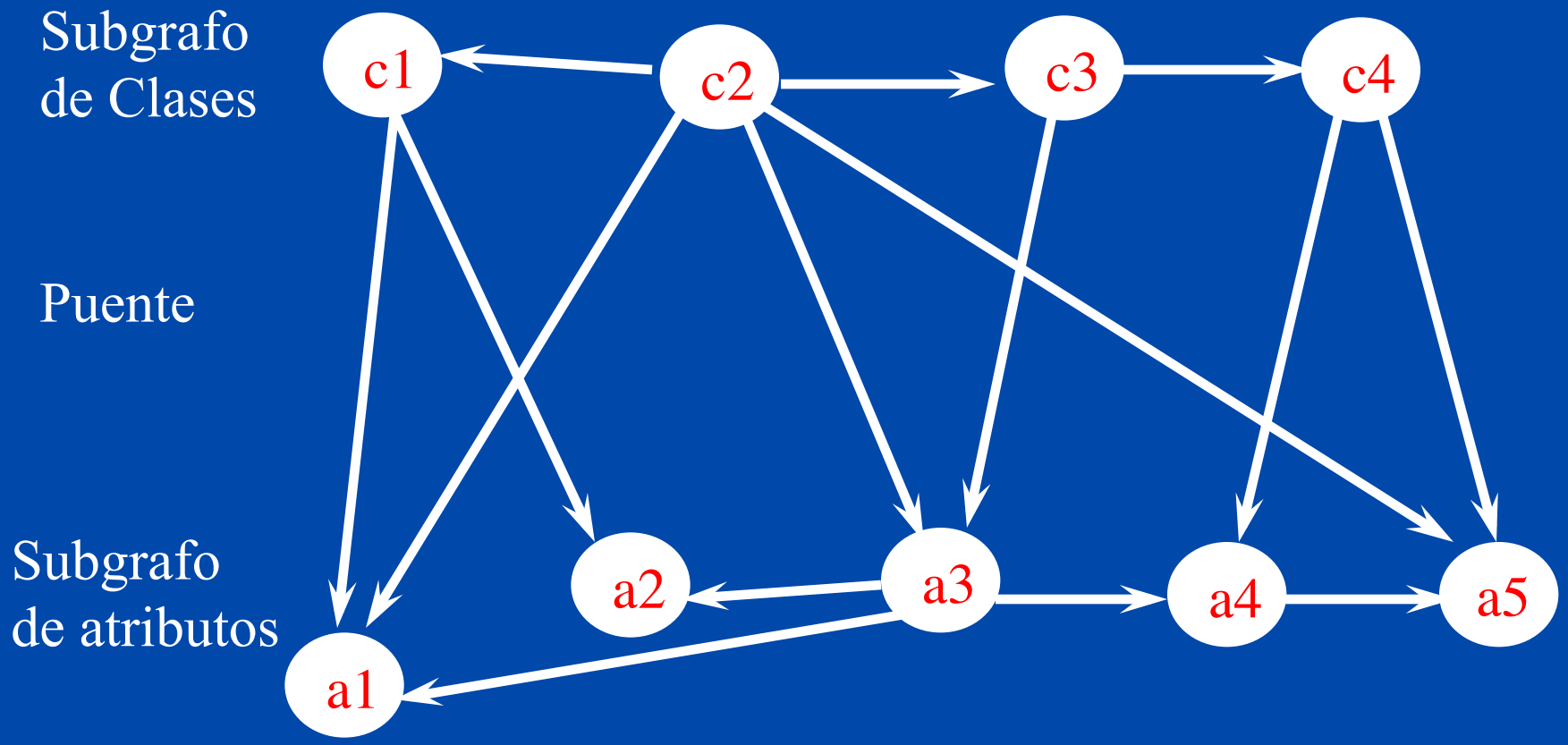
Enfoques Básicos

- Existen dos enfoques básicos para resolver este problema:
  - Binario: se separa en “m” clasificadores independientes y se obtiene un clasificador independiente para cada clase, y luego se concatena el resultado (no considera la dependencia entre clases)
  - Conjunto potencia: se forma una variable clase global que es la combinación de todas las clases individuales (complejidad computacional, aumenta exponencialmente con el número de clases base)

Clasificadores Bayesianos Multidimensionales

- Para considerar las dependencias entre clases y atributos, se construye un modelo gráfico de 3 capas:
  - Dependencias entre Clases
  - Dependencias entre Atributos
  - Dependencias entre Clases y Atributos
- Aprender y clasificar con este modelo es computacionalmente complejo (explicación más probable o *MAP*)

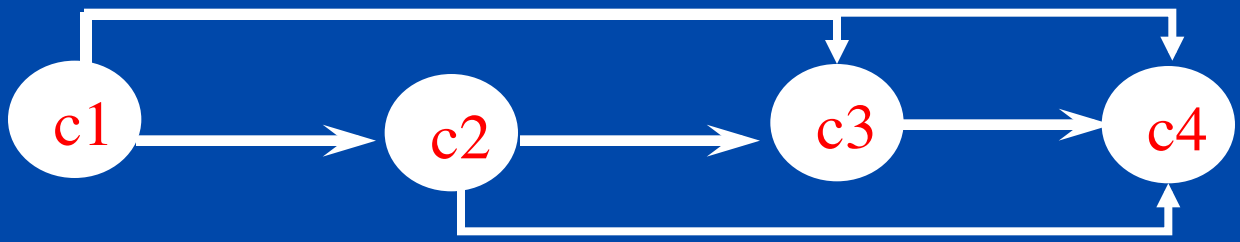
Clasificadores Bayesianos Multidimensionales



Clasificadores en Cadena

- Otra alternativa más sencilla es partir de clasificadores binarios y agregar otras clases como atributos para de cierta forma tomar en cuenta las dependencias entre clases
- Se forma una "cadena" entre las clases de forma que las clases previas en la cadena se incluyen como atributos de las siguientes clases

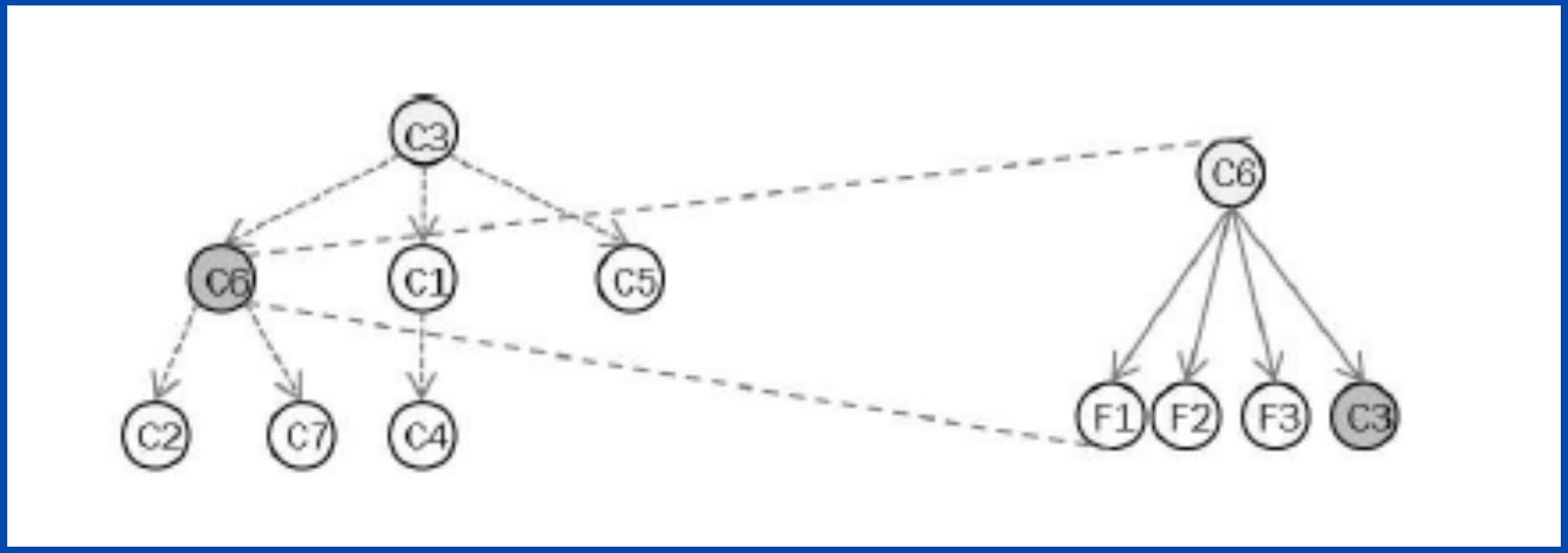
Clasificadores en Cadena



Clasificadores en Cadena Bayesianos

- Se aprende un modelo de dependencias entre clases (red bayesiana)
- El modelo de dependencias puede ser de diversos tipos: árbol, poliárbol, red multiconectada
- Dicho modelo se utiliza para definir el orden de la cadena y los atributos adicionales que se incluyen en cada clase

Clasificadores en Cadena Bayesianos



3: Redes Bayesianas

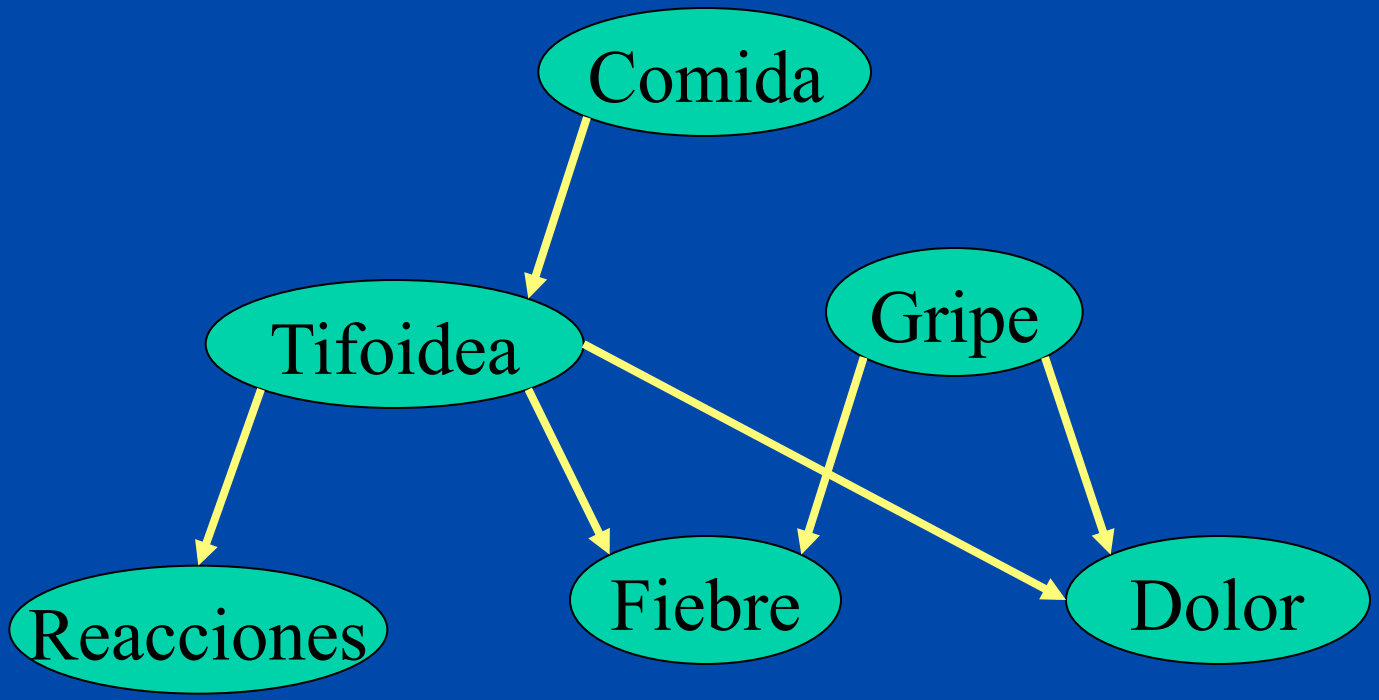
"La probabilidad no es realmente sobre números, es sobre la estructura del razonamiento"

[G. Shafer]

Representación

- Las redes bayesianas son una representación gráfica de dependencias para razonamiento probabilístico, en la cual los nodos y arcos representan:
  - Nodos: Variables proposicionales.
  - Arcos: Dependencia probabilística
- La variable a la que apunta el arco es dependiente (causa-efecto) de la que está en el origen de éste.

Ejemplo de una red bayesiana



Estructura

- La topología o estructura de la red nos da información sobre las dependencias probabilísticas entre las variables.

- La red también representa las independencias condicionales de una variable (o conjunto de variables) dada otra variable(s).

Independencias condicionales

- En una RB, las relaciones de independencia condicional representadas en el grafo corresponden a relaciones de independencia en la distribución de probabilidad.

- Dichas independencias simplifican la representación del conocimiento (menos parámetros) y el razonamiento (propagación de las probabilidades).

Separación “D”

- El conjunto de variables A es independiente del conjunto B dado el conjunto C, si no existe trayectoria entre A y B en que
  1. Todos los nodos convergentes están o tienen descendientes en C
  2. Todos los demás nodos están fuera de C

Separación “D”

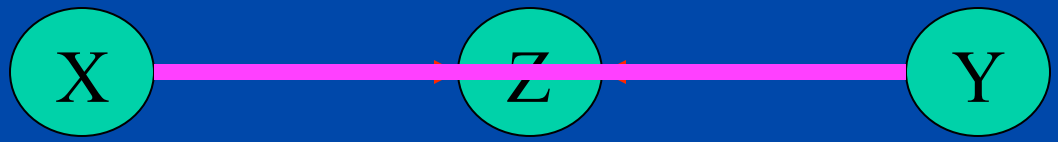
- Tres casos básicos
  - Arcos divergentes
  - Arcos en secuencia
  - Arcos convergentes

Separación “D” – casos básicos

- caso 1: Secuencia:

- caso 2: Divergentes:

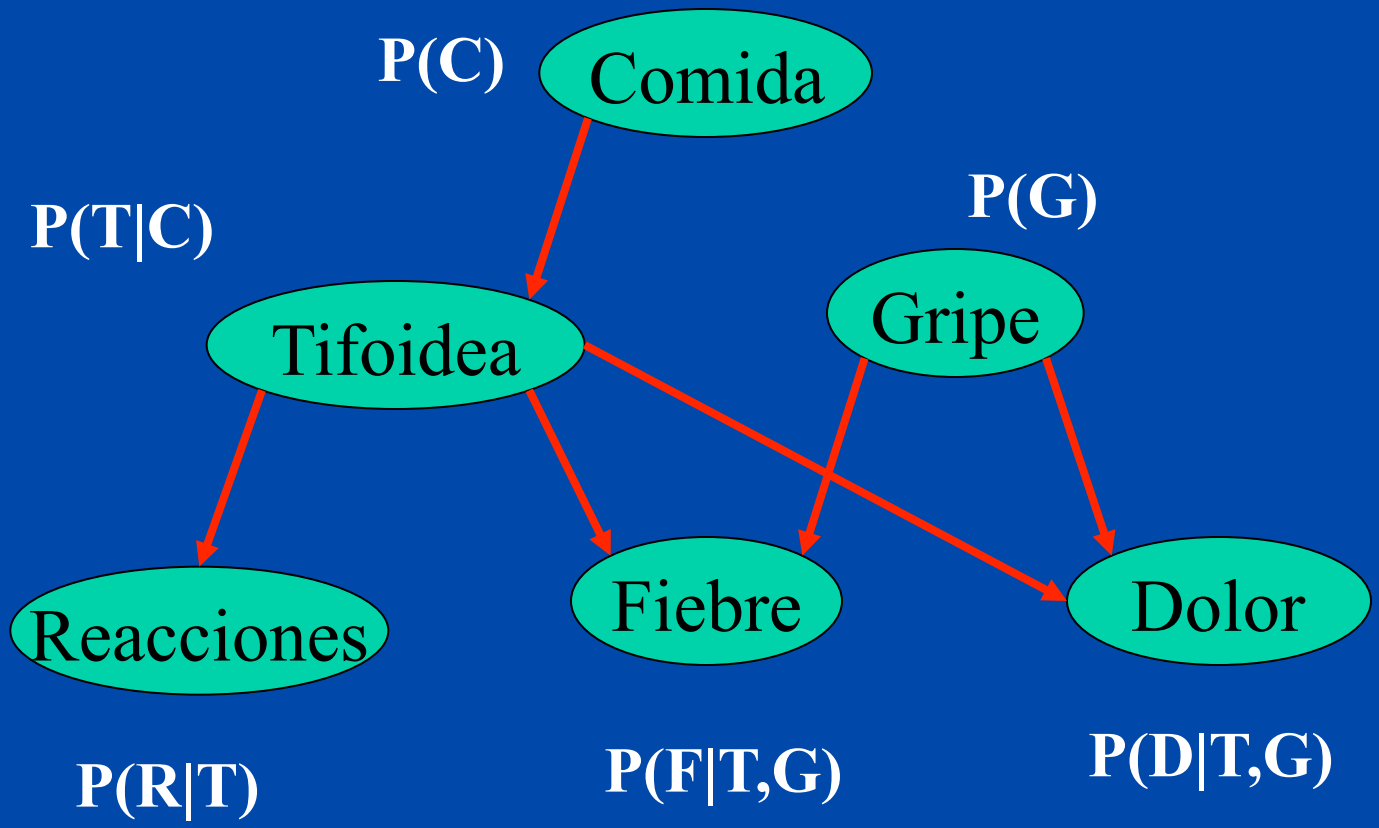
- caso 3: Convergentes:



Parámetros

- Complementan la definición de una red bayesiana las probabilidades condicionales de cada variable dados sus padres.
  - **Nodos raíz**: vector de probabilidades marginales
  - **Otros nodos**: matriz de probabilidades condicionales dados sus padres

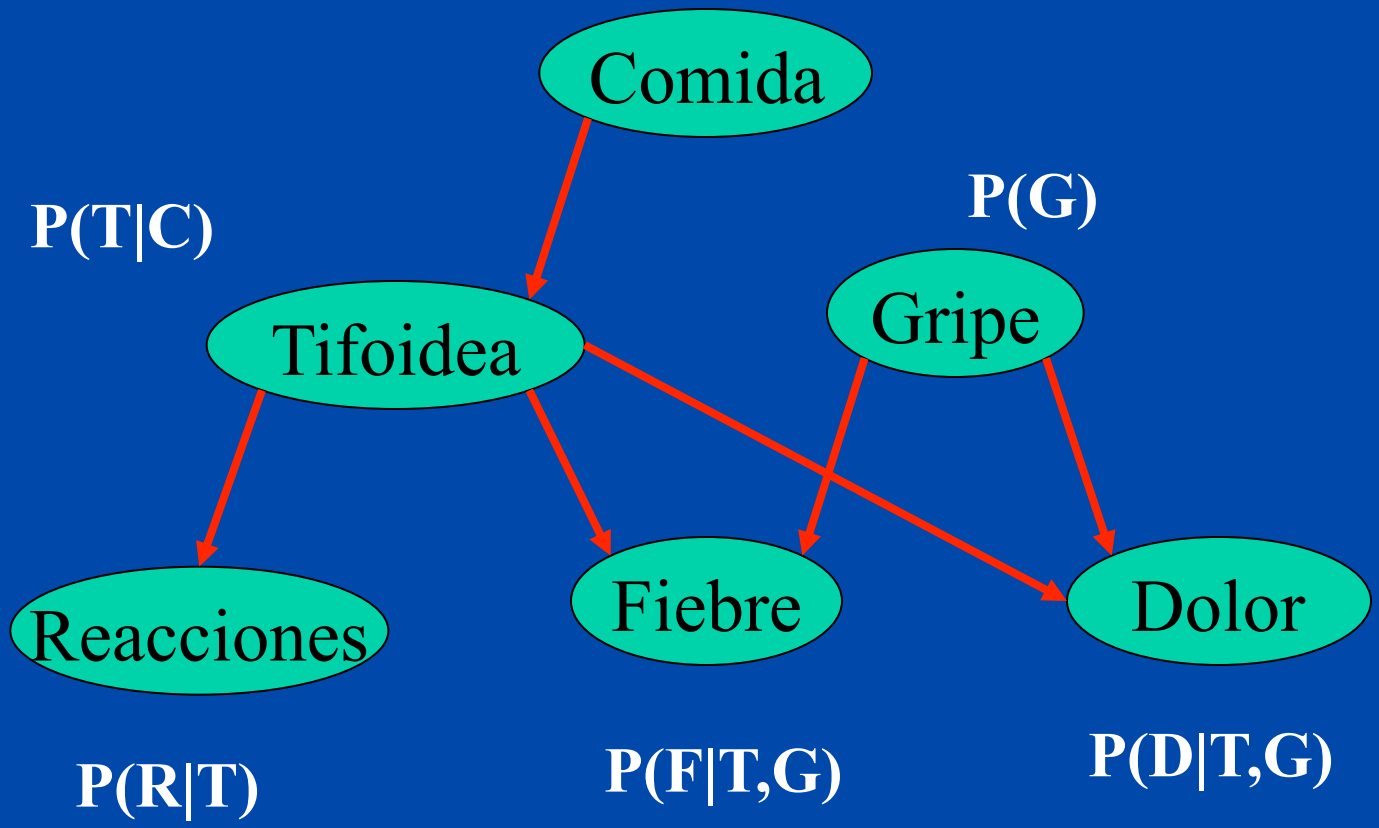
Ejemplo

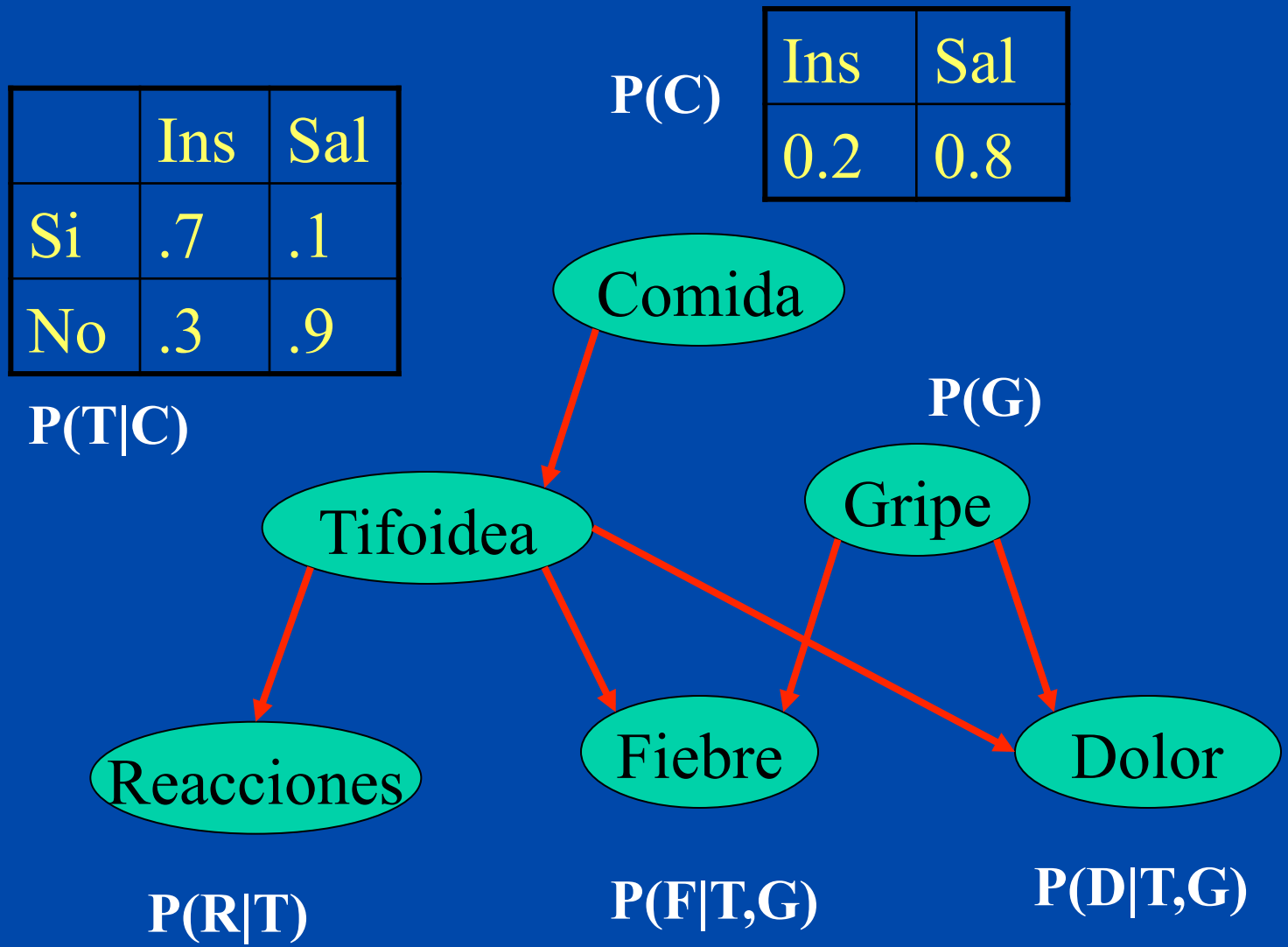

P(C)
Comida
P(T|C)
P(G)
Tifoidea
Gripe
Reacciones
Fiebre
Dolor
P(R|T)
P(F|T,G)
P(D|T,G)

Ejemplo

P(C)

Ins	Sal
0.2	0.8



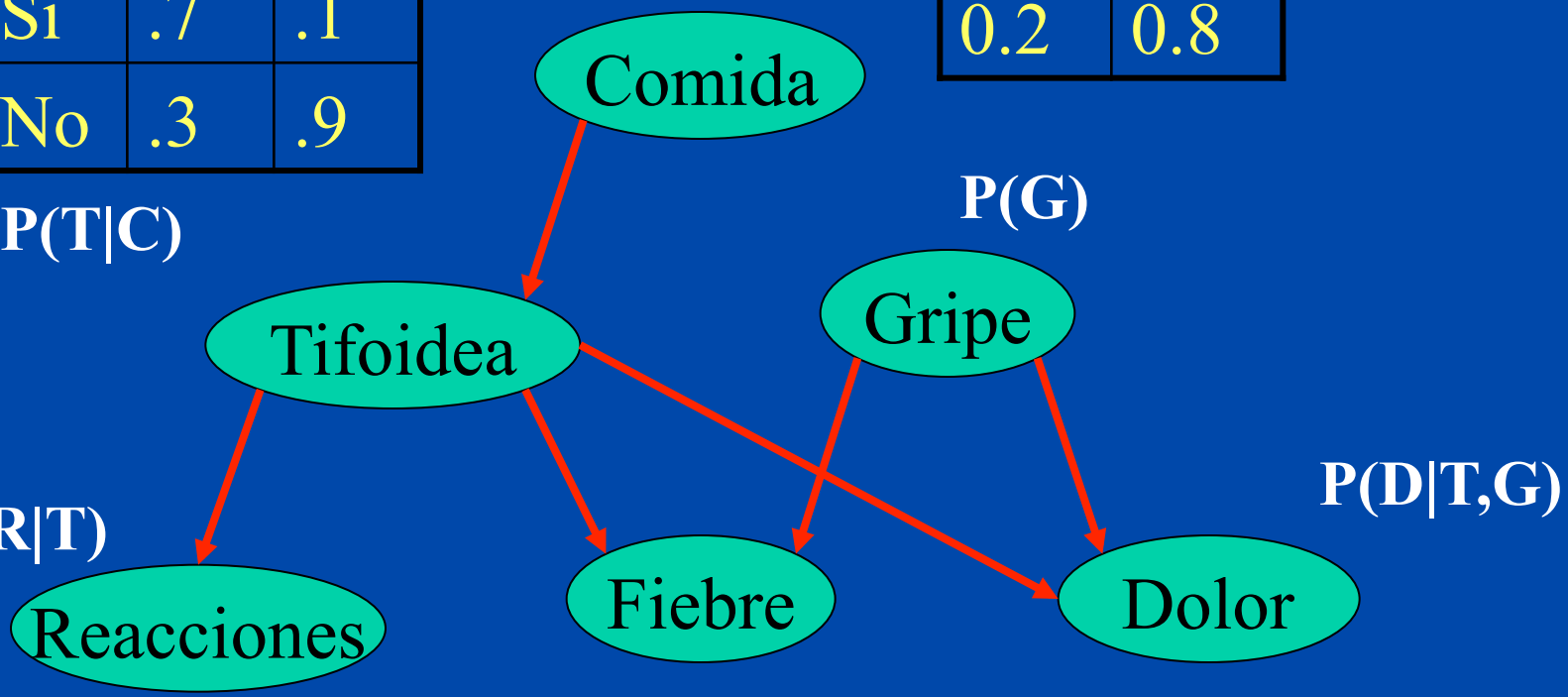

P(C)
Ins
Sal
0.2
0.8
Comida
P(G)
Gripe
Tifoidea
Reacciones
Fiebre
Dolor
Si
.7
.1
No
.3
.9
P(T|C)
P(R|T)
P(F|T,G)
P(D|T,G)

	Ins	Sal
Si	.7	.1
No	.3	.9

P(T|C)

P(C)

Ins	Sal
0.2	0.8



P(G)

P(R|T)

P(D|T,G)

P(F|T,G)

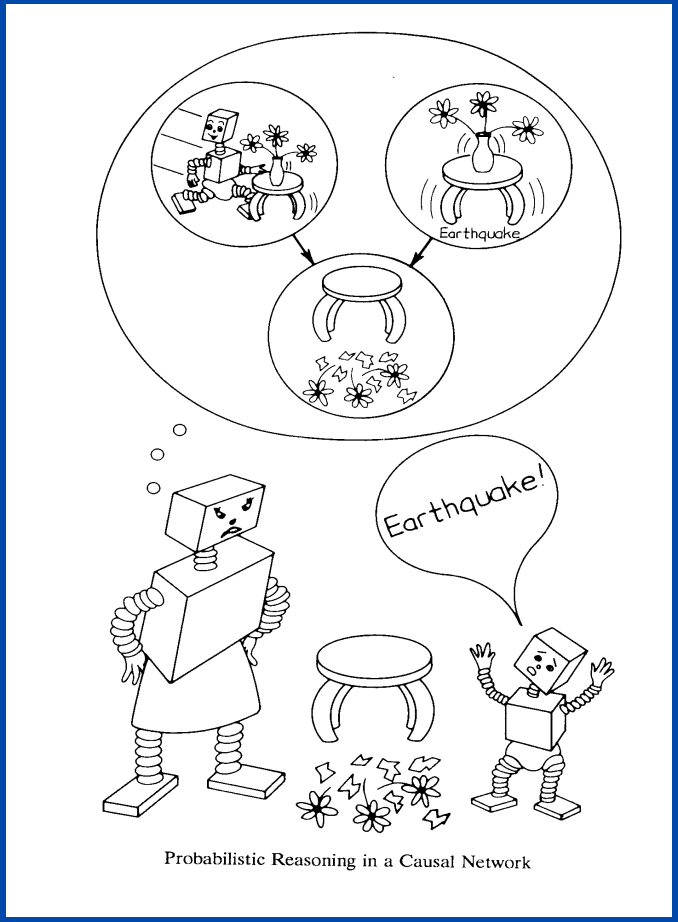
	Si, Si	Si,No	No,Si	No,No
F	0.8	0.6	0.5	0.1
~F	0.2	0.4	0.5	0.9

Especificación Paramétrica

- En general, la probabilidad conjunta se especifica por el producto de las probabilidades de cada variable dados sus padres:

$$P(X1,X2, ..., Xn) = \prod P(Xi \mid Pa(Xi))$$

4: Redes Bayesianas – Inferencia



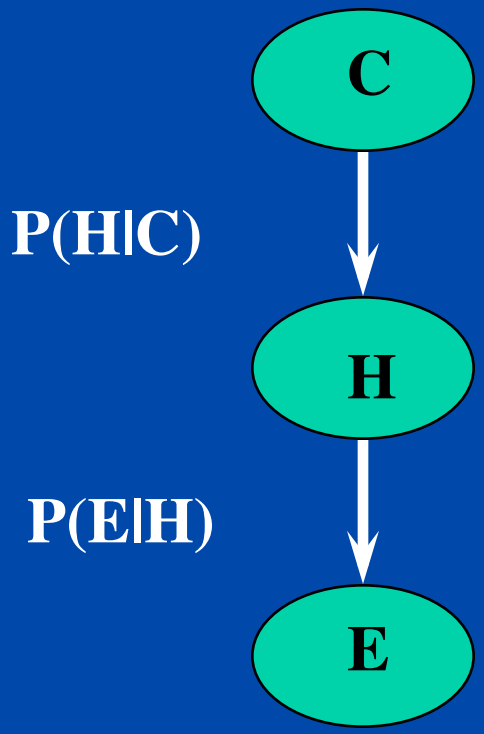
[Neapolitan 90]

Inferencia probabilística

- En RB, la inferencia probabilística consiste en:

  "dadas ciertas variables conocidas (evidencia), calcular la probabilidad posterior de las demás variables (desconocidas)"
- Es decir, calcular: P(Xi | **E**), donde:
  - **E** es un subconjunto de variables de la RB (posiblemente vació)
  - Xi es cualquier variable en la RB, no en **E**

Inferencia bayesiana



***Causal*:**
C → H

***Evidencial*:**
E → H

***Mixta*:**
C, E → H

Tipos de Técnicas

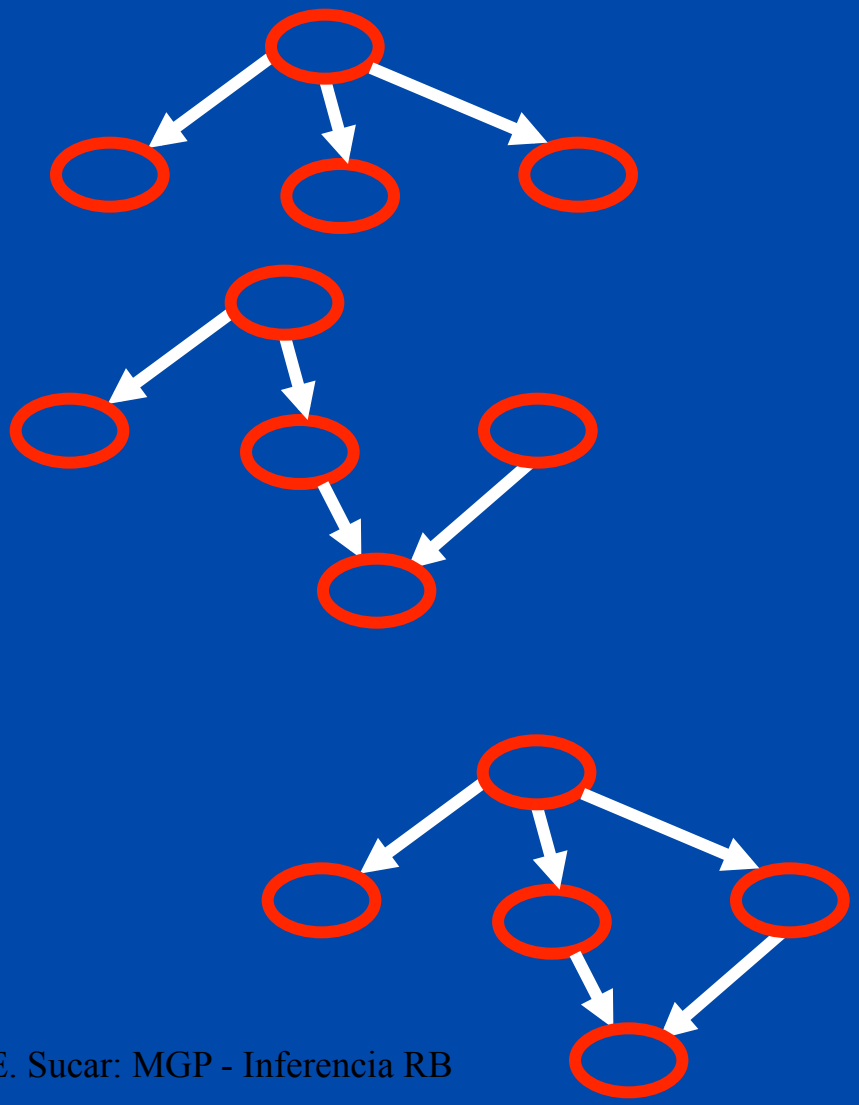
- Calcular probabilidades posteriores:
  - Una variable, cualquier estructura: algoritmo de eliminación (*variable elimination*)
  - Todas las variable, estructuras sencillamente conectadas (árboles, poliárboles): propagación
  - Todas las variables, cualquier estructura:
    - Agrupamiento (*junction tree*)
    - Condicionamiento
    - Técnicas aproximadas (*Loopy BP*, simulación estocástica)

Tipos de Técnicas

- Obtener variable(s) de mayor probabilidad dada cierta evidencia – abducción:
  - Abducción total
  - Abducción parcial

Tipos de estructuras

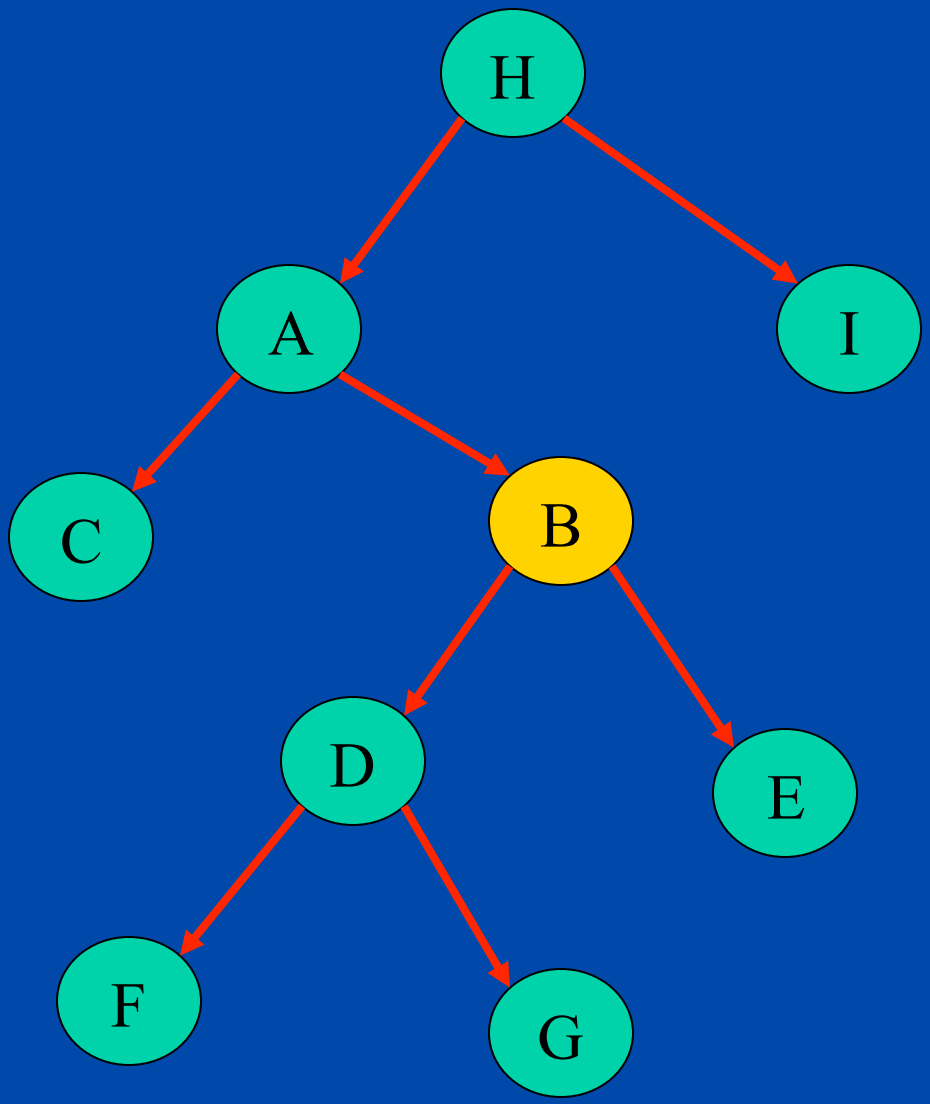
- Sencillamente conectadas
  - Árboles
  - Poliárboles
- Multiconectadas



Propagación en Árboles

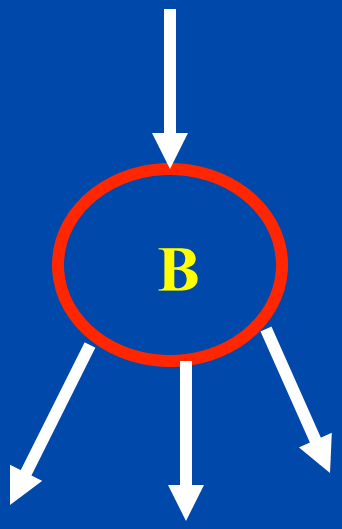
Cada nodo corresponde a una variable discreta, B ($B_1, B_2, \ldots, B_m$) con su respectiva matriz de probabilidad condicional, $P(B|A)=P(B_j|A_i)$

Propagación en Árboles

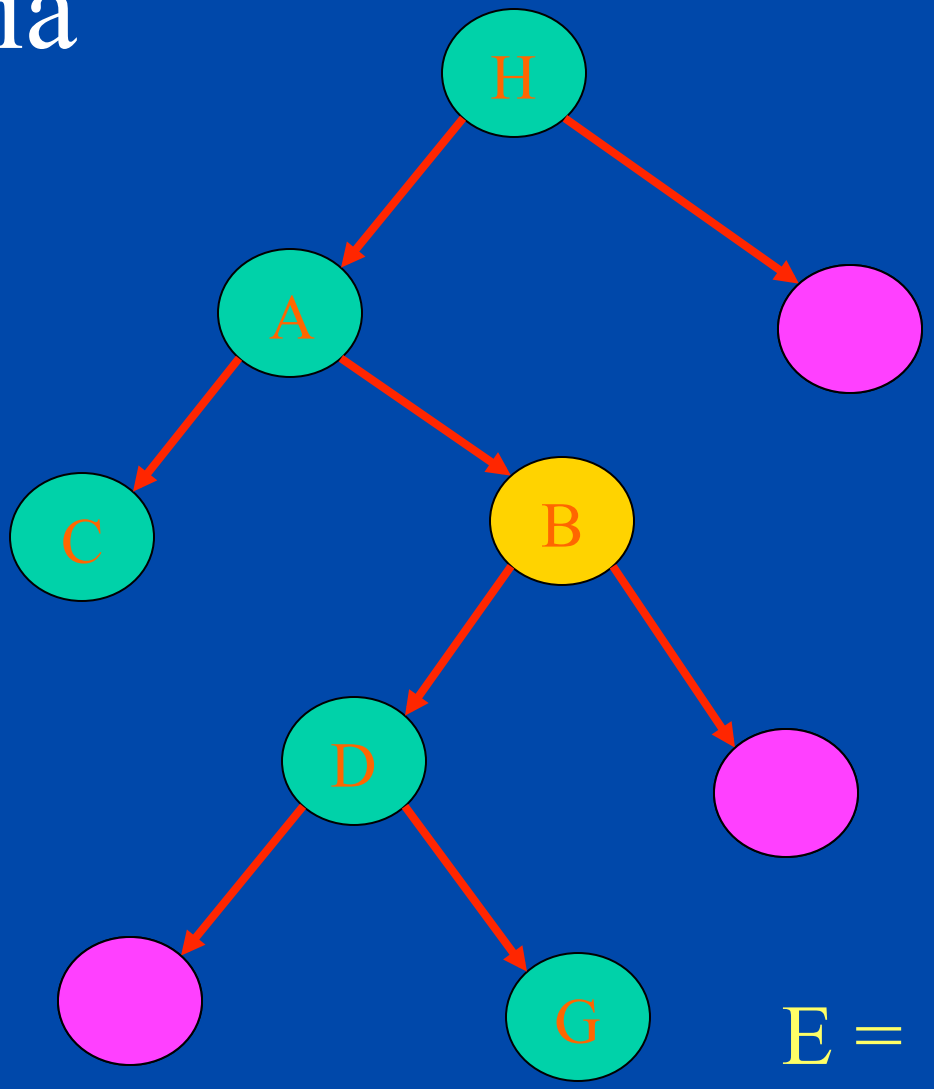


Dada cierta evidencia *E* -representada por la instanciación de ciertas variables- la probabilidad posterior de cualquier variable *B*, por el teorema de Bayes:

$$P(B_i \mid E) = P(B_i)\, P(E \mid B_i) / P(E)$$



Evidencia



E = {I,F,E}

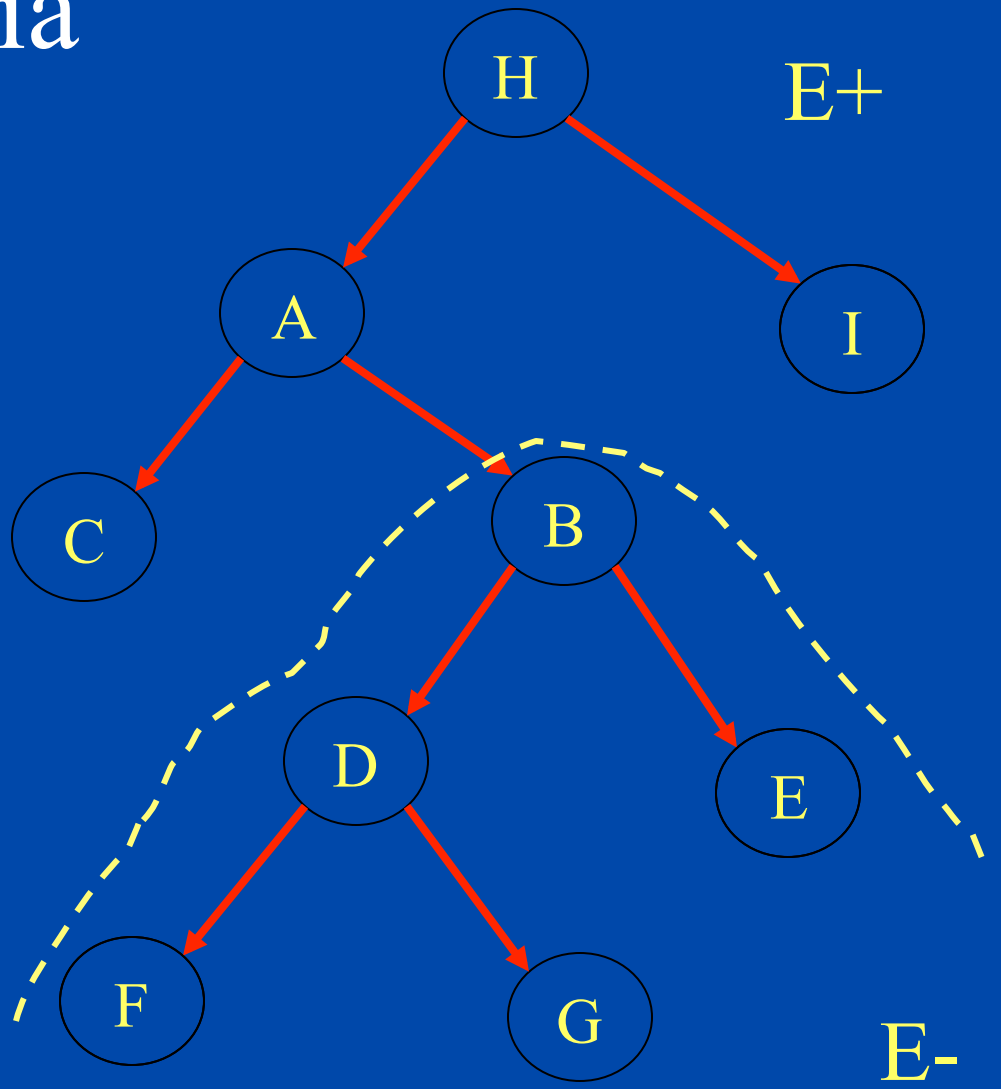
Evidencia

Ya que la estructura de la red es un árbol, el Nodo *B* la separa en dos subárboles, por lo que podemos dividir la evidencia en dos grupos:

E-: Datos en el árbol que cuya raíz es *B*

E+: Datos en el resto del árbol

Evidencia



Entonces:

$$P(B_i \mid E) = P(B_i)\, P(E^-, E^+ \mid B_i) / P(E)$$

Pero dado que ambos son independientes y aplicando nuevamente Bayes:

$$P(B_i \mid E) = \alpha\, P(B_i \mid E^+)\, P(E^- \mid B_i)$$

Donde α es una constante de normalización

Definiciones:

Si definimos los siguientes términos:

$$\lambda(B_i) = P(E^- \mid B_i)$$

$$\pi(B_i) = P(B_i \mid E^+)$$

Entonces:

$$P(B_i \mid E) = \alpha\, \pi(B_i)\, \lambda(B_i)$$

Algoritmo

- Mediante estas ecuaciones se integra un algoritmo de propagación de probabilidades en árboles.
- Cada nodo guarda los valores de los vectores π y λ, así como su matriz de probabilidad condicional (CPT), P.
- La propagación se hace por un mecanismo de paso de mensajes, en donde cada nodo envía los mensajes correspondientes a su padre e hijos

Mensaje al padre (hacia arriba) – nodo B a su padre A:

$$\lambda_B(A_i) = \sum_j P(B_j|A_i)\lambda(B_j)$$

Mensaje a los hijos (hacia abajo) - nodo B a su hijo S_k :

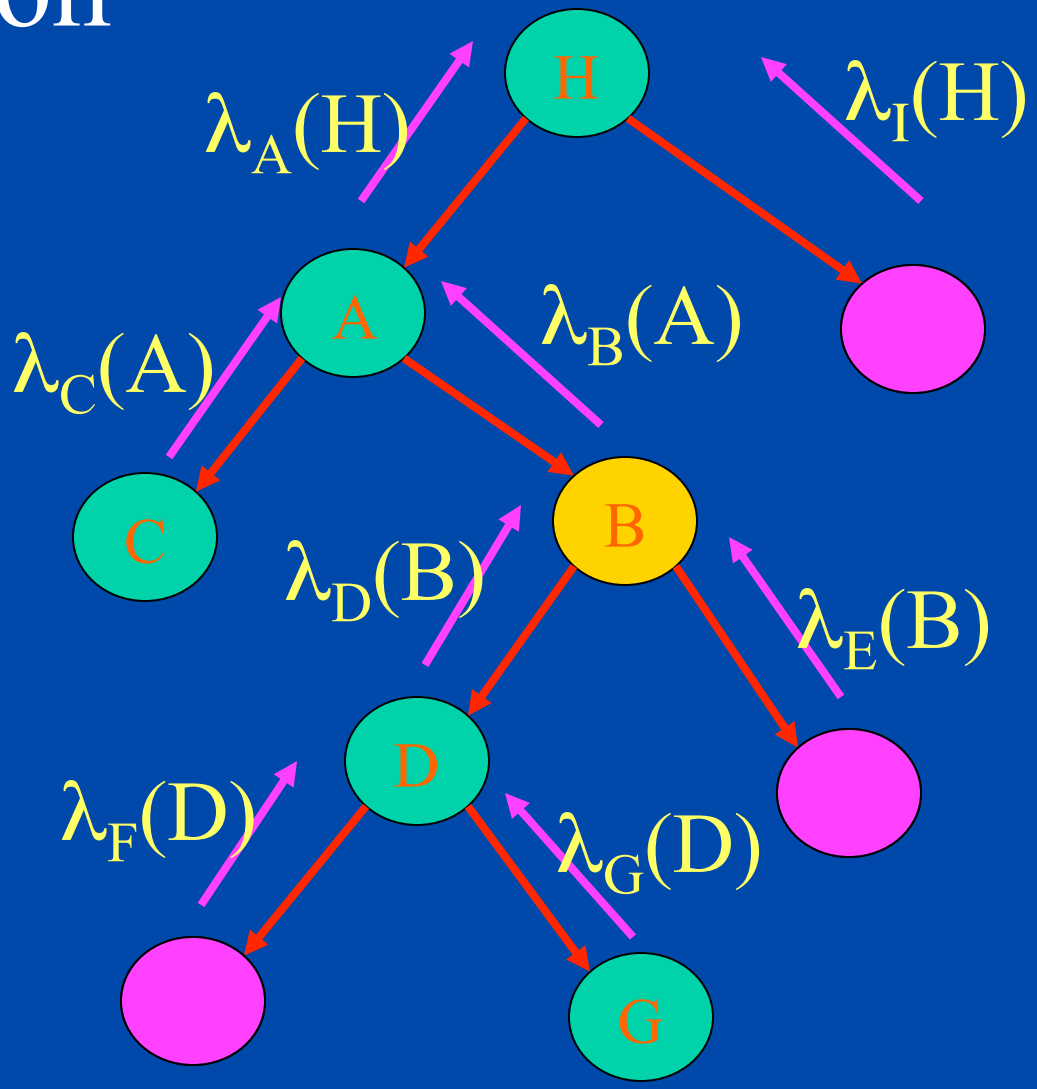
$$\pi_k(B_i) = \alpha\pi(B_j)\prod_{I\neq k}\lambda_I(B_j)$$

Algoritmo

- Al instanciarse ciertos nodos, éstos envían mensajes a sus padres e hijos, y se propagan hasta a llegar a la raíz u hojas, o hasta encontrar un nodo instanciado.

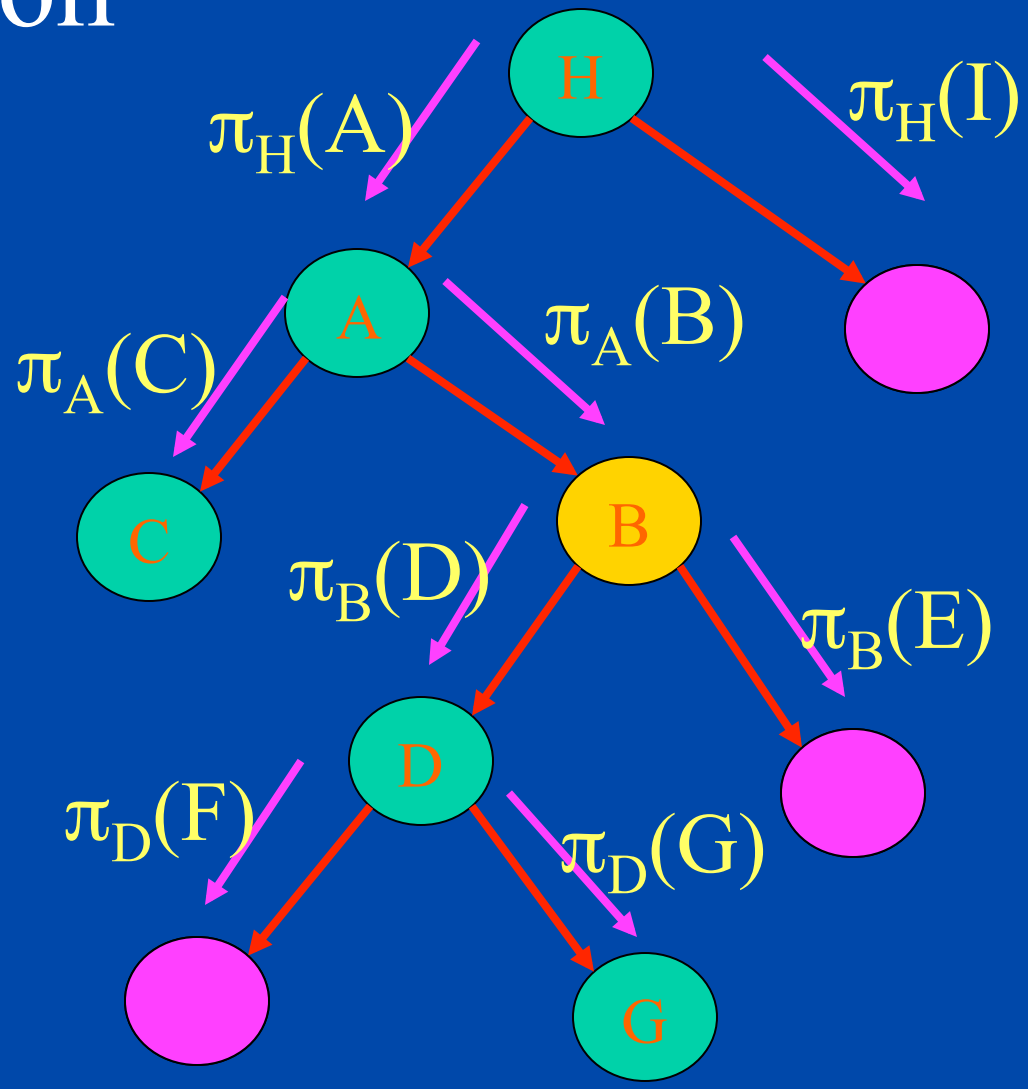
- Así que la propagación se hace en un solo paso, en un tiempo proporcional al diámetro de la red.

Propagación λ



Propagación

π



Condiciones Iniciales

- Nodos hoja no conocidos:

  $\lambda (B_i) = [1,1, \ldots]$

- Nodos asignados (conocidos):

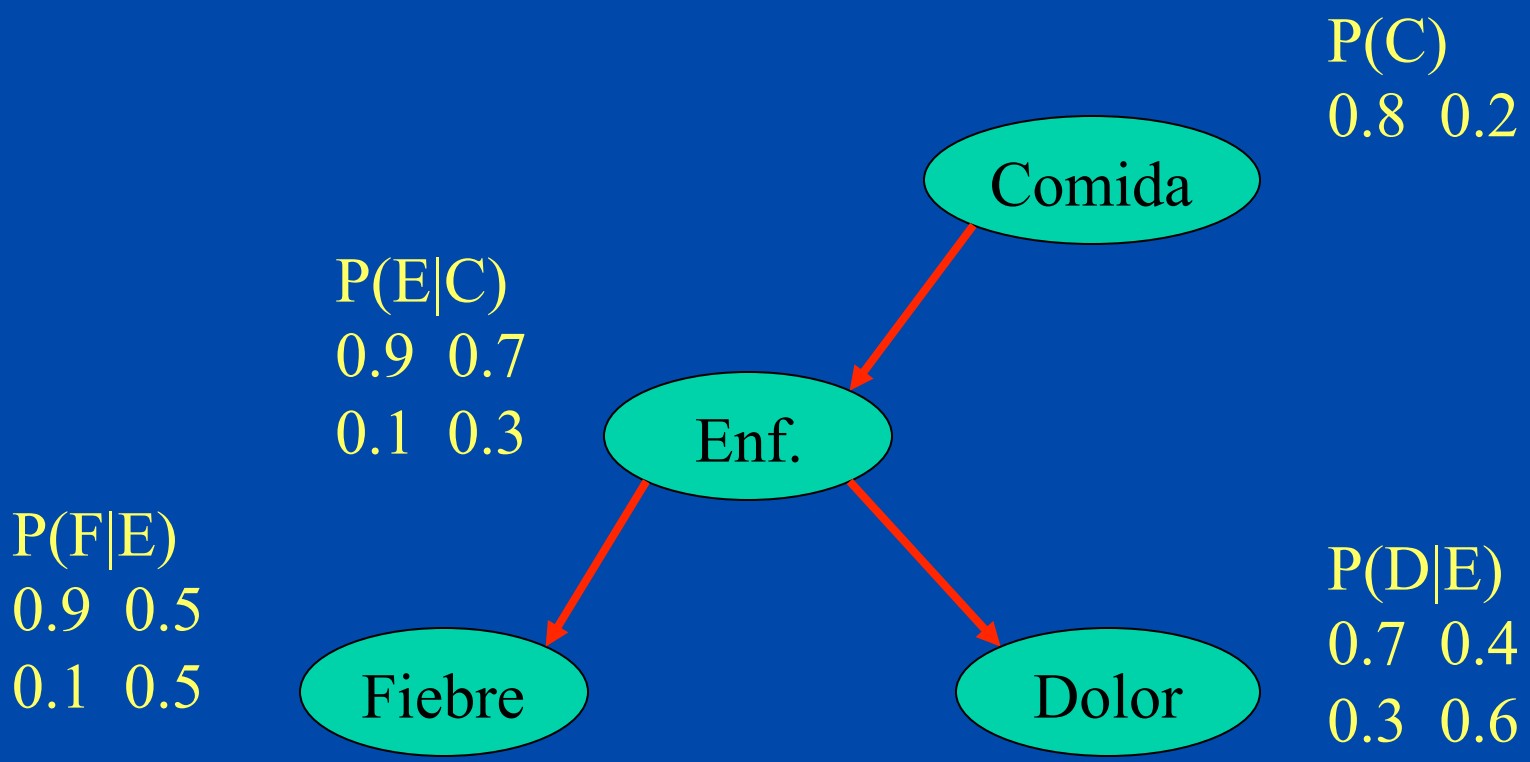
  $\lambda (B_i) = [0,0, ..1, 0, \ldots, 0]$ (1 para valor asignado)

  $\pi (B_i) = [0,0, ..1, 0, \ldots, 0]$ (1 para valor asignado)

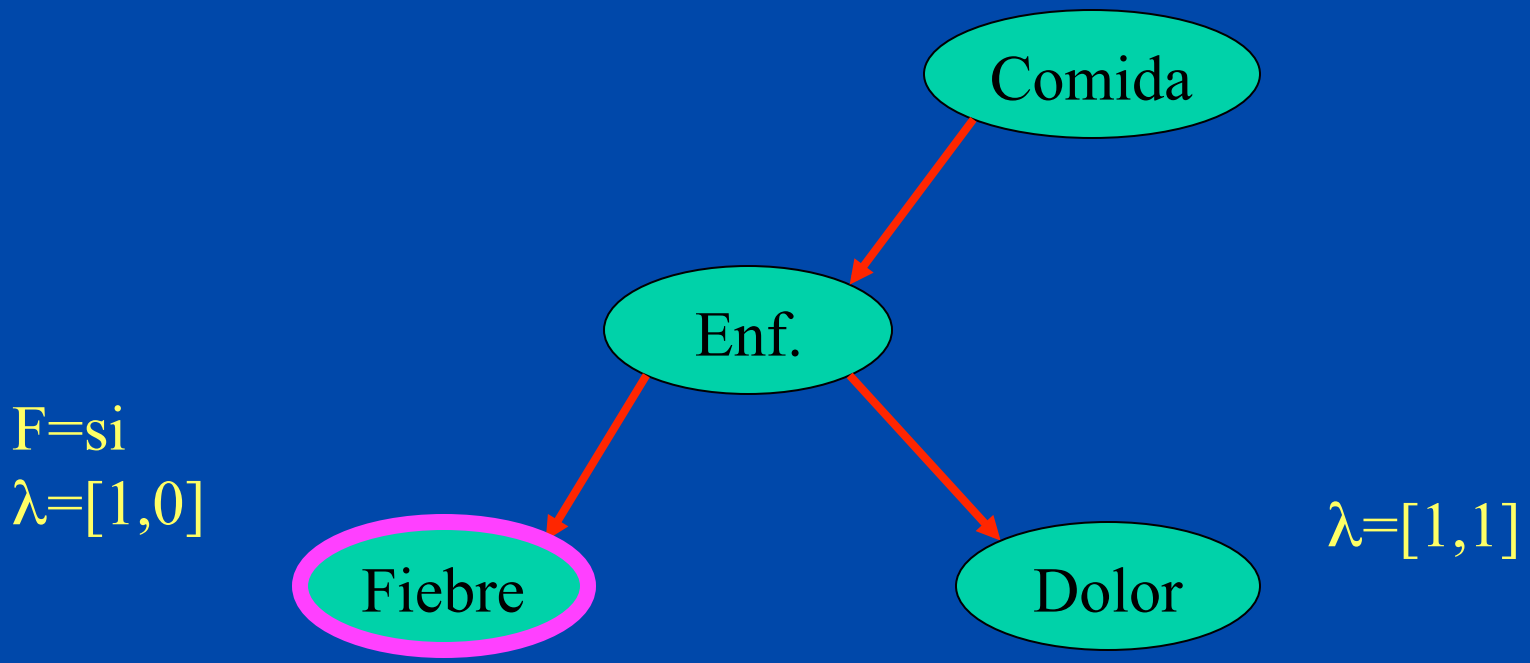
- Nodo raíz no conocido:

  $\pi (A) = P(A)$, (probabilidad marginal inicial)

Ejemplo

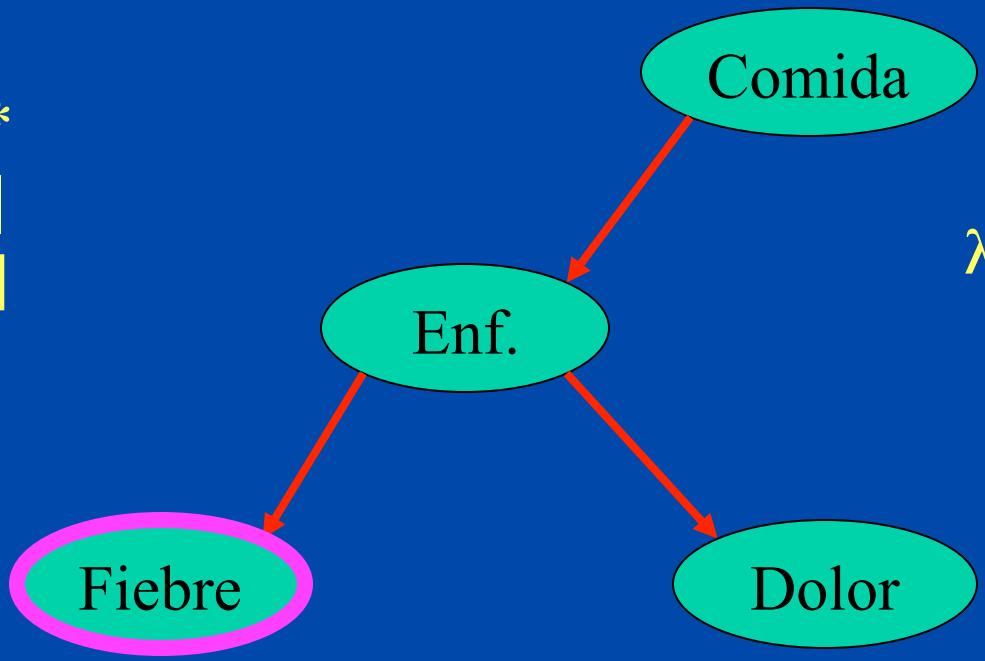

P(C)
0.8 0.2
Comida
P(E|C)
0.9 0.7
0.1 0.3
Enf.
P(F|E)
0.9 0.5
0.1 0.5
Fiebre
Dolor
P(D|E)
0.7 0.4
0.3 0.6

Ejemplo



Ejemplo

λ_F= [1,0] *
[.9 .5 |
.1 .5]
= [.9 .5]



λ_D= [1,1] *
[.7 .4 |
.3 .6]
= [1 1]

P(F|E)
0.9 0.5
0.1 0.5

P(D|E)
0.7 0.4
0.3 0.6

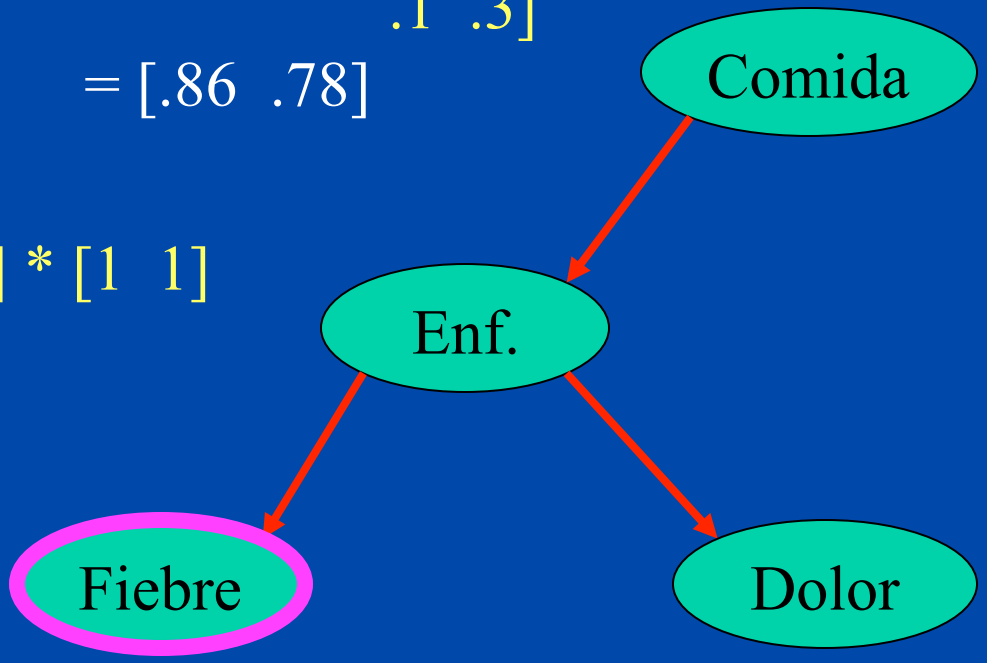
Ejemplo

$\lambda(C) = [.9\ .5] * [.9\ \ .7 \mid .1\ \ .3]$

$= [.86\ \ .78]$

$\lambda(E) = [.9\ .5] * [1\ \ 1]$

$= [.9\ .5]$



P(E|C)

0.9	0.7
0.1	0.3

P(F|E)

0.9	0.5
0.1	0.5

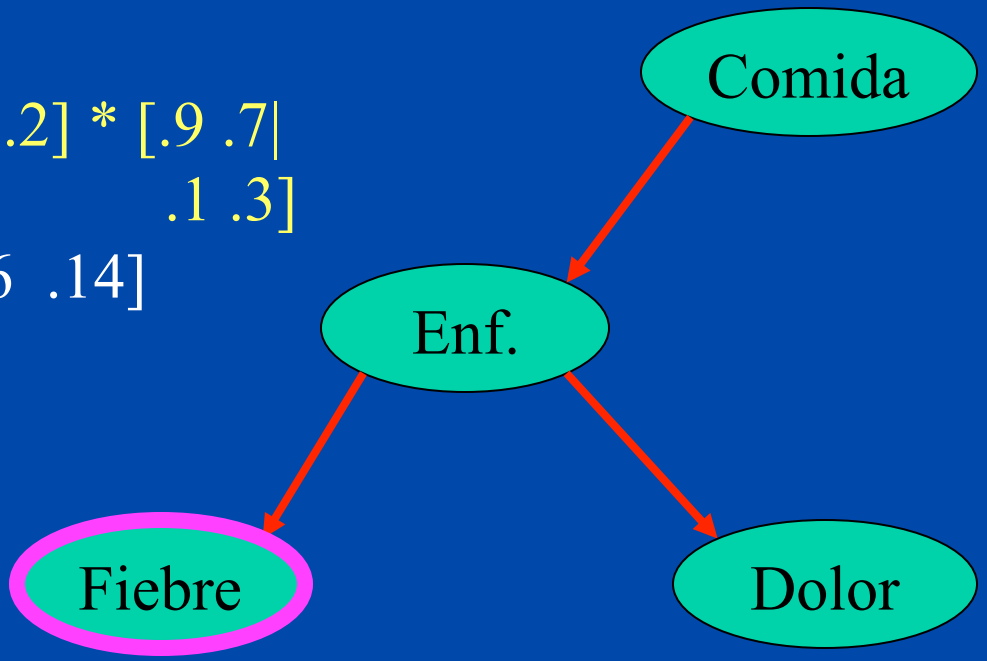
P(D|E)

0.7	0.4
0.3	0.6

© L.E. Sucar: MGP - Inferencia RB

Ejemplo

$\pi(C) = [.8 \quad .2]$



$\pi(E) = [.8 \ .2] * [.9 \ .7 | .1 \ .3]$

$= [.86 \quad .14]$

P(E|C)

0.9	0.7
0.1	0.3

P(F|E)

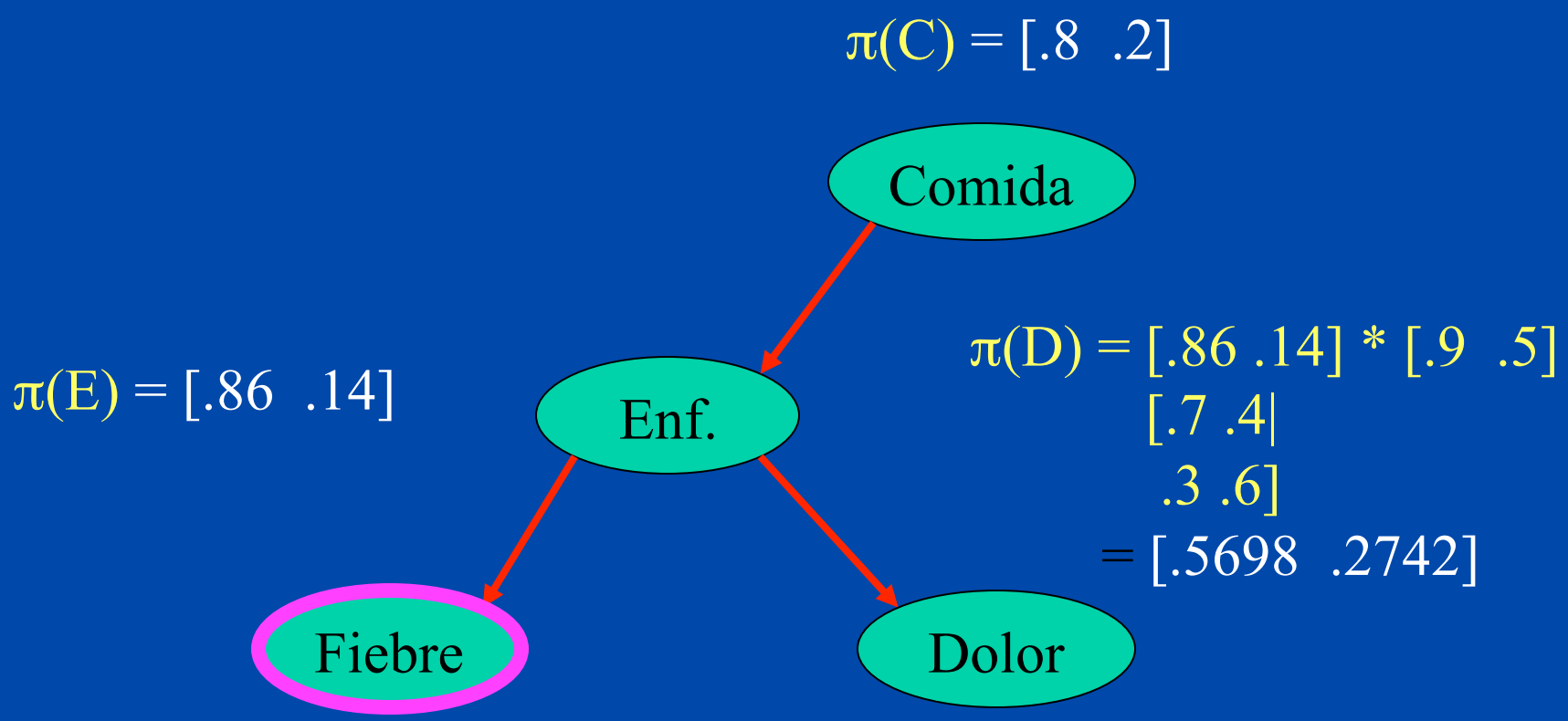
0.9	0.5
0.1	0.5

P(D|E)

0.7	0.4
0.3	0.6

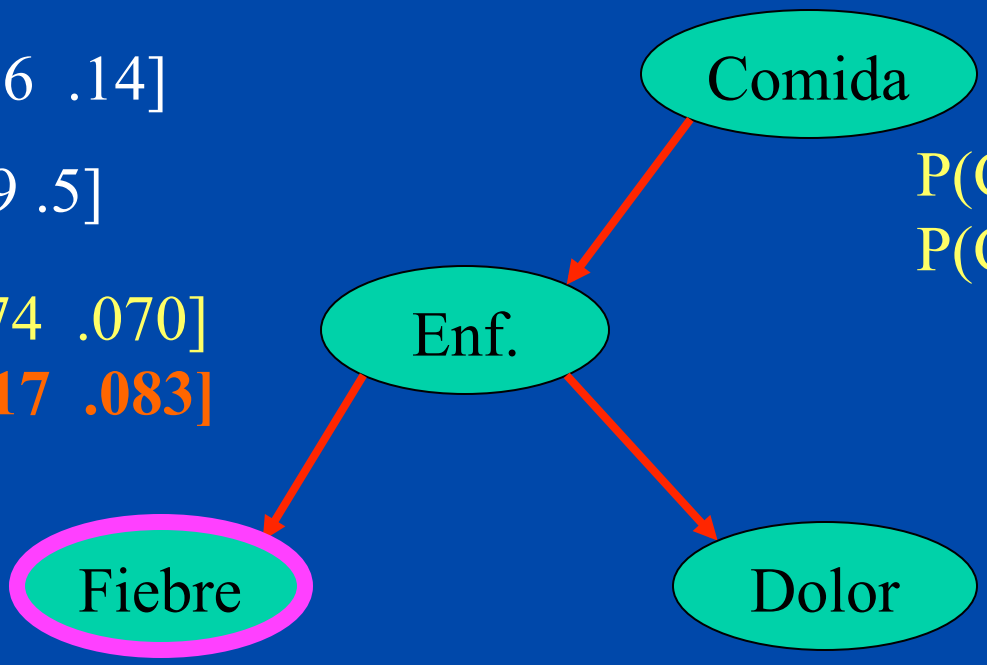
© L.E. Sucar: MGP - Inferencia RB

Ejemplo



P(D|E)
0.7 0.4
0.3 0.6

Ejemplo



$\pi(C) = [.8 \quad .2]$

$\lambda C) = [.86 \quad .78]$

$P(C)=\alpha[.688 \quad .156]$

$P(C)= \quad \mathbf{[.815 \quad .185]}$

$\pi(E) = [.86 \quad .14]$

$\lambda(E) = [.9 \quad .5]$

$P(E)=\alpha[.774 \quad .070]$

$P(E)= \quad \mathbf{[.917 \quad .083]}$

$\pi(D) = [.57 \quad .27]$

$\lambda(D)=[1,1]$

$P(D)=\alpha[.57 \quad .27]$

$P(D)= \quad \mathbf{[.67 \quad .33]}$

Propagación en redes multiconectadas

- Una red multiconectada es un grafo no conectado en forma sencilla, es decir, en el que hay múltiples trayectorias entre nodos.
- Para este tipo de redes existen varios tipos de técnicas de inferencia:
  - **Propagación "Loopy"**
  - **Condicionamiento**
  - **Simulación estocástica**
  - **Agrupamiento**

Simulación estocástica

- Se asignan valores aleatorios a las variables no asignadas, se calcula la distribución de probabilidad, y se obtienen valores de cada variable dando una muestra.
- Se repite el procedimiento para obtener un número apreciable de muestras y en base al número de ocurrencias de cada valor se determina la probabilidad de dicha variable.

Muestreo Lógico

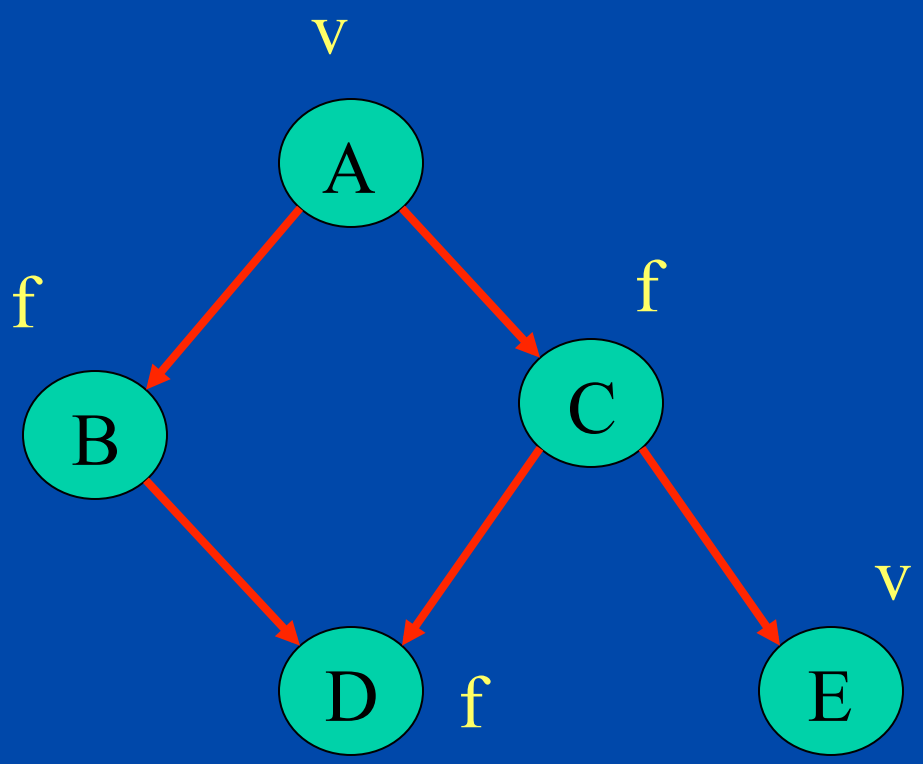
Para "N" muestras, repetir:

1. Dar valores aleatorios a los nodos raíz de acuerdo a sus probabilidades
2. En base a los valores anteriores, dar valores aleatorios a las siguientes variables (hijos de los nodos raíz) en función de la probabilidad condicional
3. Repetir (2) hasta llegar a los nodos hoja

Obtener probabilidades posteriores como frecuencias

- Si hay nodos evidencia, sólo considerar las muestras que correspondan a dichos valores

Muestreo Lógico: ejemplo



vfffv

fvvff

vffvf

ffvfv

vfvvf

ffffv

fvvvf

fffff

fffvf

vvvvf

Ejemplo

- Sin evidencia:
  - P(A=V) = 4/10 = 0.4
  - P(B=V) = 3/10 = 0.3
  - P(C=V) = 5/10 = 0.5
  - P(D=V) = 5/10 = 0.5
  - P(E=V) = 3/10 = 0.3
- Con evidencia: D=V (aplican 5 muestras):
  - P(A=V) = 3/5 = 0.6
  - P(B=V) = 2/5 = 0.4
  - P(C=V) = 3/5 = 0.6
  - P(E=V) = 1/5 = 0.2

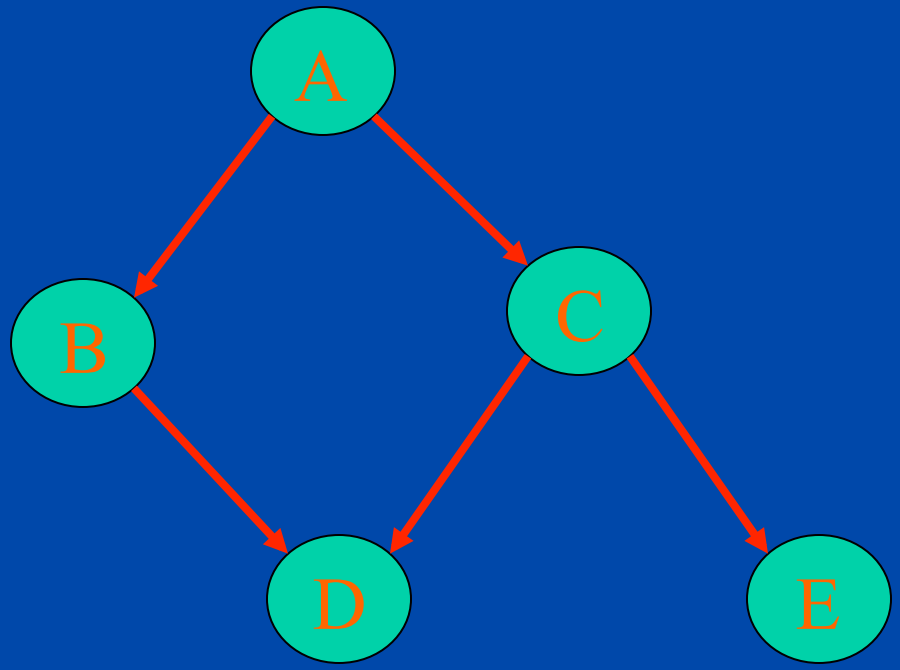
Agrupamiento

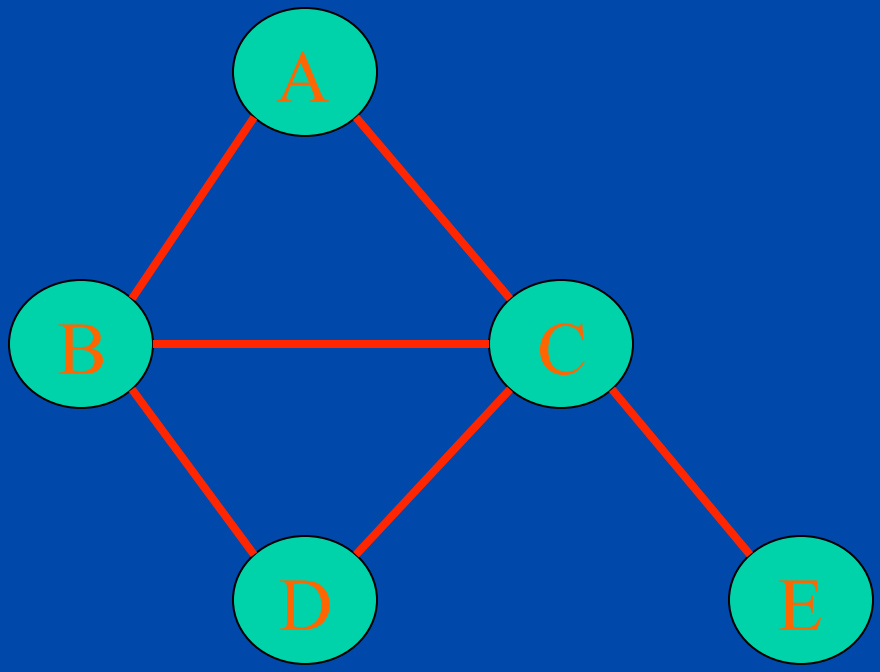
- El método de agrupamiento consiste en transformar la estructura de la red para obtener un árbol, mediante agrupación de nodos usando la teoría de grafos.
- La propagación se realiza sobre el árbol de macro-nodos obtenido, donde cada macro-nodo corresponde a un clique o *unión* de la RB original (*junction tree*)

Agrupamiento

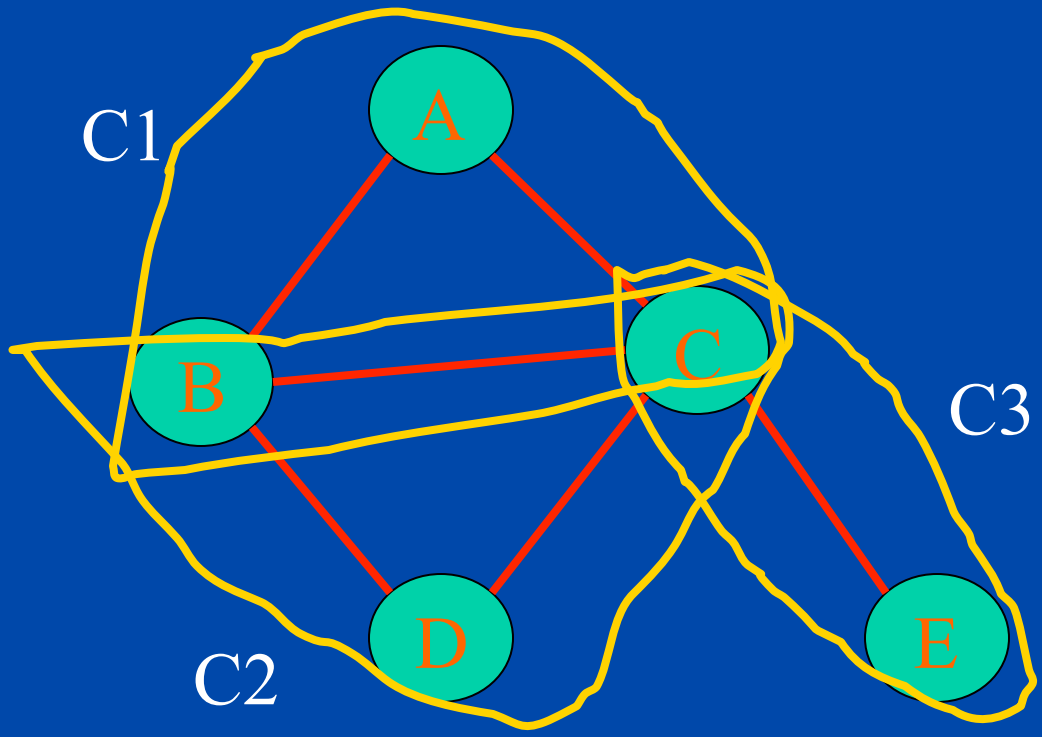
- Transformación:
  - Eliminar direccionalidad de los arcos
  - Ordenamiento de los nodos por máxima cardinalidad
  - Moralizar el grafo (arco entre nodos con hijos comunes)
  - Triangular el grafo
  - Obtener los *cliques* y ordenar
  - Construir árbol de *cliques*

Ejemplo

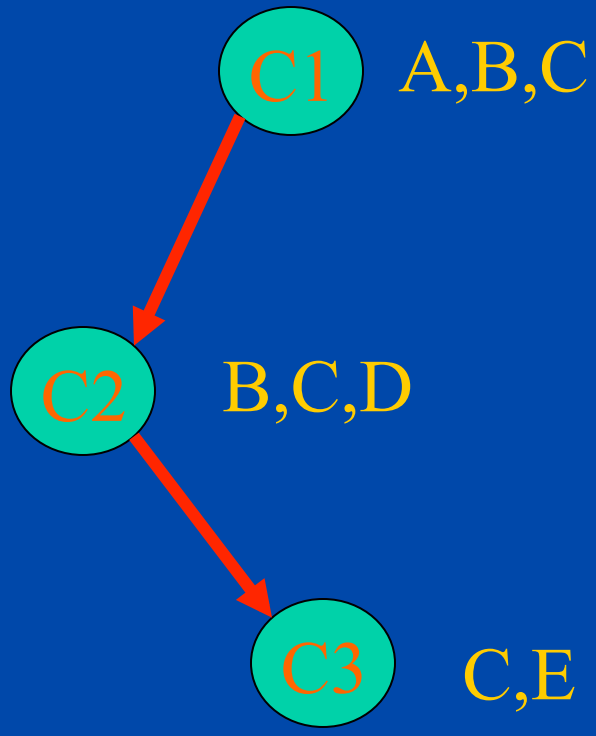




Ordenamiento de Cliques



Árbol de Cliques



Propagación

- La propagación es mediante el envío de mensajes en el árbol de *cliques* (en forma similar a árboles)
- Inicialmente se calcula la probabilidad conjunta (potencial) de cada *clique*, y la condicional dado el padre
- Dada cierta evidencia se recalculan las probabilidades de cada *clique*
- La probabilidad individual de cada variable se obtiene de la del *clique* por marginalización

Complejidad

- En el peor caso, la propagación en redes bayesianas es un problema NP-duro
- En la práctica, en muchas aplicaciones se tienen redes no muy densamente conectadas y la propagación es eficiente aún para redes muy grandes (función del *clique* mayor)
- Para redes muy complejas (muchas conexiones), la mejor alternativa son técnicas de simulación estocástica o técnicas aproximadas

5: Redes Bayesianas – Aprendizaje

"Preferiría descubrir una ley causal que ser rey de Persia" [Democritus]

Aprendizaje

El aprendizaje inductivo consiste en obtener conocimiento a partir de datos.

En redes bayesianas se divide en 2 aspectos:

- Obtener la estructura de la red – aprendizaje estructural
- Obtener las probabilidades asociadas – aprendizaje paramétrico

Aprendizaje Paramétrico

- Datos completos - se estiman las probabilidades a partir de frecuencias

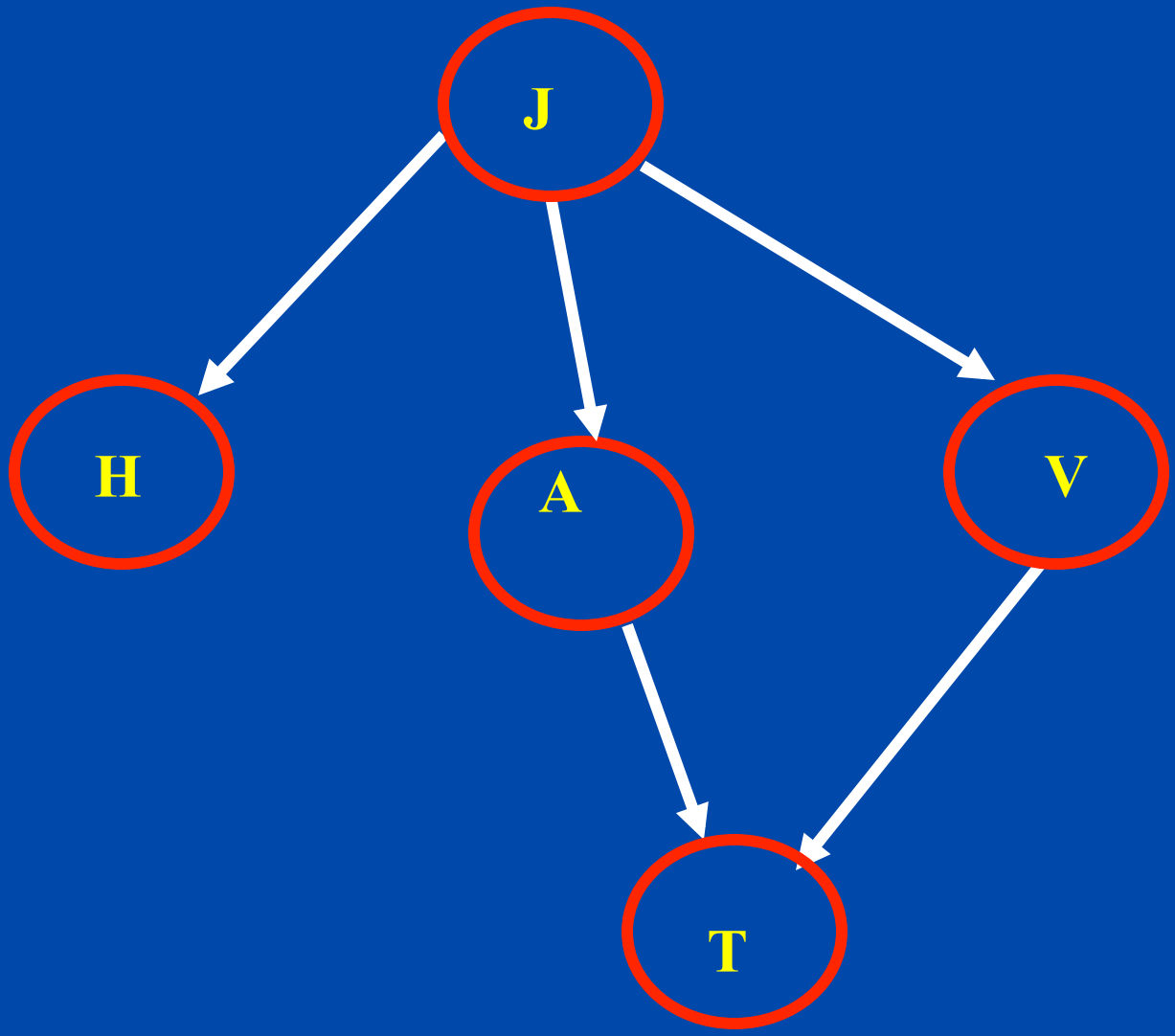
$$P(A) \sim N_a / N_t$$

$$P(B|A1, ..., An) \sim N_{a1, ..., an, b} / N_{a1, ..., an}$$

Ejemplo - ¿Cuándo jugar golf?

Ambiente	Temp.	Humedad	Viento	Jugar
soleado	alta	alta	no	N
soleado	alta	alta	si	N
nublado	alta	alta	no	P
lluvia	media	alta	no	P
lluvia	baja	normal	no	P
lluvia	baja	normal	si	N
nublado	baja	normal	si	P
soleado	media	alta	no	N
soleado	baja	normal	no	P
lluvia	media	normal	no	P
soleado	media	normal	si	P
nublado	media	alta	si	P
nublado	alta	normal	no	P
lluvia	media	alta	si	N

Ejemplo – estructura



Ejemplo

- P(J)
  - P(N) = 5/14
  - P(P) = 9/14
- P(V|J)
  - P(si|N)=3/5, P(si|P)=3/9
  - P(no|N)=2/5, P(no|P)=6/9
- Etc.

Información incompleta

- En la práctica, en muchas ocasiones los datos no están completos
- Dos tipos básicos de información incompleta:
  - Faltan algunos valores de una de las variables en algunos casos – **datos incompletos**
  - Faltan todos los valores de una variable – **nodos ocultos**

Información incompleta

Ambiente	Temp.	Humedad	Viento	Jugar
soleado	xxx	alta	--	N
soleado	alta	alta	--	N
nublado	alta	alta	--	P
lluvia	media	alta	--	P
lluvia	baja	normal	--	P
lluvia	baja	normal	--	N
nublado	baja	normal	--	P
soleado	media	alta	--	N
soleado	xxx	normal	--	P
lluvia	media	normal	--	P
soleado	media	normal	--	P
nublado	media	alta	--	P
nublado	alta	normal	--	P
lluvia	media	alta	--	N

© L.E. Sucar: MGP - Aprend. RB

Datos incompletos

Existen varias alternativas:

1. Considerar un nuevo valor “desconocido”
2. Tomar el valor más probable (promedio) de la variable
3. Considerar el valor más probable en base a las otras variables
4. Considerar la probabilidad de los diferentes valores en base a las otras variables

Datos incompletos

Valor más probable:

1. Asignar todas las variables observables.
2. Propagar su efecto y obtener las probabilidades posteriores de las no observables.
3. Para las variables no observables, asumir el valor con probabilidad mayor como observado.
4. Actualizar las probabilidades previas y condicionales de acuerdo a las fórmulas anteriores.
5. Repetir 1 a 4 para cada observación.

Datos incompletos

Ambiente	Temp.	Humedad	Viento	Jugar
soleado	xxx	alta	--	N
soleado	alta	alta	--	N
nublado	alta	alta	--	P
lluvia	media	alta	--	P
lluvia	baja	normal	--	P
lluvia	baja	normal	--	N
nublado	baja	normal	--	P
soleado	media	alta	--	N
soleado	xxx	normal	--	P
lluvia	media	normal	--	P
soleado	media	normal	--	P
nublado	media	alta	--	P
nublado	alta	normal	--	P
lluvia	media	alta	--	N

P(T|sol,alta,N)

P(T|sol,nor,P)

Datos incompletos

Ambiente	Temp.	Humedad	Viento	Jugar
soleado	media	alta	--	N
soleado	alta	alta	--	N
nublado	alta	alta	--	P
lluvia	media	alta	--	P
lluvia	baja	normal	--	P
lluvia	baja	normal	--	N
nublado	baja	normal	--	P
soleado	media	alta	--	N
soleado	media	normal	--	P
lluvia	media	normal	--	P
soleado	media	normal	--	P
nublado	media	alta	--	P
nublado	alta	normal	--	P
lluvia	media	alta	--	N

P(T|sol,alta,N)

P(T|sol,nor,P)

Nodos ocultos – algoritmo EM

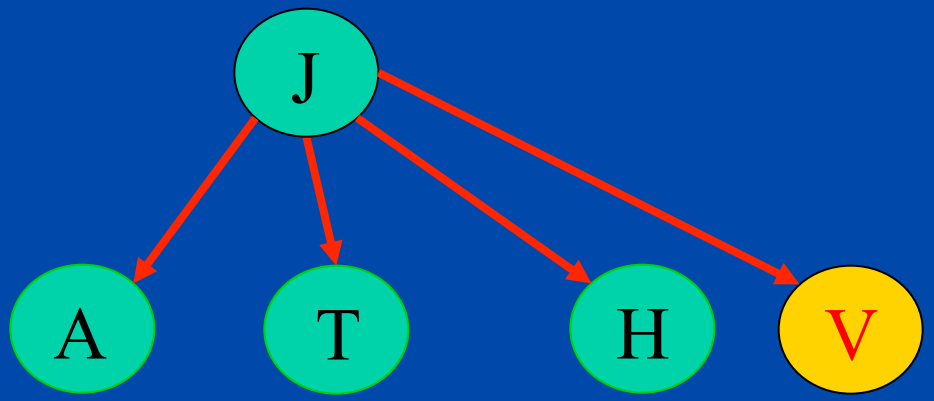
- El algoritmo EM es un método estadístico muy utilizado para estimar probabilidades cuando hay variables no observables (un caso especial es el algoritmo de Baum-Welch en HMM)
- Consiste básicamente de 2 pasos que se repiten en forma iterativa:

1. Paso E: se estiman los datos faltantes en base a los parámetros (P) actuales
2. Paso M: se estiman las probabilidades (parámetros) considerando los datos estimados

EM para RB con nodos ocultos

1. Iniciar los parámetros desconocidos (CPTs) con valores aleatorios (o estimaciones de expertos)
2. Utilizar los datos conocidos con los parámetros actuales para estimar los valores de la variable(s) oculta(s)
3. Utilizar los valores estimados para completar la tabla de datos
4. Re-estimar los parámetros con los nuevos datos
5. Repetir 2→4 hasta que no haya cambios significativos en las probabilidades

Ejemplo



- V es un nodo oculto
- Se seleccionan valores aleatorios para P(V|J)
- Se calcula la probabilidad de V para cada caso, dados los valores de A, T, H, J
- Cada caso se “pesa” de acuerdo a las probabilidades posteriores de V (un caso puede representar “n” datos)
- Se recalculan los parámetros (P(V|J)) en base a los casos obtenidos
- Se repite el proceso hasta que converja

EM: inicio

Ambiente	Temp.	Humedad	Viento	Jugar
soleado	media	alta	--	N
soleado	alta	alta	--	N
nublado	alta	alta	--	P
lluvia	media	alta	--	P
lluvia	baja	normal	--	P
lluvia	baja	normal	--	N
nublado	baja	normal	--	P
soleado	media	alta	--	N
soleado	media	normal	--	P
lluvia	media	normal	--	P
soleado	media	normal	--	P
nublado	media	alta	--	P
nublado	alta	normal	--	P
lluvia	media	alta	--	N

"Adivinar"
P(V | J):

V\J	N	P
no	0.5	0.5
si	0.5	0.5

EM: paso E

Ambiente	Temp.	Humedad	Viento	Jugar
soleado	media	alta	no	N
soleado	alta	alta	no	N
nublado	alta	alta	no	P
lluvia	media	alta	no	P
lluvia	baja	normal	si	P
lluvia	baja	normal	si	N
nublado	baja	normal	si	P
soleado	media	alta	no	N
soleado	media	normal	no	P
lluvia	media	normal	no	P
soleado	media	normal	si	P
nublado	media	alta	si	P
nublado	alta	normal	si	P
lluvia	media	alta	si	N

Estimar valores de V en base a P(V | J) y los datos

EM: paso M

Ambiente	Temp.	Humedad	Viento	Jugar
soleado	media	alta	no	N
soleado	alta	alta	no	N
nublado	alta	alta	no	P
lluvia	media	alta	no	P
lluvia	baja	normal	si	P
lluvia	baja	normal	si	N
nublado	baja	normal	si	P
soleado	media	alta	no	N
soleado	media	normal	no	P
lluvia	media	normal	no	P
soleado	media	normal	si	P
nublado	media	alta	si	P
nublado	alta	normal	si	P
lluvia	media	alta	si	N

Re-estimar P(V | J) con los Nuevos datos:

V\J	N	P
no	0.6	0.44
si	0.4	0.66

Aprendizaje Estructural

Diversos métodos:

- Aprendizaje de árboles
- Aprendizaje de poliárboles
- Aprendizaje de redes multiconectadas
  - Métodos basados en medidas
  - Métodos basados en relaciones de dependencia

Aprendizaje de árboles

- Algoritmo desarrollado por Chow y Liu para aproximar una distribución de probabilidad por un producto de probabilidades de segundo orden (árbol).
- La probabilidad conjunta de n variables se puede representar como:

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid X_{j(i)})$$

- donde $X_{j(i)}$ es la causa o padre de X_i.

Aprendizaje de árboles

- Se plantea el problema como uno de optimización - obtener la estructura que más se aproxime a la distribución "real".
- Medida de la diferencia de información entre la distribución real (P) y la aproximada (P*):

$$I(P,P^*)=\sum_x P(X)\log\frac{P(X)}{P^*(X)}$$

- El objetivo es minimizar I.

Aprendizaje de árboles

- La información mutua entre pares de variables se define como:

$$I(X_i, X_j) = \sum_{x_i, x_j} P(X_i, X_j) \log \frac{P(X_i, X_j)}{P(X_i)P(X_j)}$$

- Se puede demostrar (Chow 68) que la diferencia de información es una función del negativo de la suma de las informaciones mutuas (pesos) de todos los pares de variables que constituyen el árbol.
- Encontrar el árbol más próximo equivale a encontrar el árbol con mayor peso.

Aprendizaje de árboles - algoritmo

1. Calcular la información mutua entre todos los pares de variables (n(n - 1)/2).
2. Ordenar las informaciones mutuas de mayor a menor.
3. Seleccionar la rama de mayor valor como árbol inicial.
4. Agregar la siguiente rama mientras no forme un ciclo, si es así, desechar.
5. Repetir (3-4) hasta que se cubran todas las variables (n -1 ramas).

- El algoritmo NO provee la dirección de los arcos, por lo que ésta se puede asignar en forma arbitraria o utilizando semántica externa (experto).

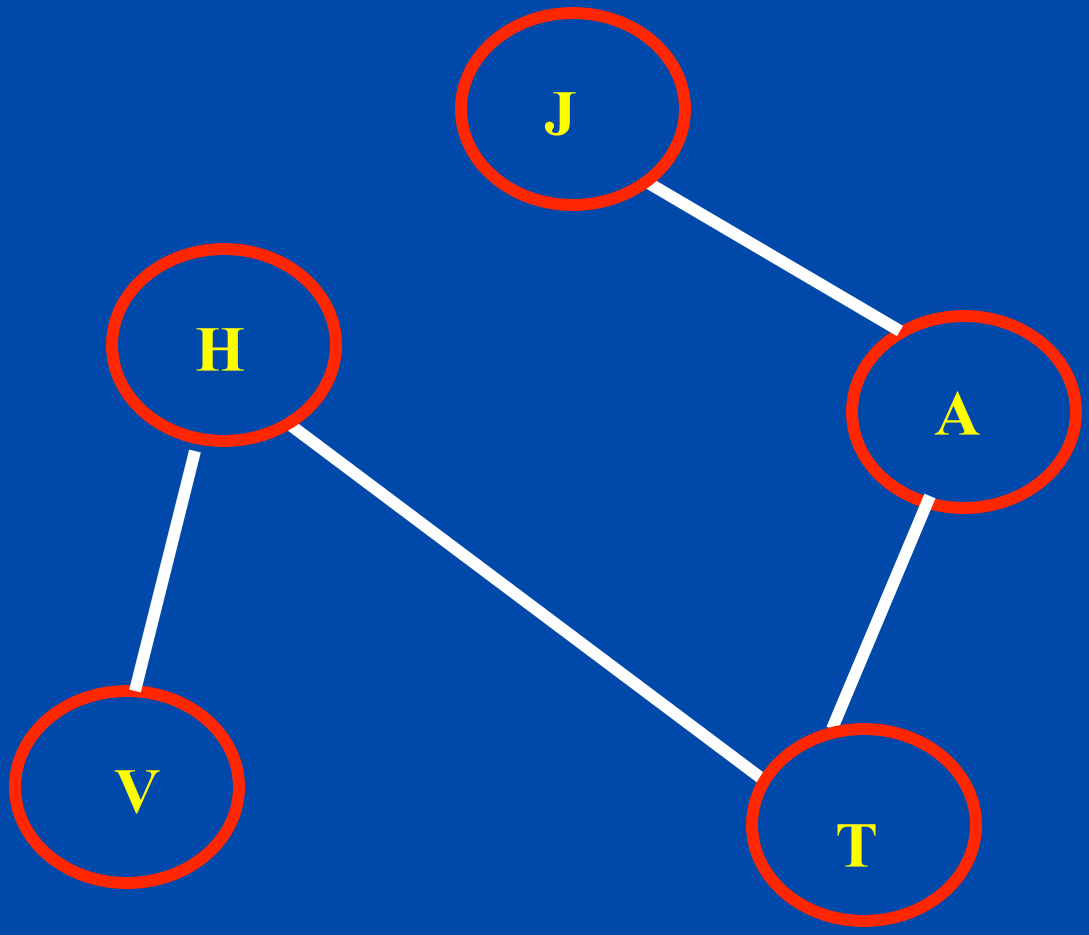
Ejemplo (golf)

- Informaciones mutuas ordenadas

No.	Var 1	Var 2	I.M.
1	temp.	humedad	.1128
2	humedad	viento	.0860
3	ambiente	juega	.0745
4	ambiente	temp.	.0074
5	humedad	juega	.0457
6	viento	juega.	.0145
7	temp.	juega	…
8	viento	ambiente	…
9	humedad	viento	…
10	viento	temp.	…

© L.E. Sucar: MGP - Aprend. RB

Ejemplo (golf)



Aprendizaje de poliárboles

- Parte del esqueleto (estructura sin direcciones) obtenido con el algoritmo anterior
- Determina la dirección de los arcos utilizando pruebas de dependencia entre tripletas de variables.
- Dadas 3 variables, existen 3 casos posibles:
  - Arcos divergentes
  - Arcos secuenciales
  - Arcos convergentes
- Los primeros dos casos son indistinguibles, pero el tercero es diferente, ya que las dos variables "padre" son marginalmente independientes.

Prueba de Tripletas

- Tripleta de variables:

$$X - Z - Y$$

- Si X – Y son independientes dado Z, entonces pueden ser secuenciales o divergentes

$$X \rightarrow Z \rightarrow Y;\ X \leftarrow Z \rightarrow Y$$

- Si X – Y no son independientes dado Z, entonces son arcos convergentes

$$X \rightarrow Z \leftarrow Y$$

Aprendizaje de poliárboles - algoritmo

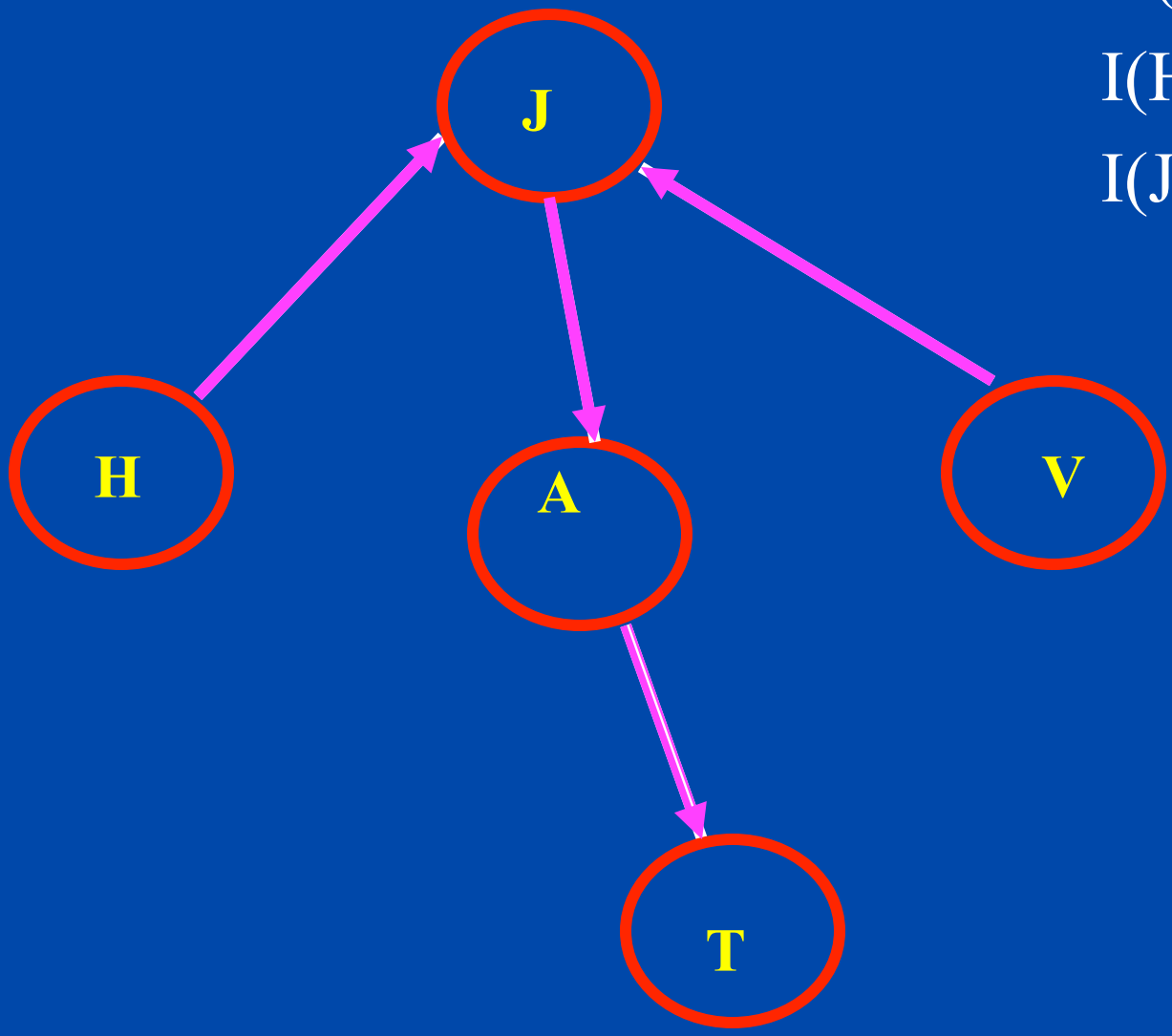
1. Obtener esqueleto utilizando el algoritmo de Chow y Liu
2. Recorrer la red hasta encontrar una tripleta de nodos que sean convergentes (tercer caso) - nodo multipadre-
3. A partir de un nodo multipadre determinar las direcciones de los arcos utilizando la prueba de tripletas hasta donde sea posible (base causal).
4. Repetir 2-3 hasta que ya no se puedan descubrir más direcciones.
5. Si quedan arcos sin direccionar, utilizar semántica externa para obtener su dirección (o fijar direcciones).

Ejemplo

~I(H,J,V)

I(H,J,A)

I(J,A,T)



Aprendizaje de redes multiconectadas

Existen dos tipos de métodos para el aprendizaje genérico de redes bayesianas:

1. Métodos basados en medidas de ajuste y búsqueda
2. Métodos basados en pruebas de independencia

Métodos basados en medidas

Se generan diferentes estructuras y se evalúan respecto a los datos utilizando alguna medida

Dos aspectos principales:

- Medida de "ajuste" de la estructura a los datos
- Búsqueda de la "mejor" estructura

Medidas

- Evalúan que tan “buena” es una estructura respecto a los datos
- Hay varias posibles medidas, las dos más comunes son:
  - Medida bayesiana
  - Medida basada en el principio de longitud de descripción mínima (MDL)

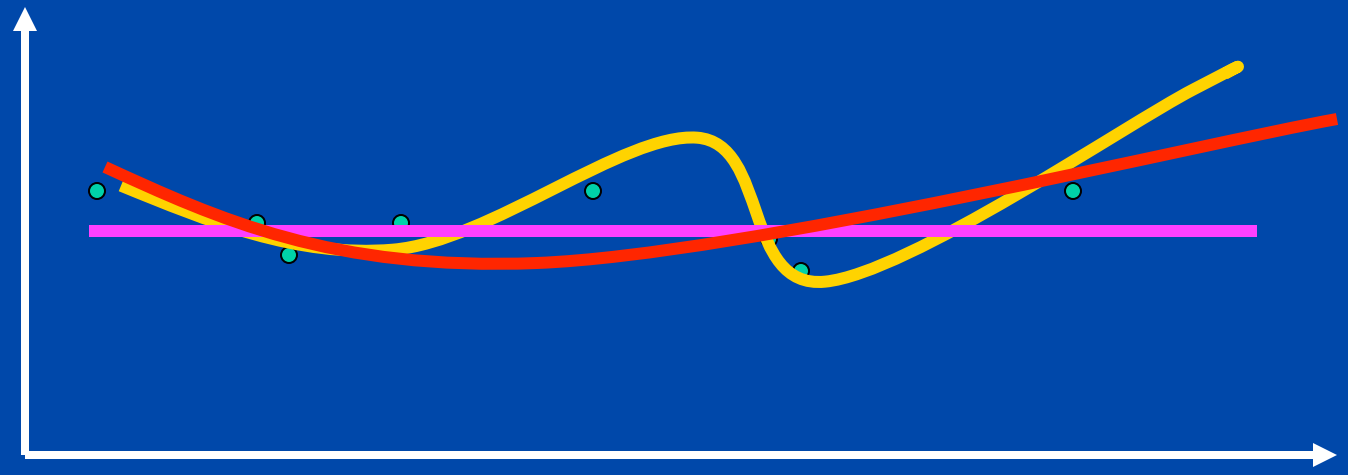
MDL

- La “calidad” de la estructura se basa en el principio de “descripción de longitud mínima” (MDL):
  - Tamaño de la descripción de la red (complejidad)
  - Tamaño de error de predicción de los datos por la red (exactitud)
- Se hace una búsqueda heurística de la estructura en base al MDL

MDL

Compromiso entre exactitud y complejidad-minimizar: ***long. de descripción del modelo +***
descripción de lo datos dado el modelo

Ejemplo – ajustar un polinomio a un conjunto de puntos:



MDL

Para redes bayesianas:

Complejidad:

$$L = \sum_i [\, k_i \log_2 n + d(S_i - 1) \prod_{Fi} s_i]$$

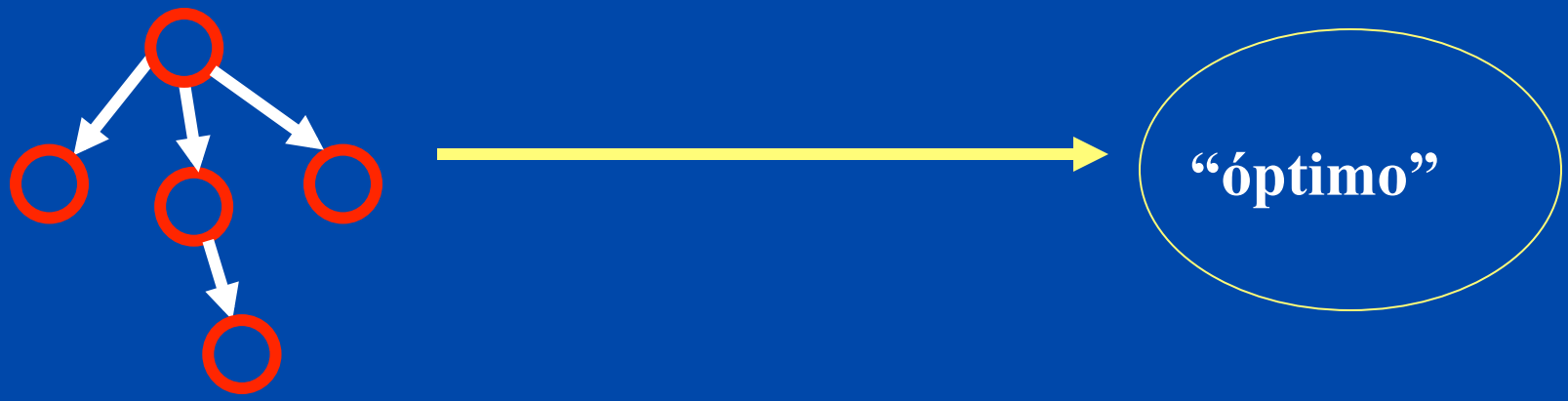
n-# de nodos, k-# padres por nodo, Si-# de valores por variable, Fi-conj. de padres, d-# de bits

Exactitud:

$$w(x_i, Fx_i) = \sum P(x_i, Fx_i) \log_2 [P(x_i,Fx_i)/P(x_i)P(Fx_i)]$$

$$W = \sum_i w(x_i, Fx_i)$$

Buscando la mejor estructura



- Búsqueda de ascenso de colinas (*hill climbing*)
- Se inicia con una estructura simple (árbol) y se van agregando arcos hasta llegar a un *mínimo local*

Métodos basados en medidas

- Se genera la estructura en base a ir agregando/eliminando arcos de acuerdo a medidas de dependencia entre variables
- Ejemplos:
  - Árboles – método de Chow y Liu
  - Poliárboles – método de Rebane y Pearl
  - Multiconectadas – existen varios algoritmos basados en diferentes medidas

Algoritmo PC

- Se basa en pruebas de independencia entre variables:

$$I\,(Xi, Xj \mid A)$$

- Donde A es un subconjunto de variables
- Asume que:
  - Se tienen suficientes datos
  - Las pruebas estadísticas no tienen errores

Prueba de Independencia

- Para probar si X, Y son independientes dado A se utiliza la entropía cruzada condicional:

$CE(X,Y \mid Z) =$

$$\Sigma_z P(z) \Sigma_{x,y} P(x,y|z) \log [P(x,y|z) / P(x|z) P(y|z)]$$

- Si es cero o cercana a cero, quiere decir que son independientes (se puede usar un umbral o una prueba estadística con cierto nivel de significancia)

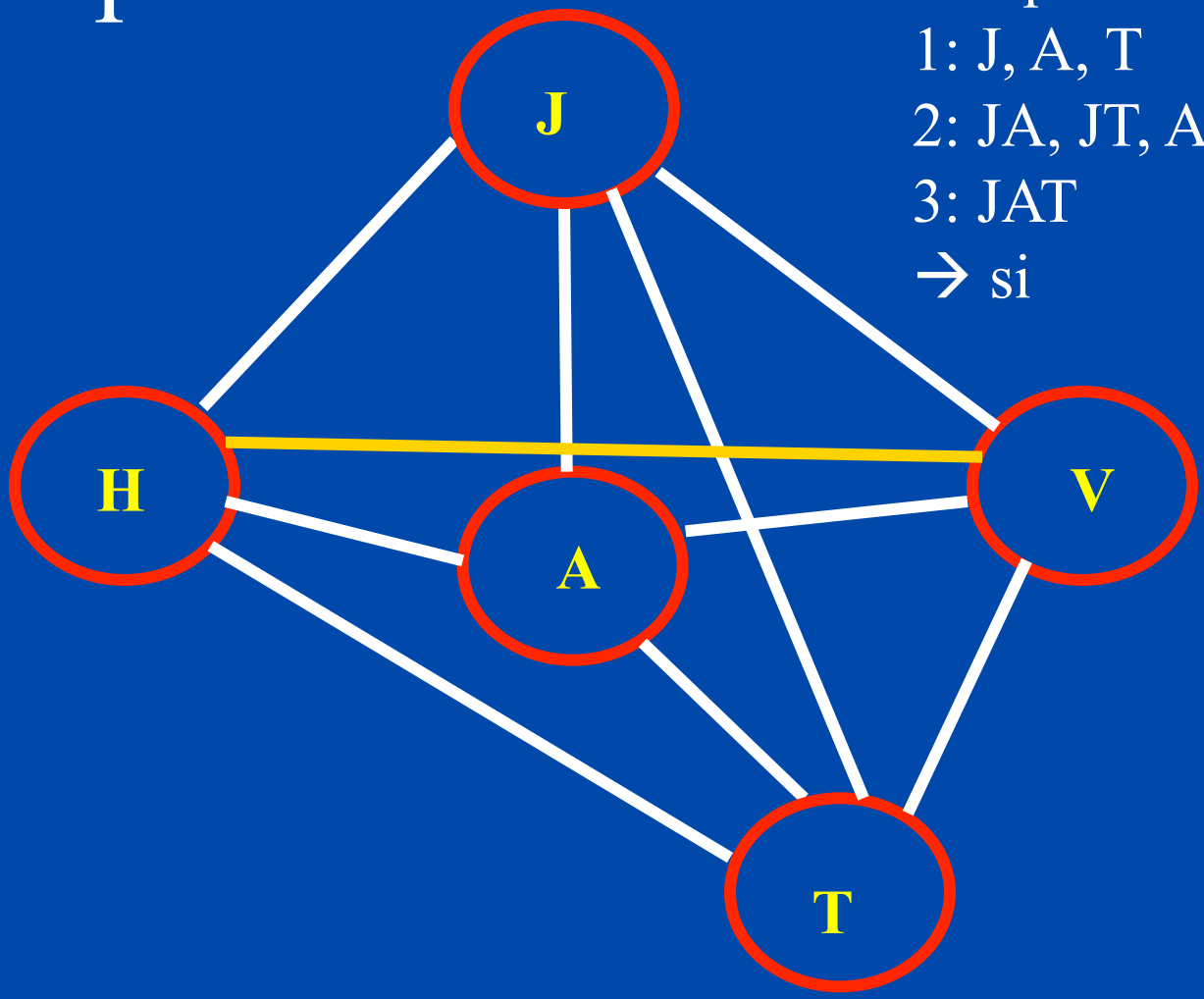
Algoritmo

1. Encontrar un "esqueleto" (grafo no dirigido)
2. Encontrar arcos convergentes en tripletas de variables por pruebas de independencia
3. Orientar el resto de las ligas de forma que no se produzcan ciclos

Esqueleto

- La idea básica para determinar el esqueleto es iniciar con un grafo completo (conectando todos vs. todos los nodos) y eliminar el arco entre X – Y si hay un subconjunto de nodos en G (excepto X, Y) que los hace independientes
- En principio se consideran todos los posibles subconjuntos de variables, de tamaño 1 hasta de tamaño N-1 (N es el número de nodos adyacentes a X y Y)
- El considerar todos los posibles subconjuntos es muy ineficiente, y normalmente se limita a considerar sólo subconjuntos de 1, 2, …, k nodos

Ejemplo



Probar si H,V son Independientes dados:

1: J, A, T

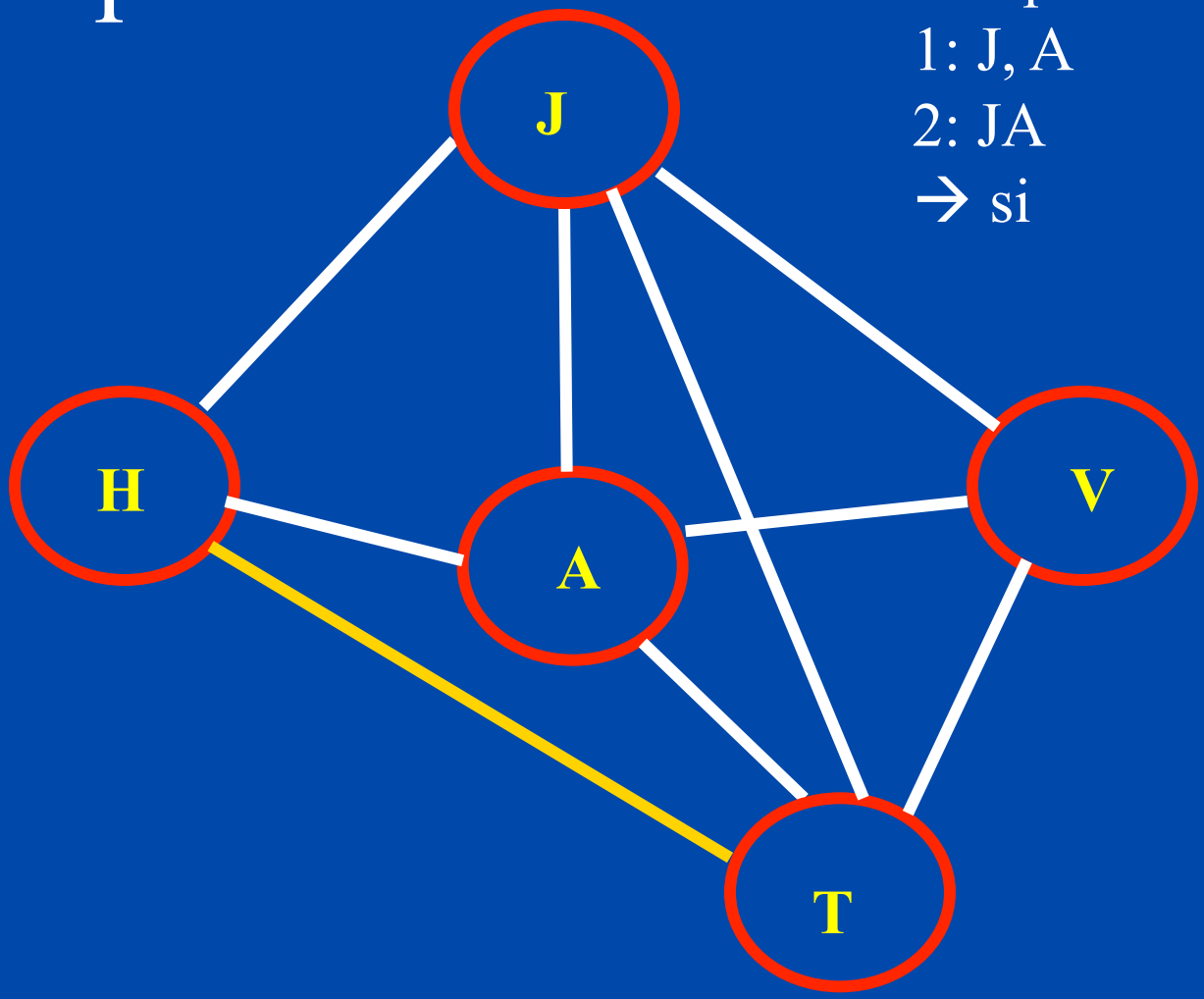
2: JA, JT, AT

3: JAT

→ si

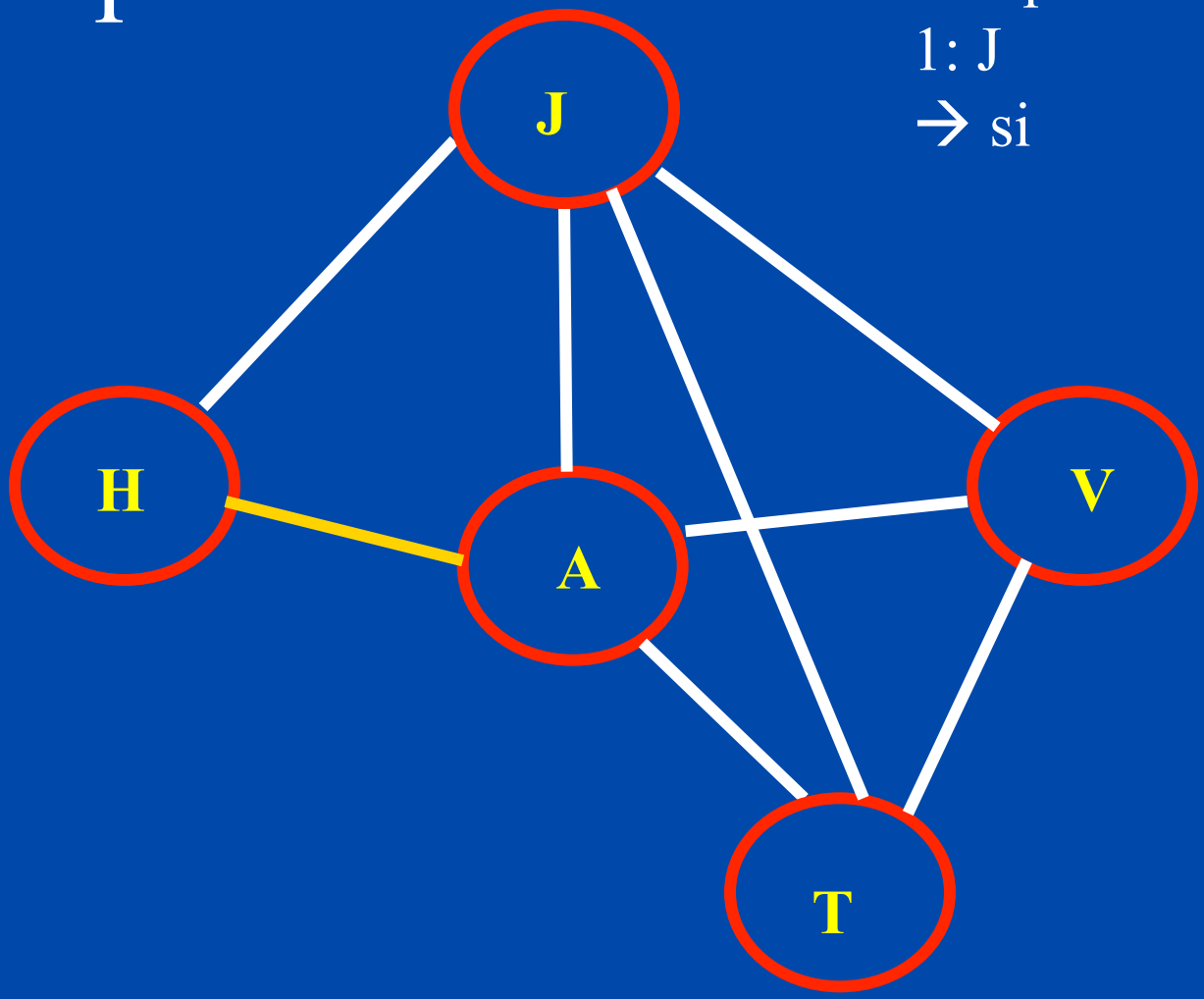
Ejemplo

Probar si H,T son Independientes dados:
1: J, A
2: JA
→ si



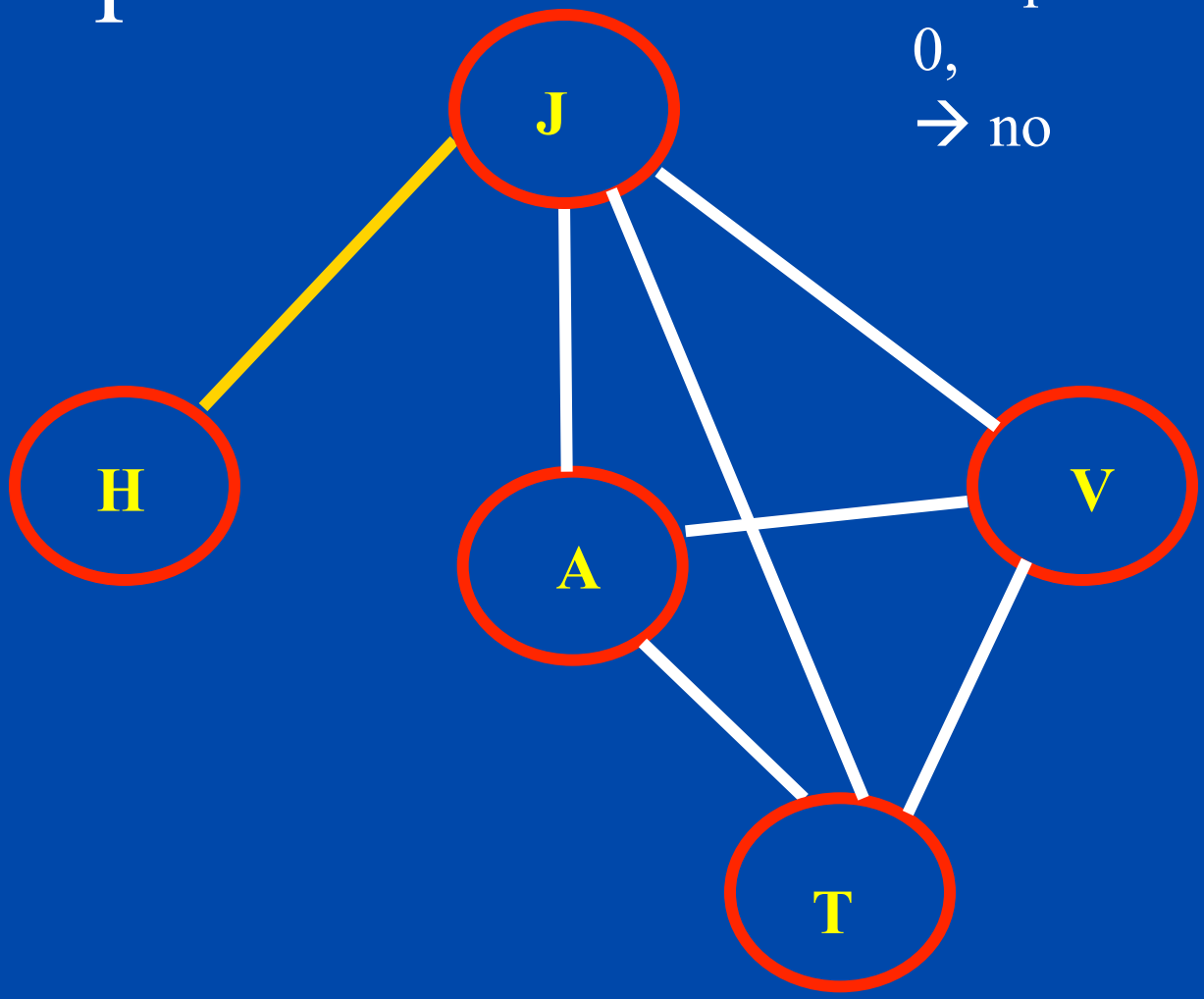
Ejemplo

Probar si H,A son Independientes dados:
1: J
→ si



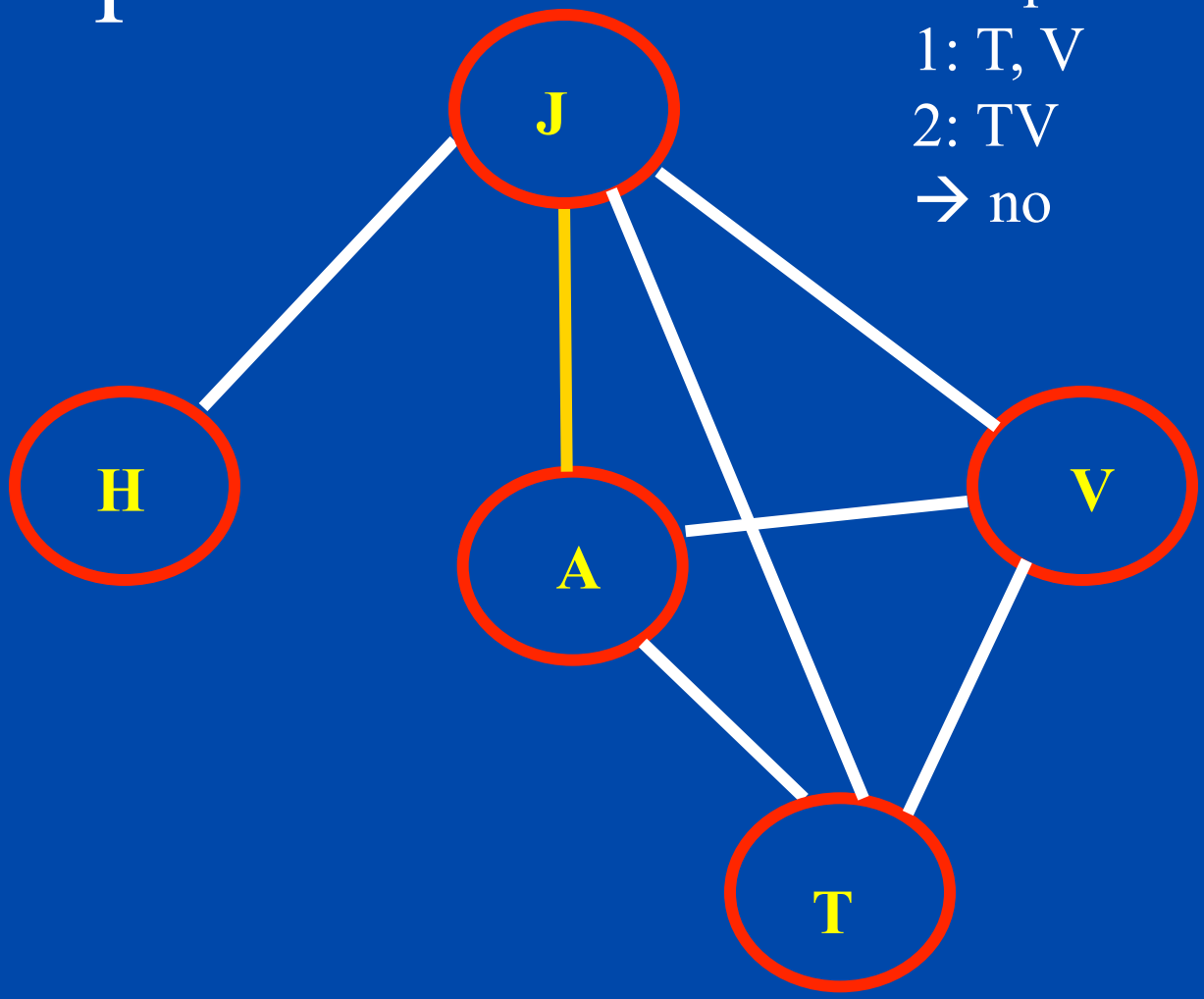
Ejemplo

Probar si H,J son Independientes dados: 0,
→ no



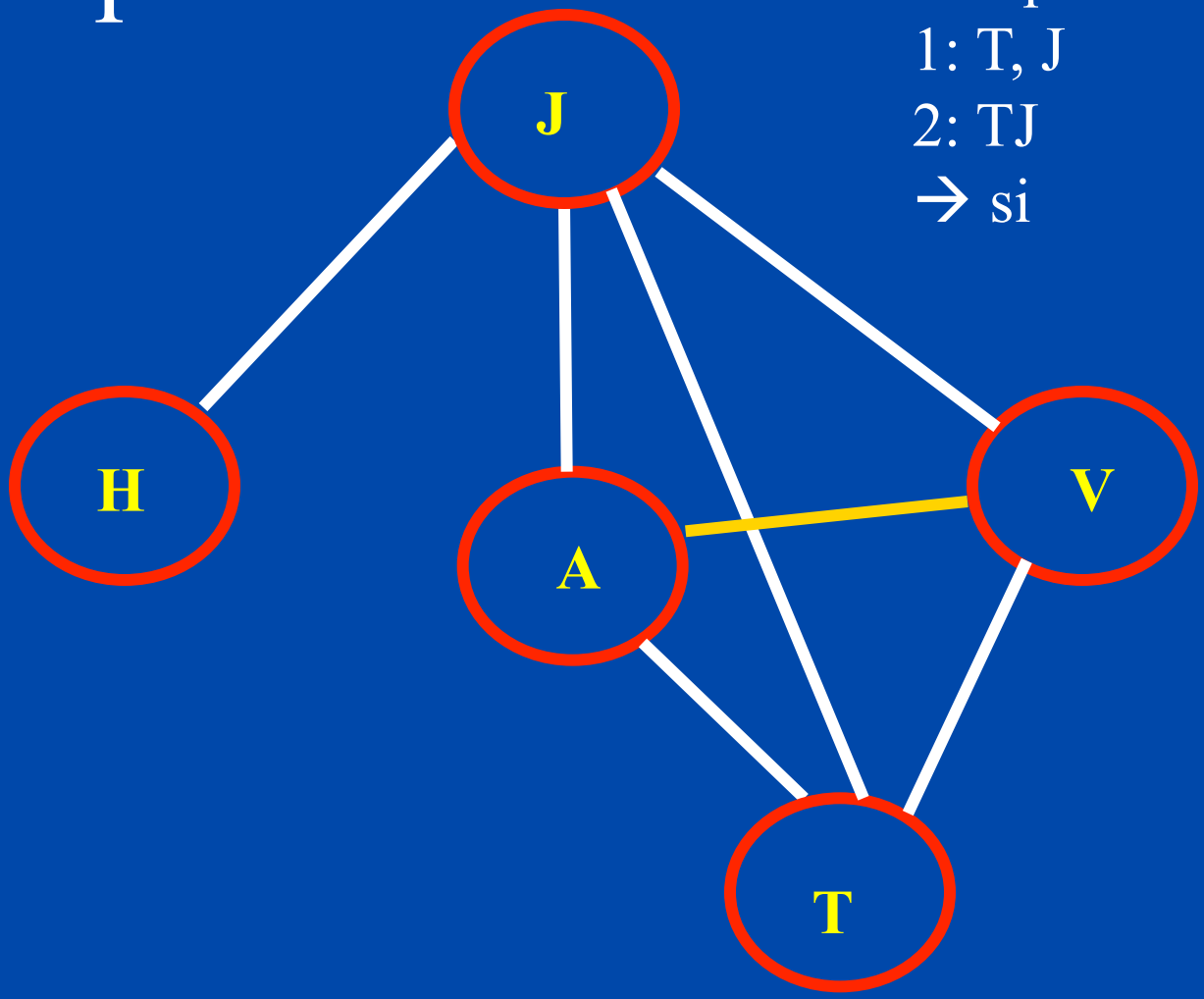
Ejemplo

Probar si A,J son Independientes dados:
1: T, V
2: TV
→ no



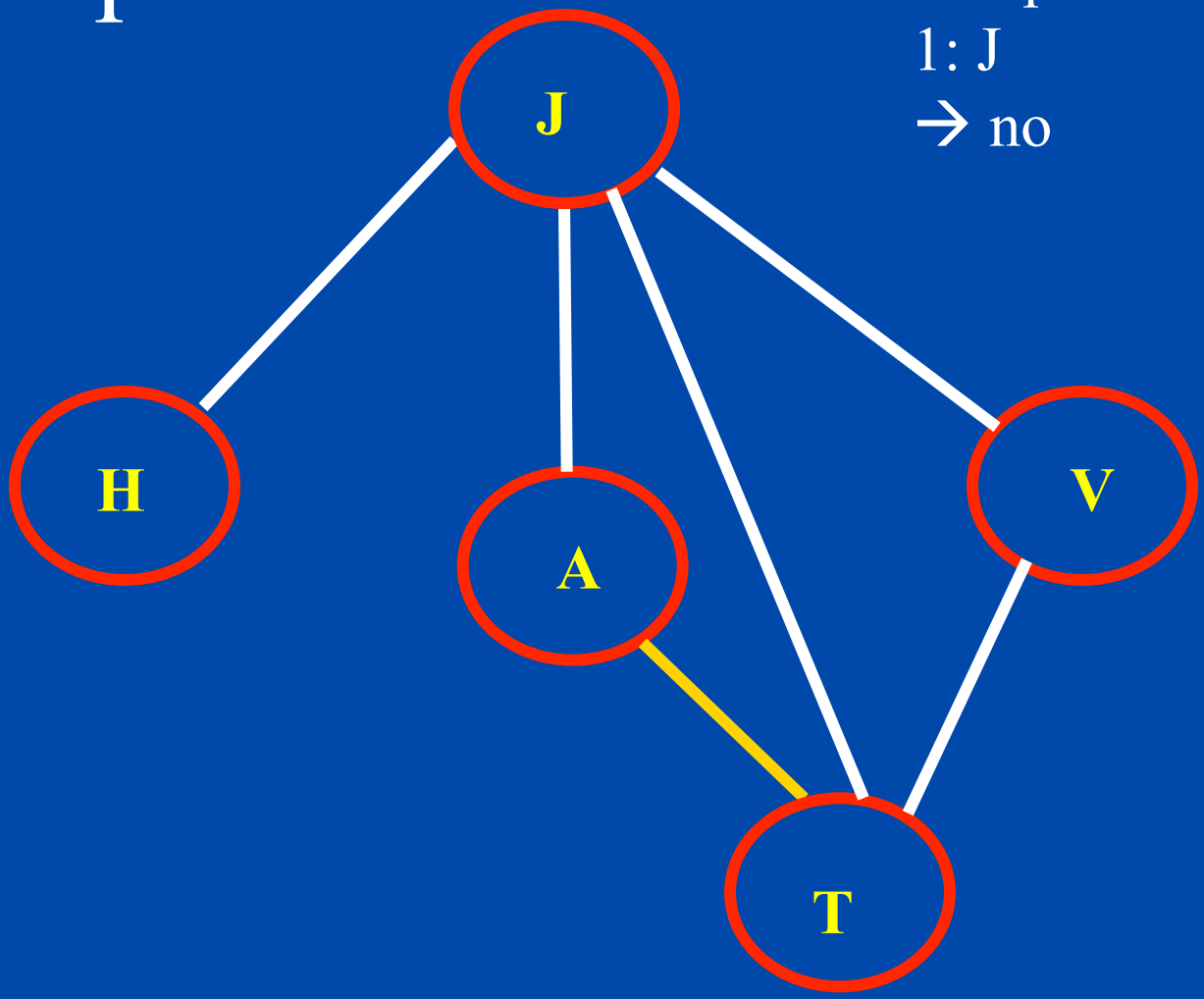
Ejemplo

Probar si A,V son Independientes dados:
1: T, J
2: TJ
→ si



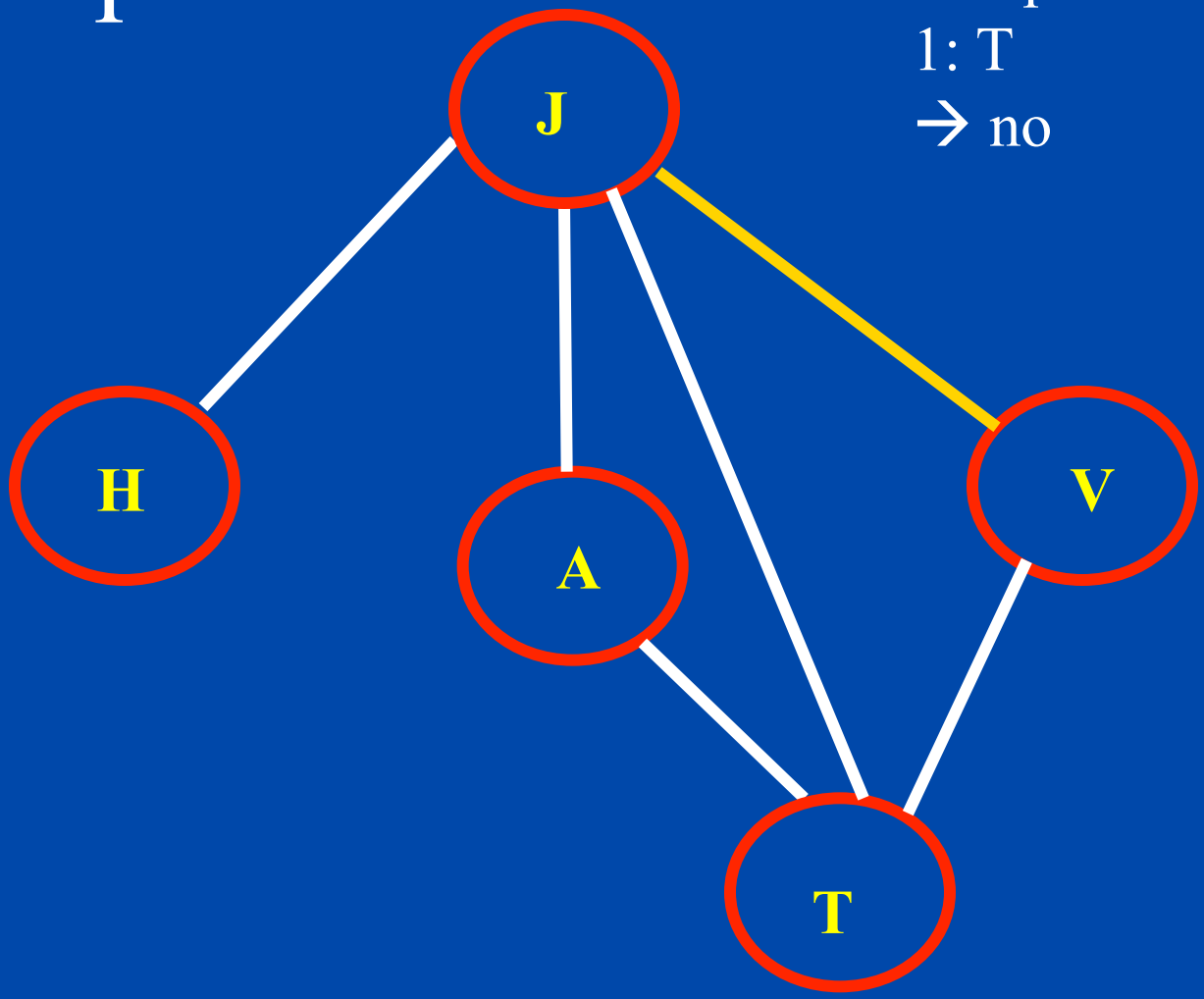
Ejemplo

Probar si A,T son Independientes dados:
1: J
→ no

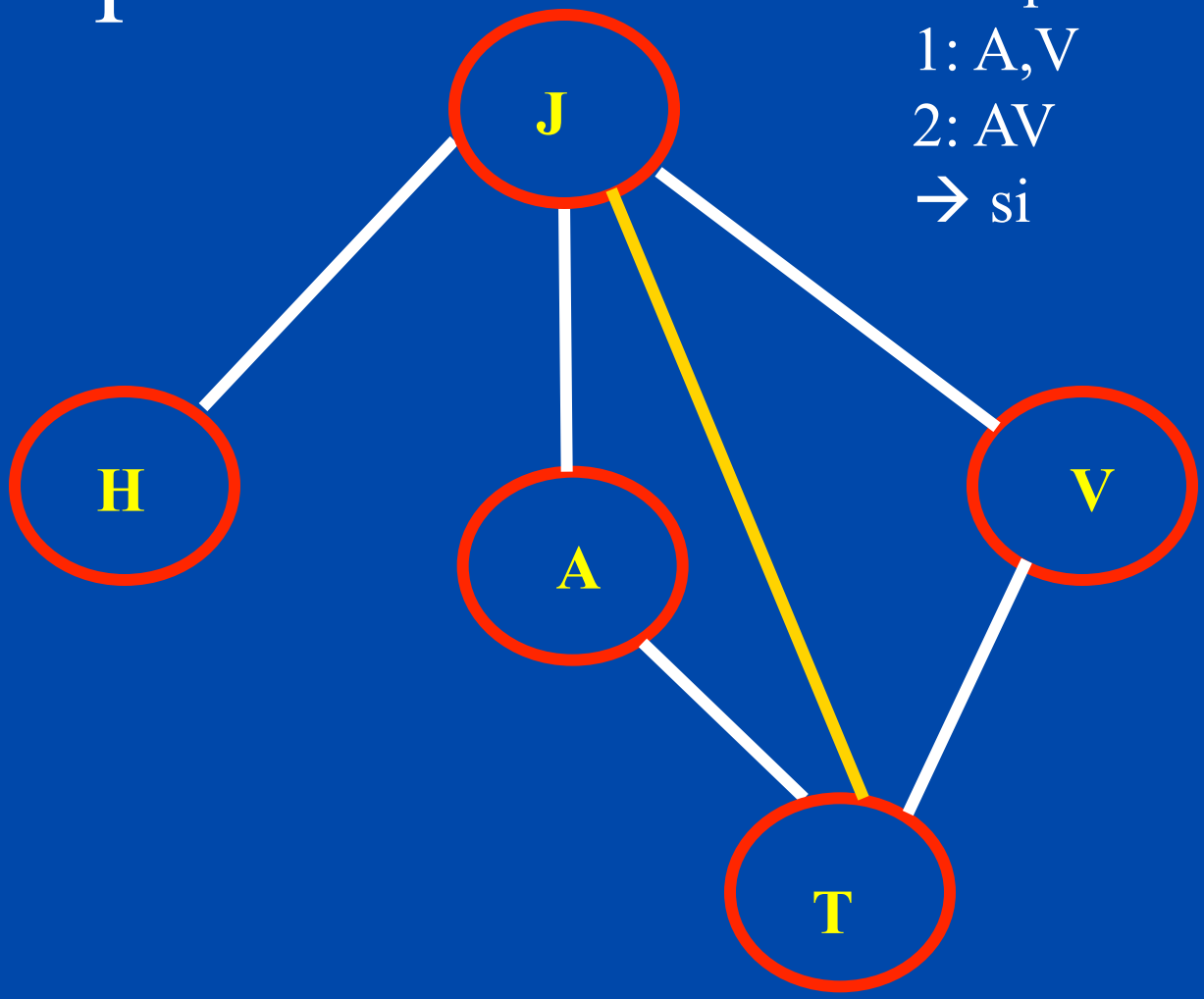


Ejemplo

Probar si J,V son Independientes dados:
1: T
→ no



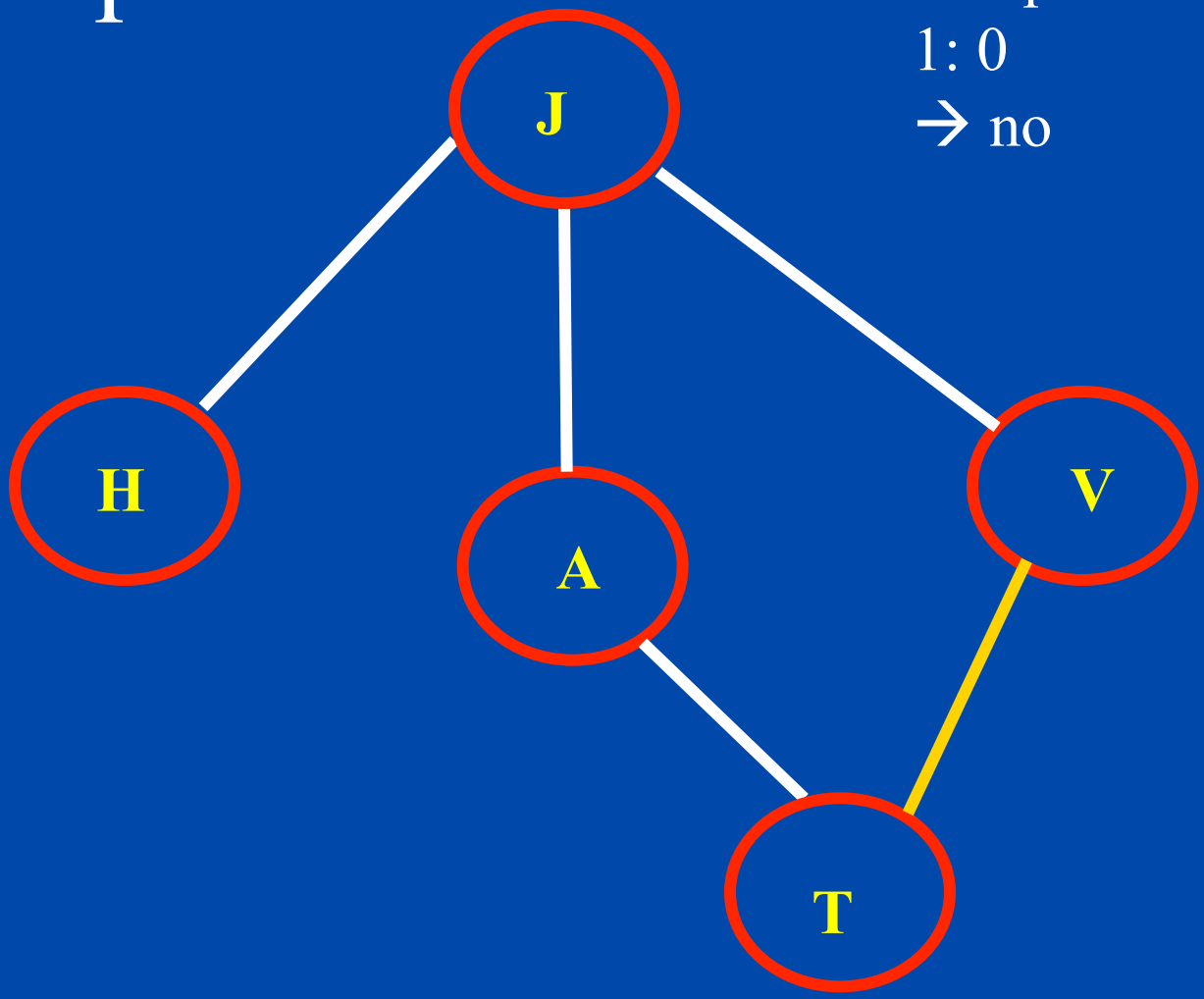
Ejemplo



Probar si J,T son Independientes dados:
1: A,V
2: AV
→ si

Ejemplo

Probar si V,T son Independientes dados:
1: 0
→ no



Arcos convergentes

- Se verifica cada tripleta de variables para encontrar arcos convergentes mediante pruebas de independencia:

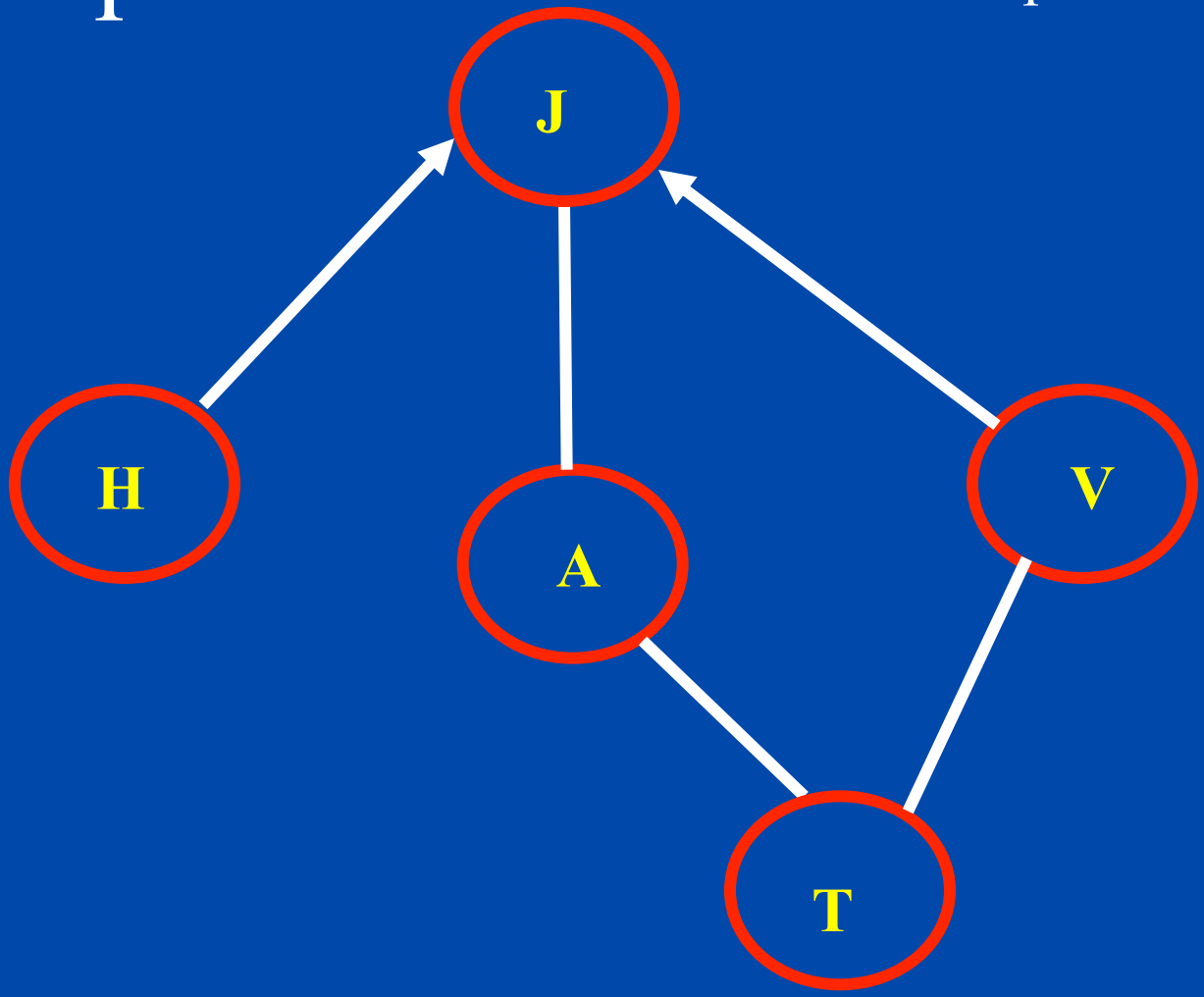
$$X - Z - Y$$

- Si X – Y no son independientes dado Z, entonces son arcos convergentes

$$X \rightarrow Z \leftarrow Y$$

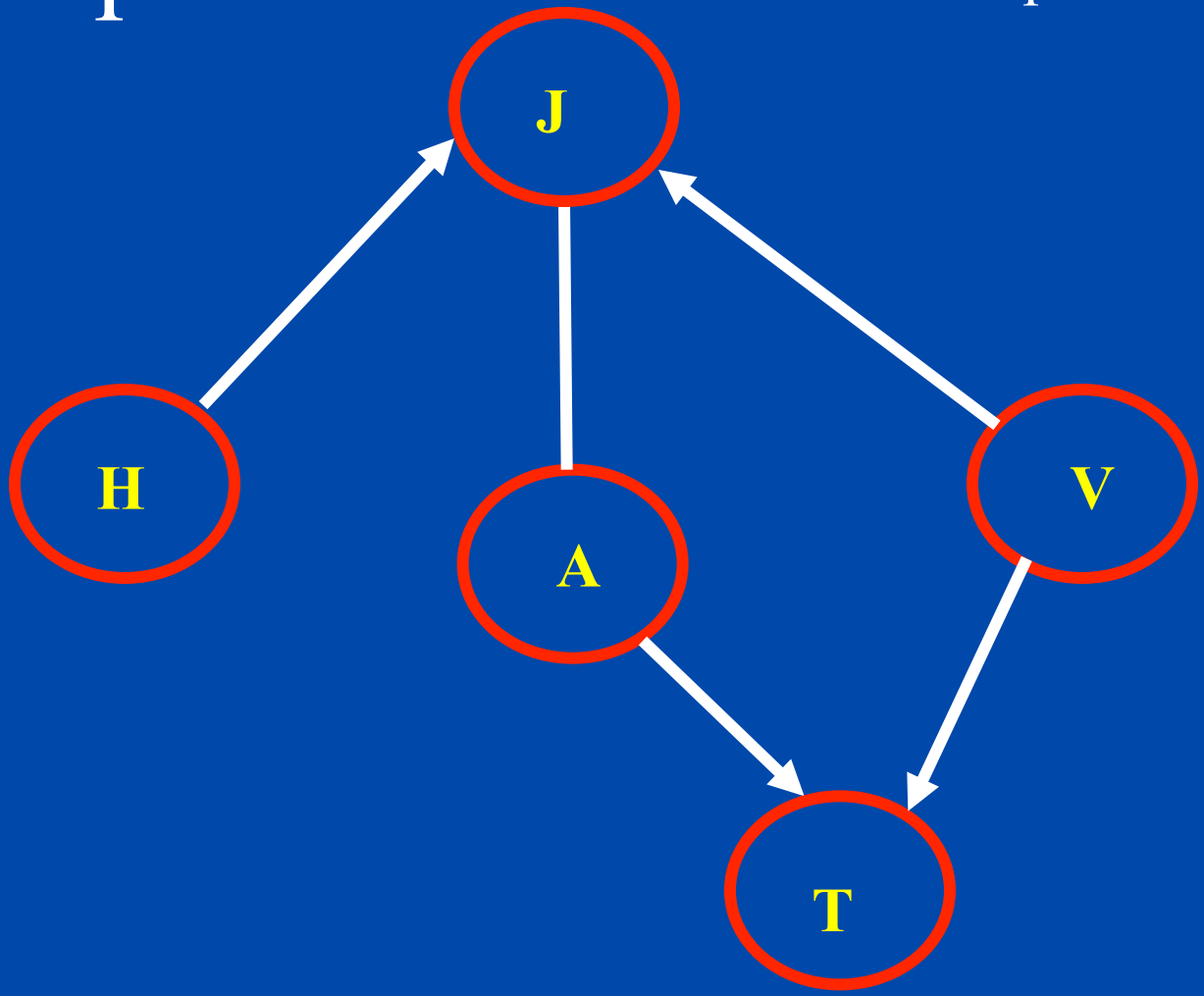
Ejemplo

H,V no son
Independientes dado J



Ejemplo

A,V no son
Independientes dado T

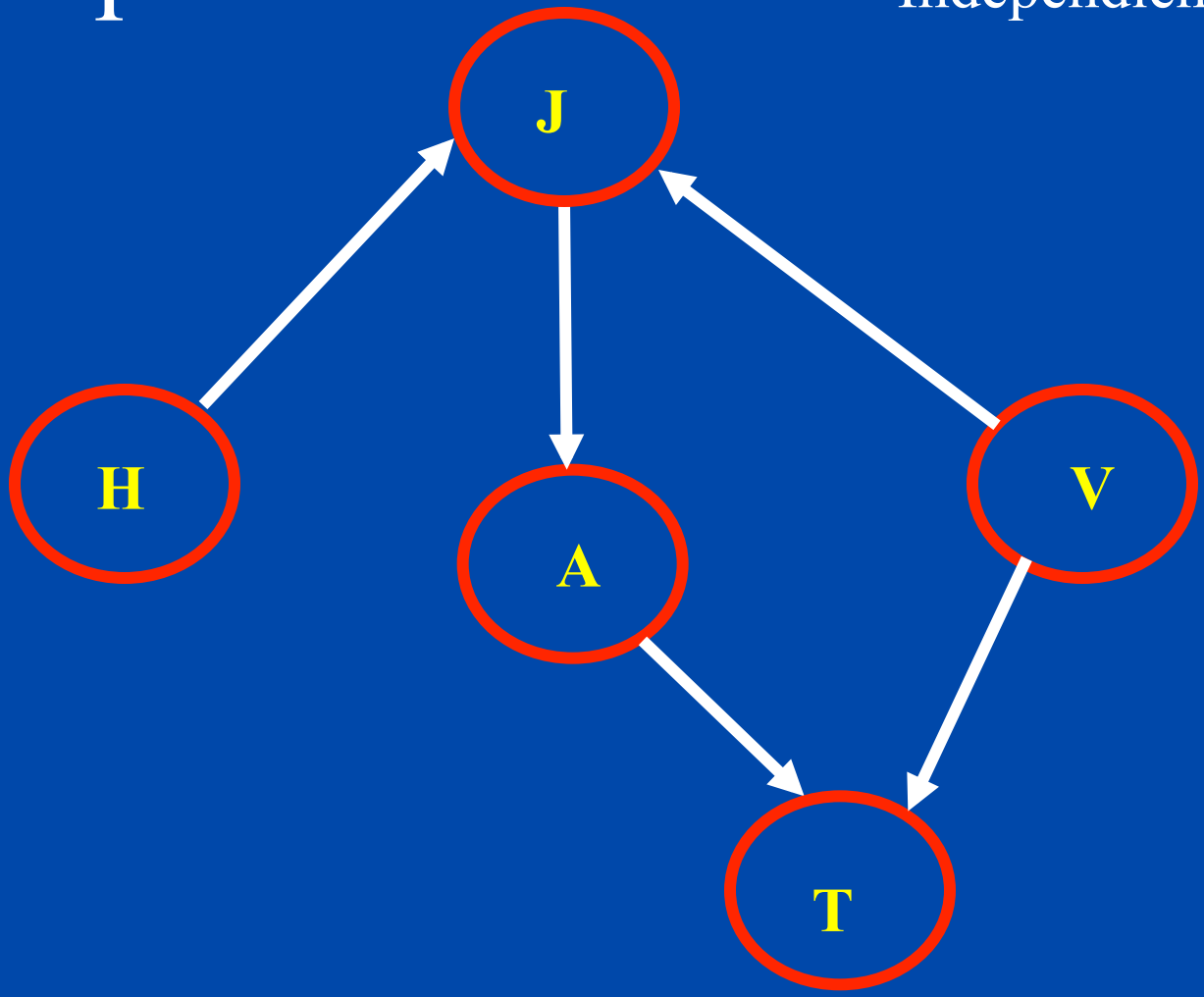


Otras orientaciones

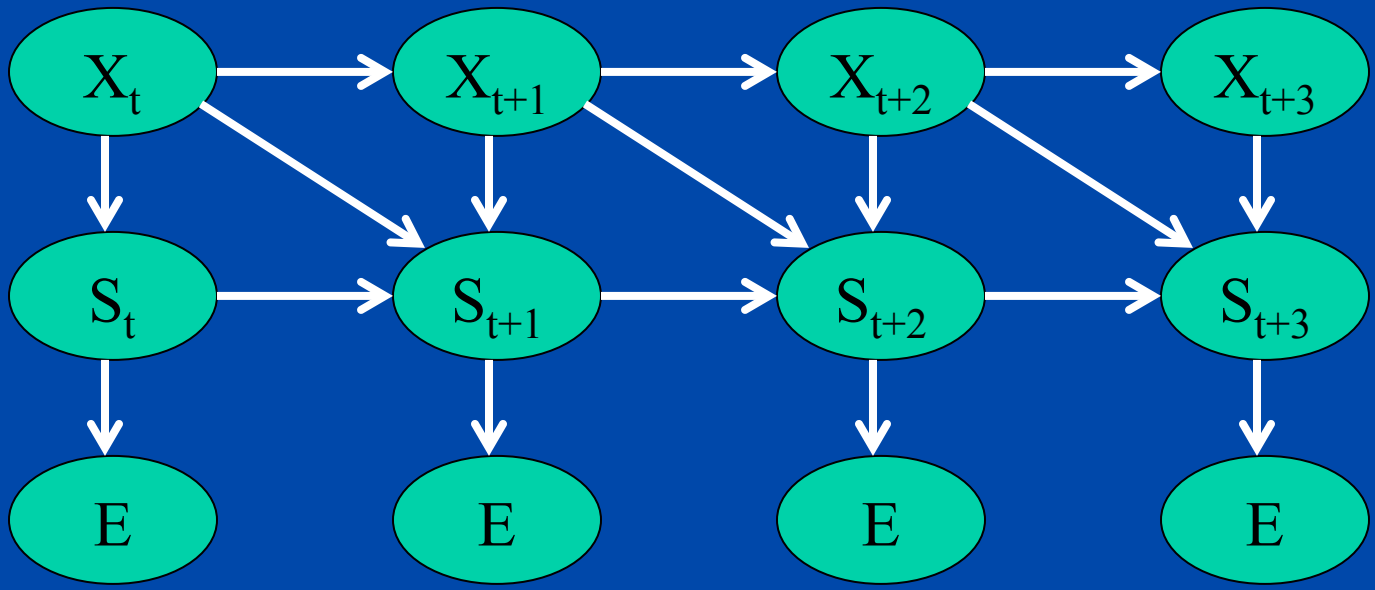
- En base a los arcos existentes, se orientan los demás con pruebas de independencia, evitando crear ciclos
- Si quedan al final arcos sin orientar, se direccionan en forma aleatoria, evitando ciclos

Ejemplo

H, A son
Independientes dado J



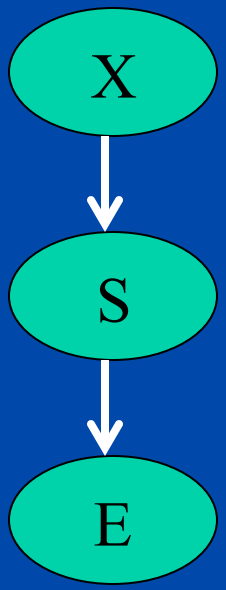
6: Redes Bayesianas Dinámicas



Redes Bayesianas Dinámicas (RBD)

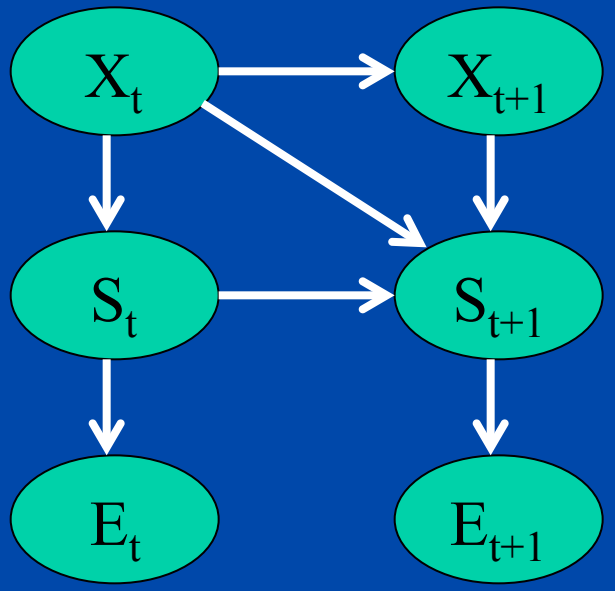
- Representan procesos dinámicos
- Consisten en una representación de los estados del proceso en un tiempo (red base) y las relaciones temporales entre dichos procesos (red de transición)
- Se pueden ver como una generalización de las cadenas (ocultas) de Markov

Red base



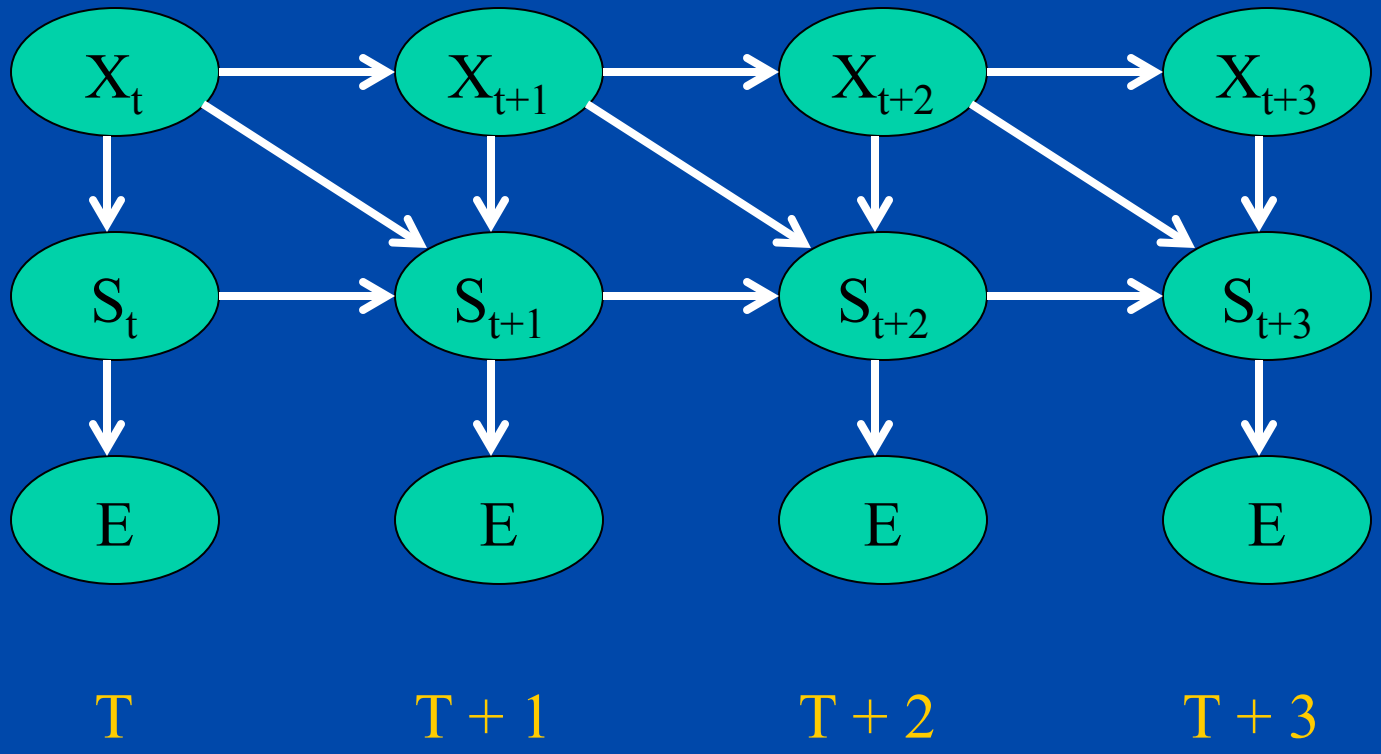
T

Red de Transición

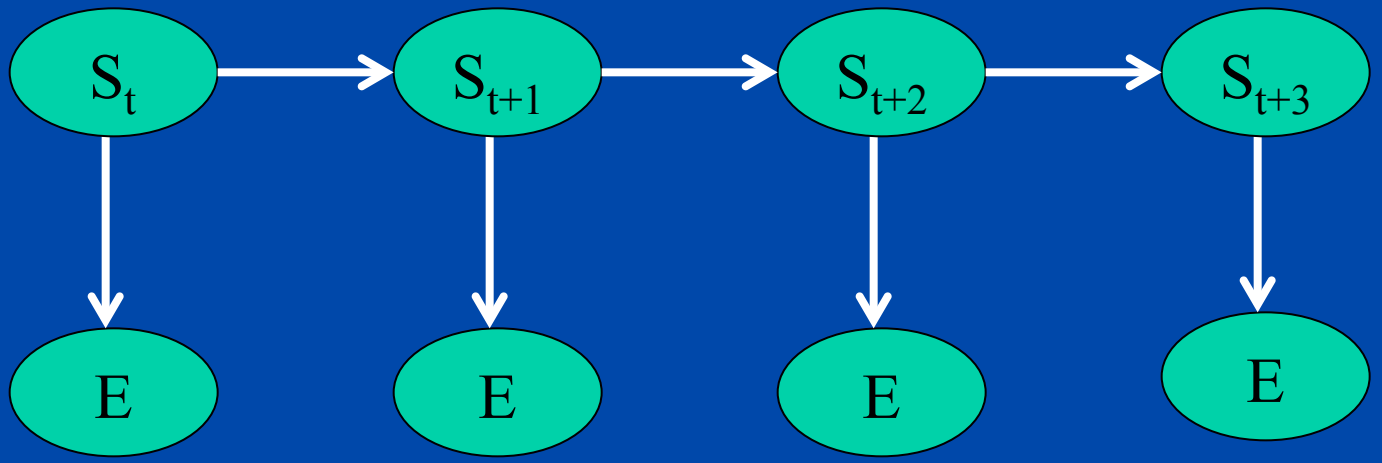


T T + 1

Modelo expandido (4 etapas temporales)



Ejemplo de RBD (equiv. HMM)



Parámetros:

- Probabilidades iniciales: $P(S_t)$
- Probabilidades de transición: $P(S_{t+1}|S_t)$
- Probabilidades de observación: $P(E|S_t)$

Suposiciones Básicas

- Proceso markoviano - el estado actual sólo depende del estado anterior (sólo hay arcos entre tiempos consecutivos)

- Proceso estacionario en el tiempo - las probabilidades de transición, $P(S_{t+1} \mid S_t)$, no cambian en el tiempo

Inferencia

- Existen 4 tipos de inferencia en RBD:
  - Filtrado: predecir el siguiente estado dadas observaciones pasadas - $P(X_t \mid Y_1 .. Y_{t-1})$
  - Predicción: predecir estados futuros basado en observaciones pasadas - $P(X_{t+n} \mid Y_1 .. Y_{t-1})$
  - Suavizamiento: estimar el estado actual dadas observaciones pasadas y futuras - $P(X_t \mid Y_1 .. Y_T)$
  - Decodificación: encontrar la secuencia más probable de variables ocultas dada las observaciones - $P(\mathbf{X} \mid Y_1 .. Y_T)$

Inferencia

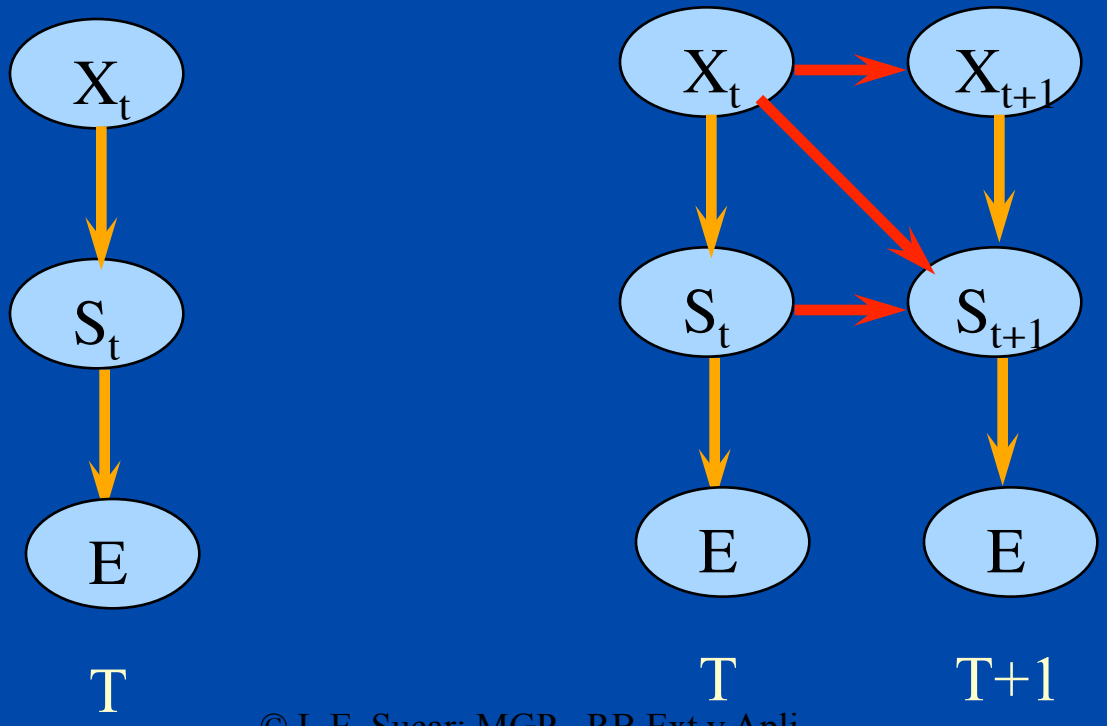
- En general el problema de inferencia es más complejo en RBD
- Existen soluciones eficientes para ciertos casos particulares, como los HMM
- En otros modelos más complejos se utilizan técnicas aproximadas basadas en simulación estocástica, como los *filtros de partículas*

Aprendizaje

- El aprendizaje de RBD es similar al de las redes bayesianas “estáticas”, pero se puede dividir en dos fases:
  - Aprender la red “base”
  - Aprender la red de transición
- Cada fase a su vez implica el aprendizaje de la estructura y de los parámetros
- También se pueden presentar los casos de información incompleta donde se aplican las técnicas vistas para redes bayesianas (EM)

Aprendizaje Estructural

- Aprender la estructura “estática”
- Aprender la estructura de “transición”



Redes Temporales

• Representaciones alternativas a RBD que incorporan aspectos temporales

• Se orientan a representar intervalos de tiempo o eventos en el tiempo vs estados

• Existen diferentes propuestas, dos ejemplos representativos son:

- Redes de tiempo (*time net*) [Kanazawa]
- Redes de nodos temporales (TNBN) [Arroyo]

Redes de Nodos Temporales

- Representan cambios de estado (eventos) de las variables

- Tienen dos tipos de nodos:

  - Nodos de estado - representan variables de estado como en las RB

  - Nodos temporales - representan cambios de estado de una variable

Nodo Temporal

- Nodo que representa un "evento" o cambio de estado de una variable de estado

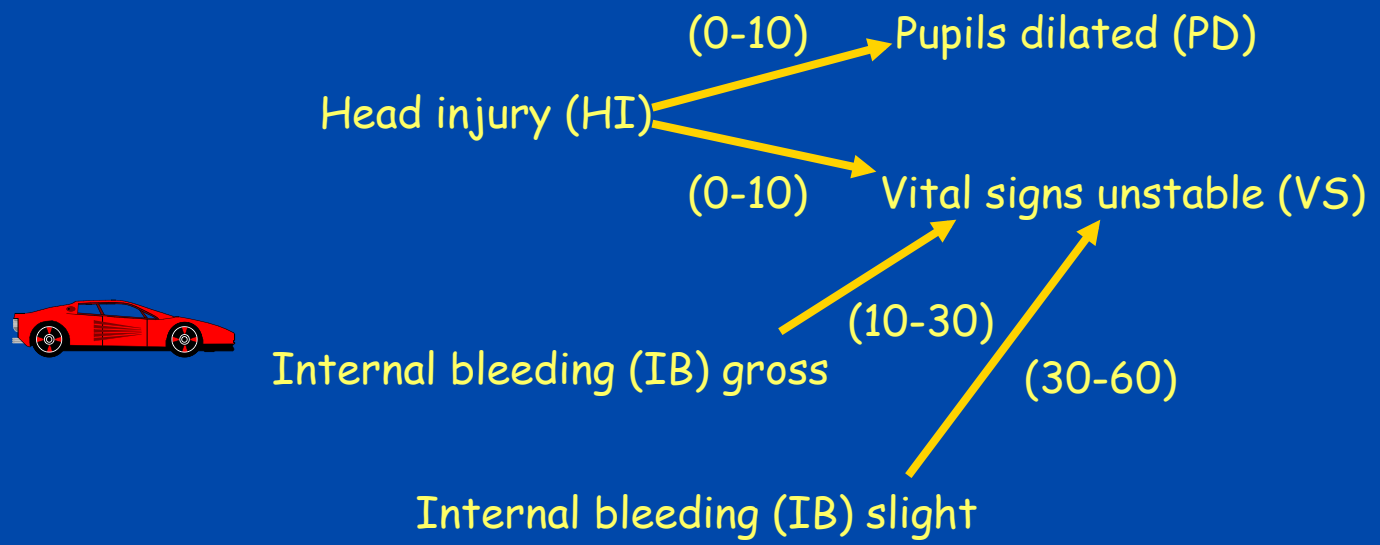
- Sus valores corresponden a diferentes intervalos de tiempo en que ocurre el cambio

- Ejemplo: *incremento de nivel*
  - Valores (3): * Cambio 0 - 1 0
    * Cambio 10 - 50
    * No Cambio

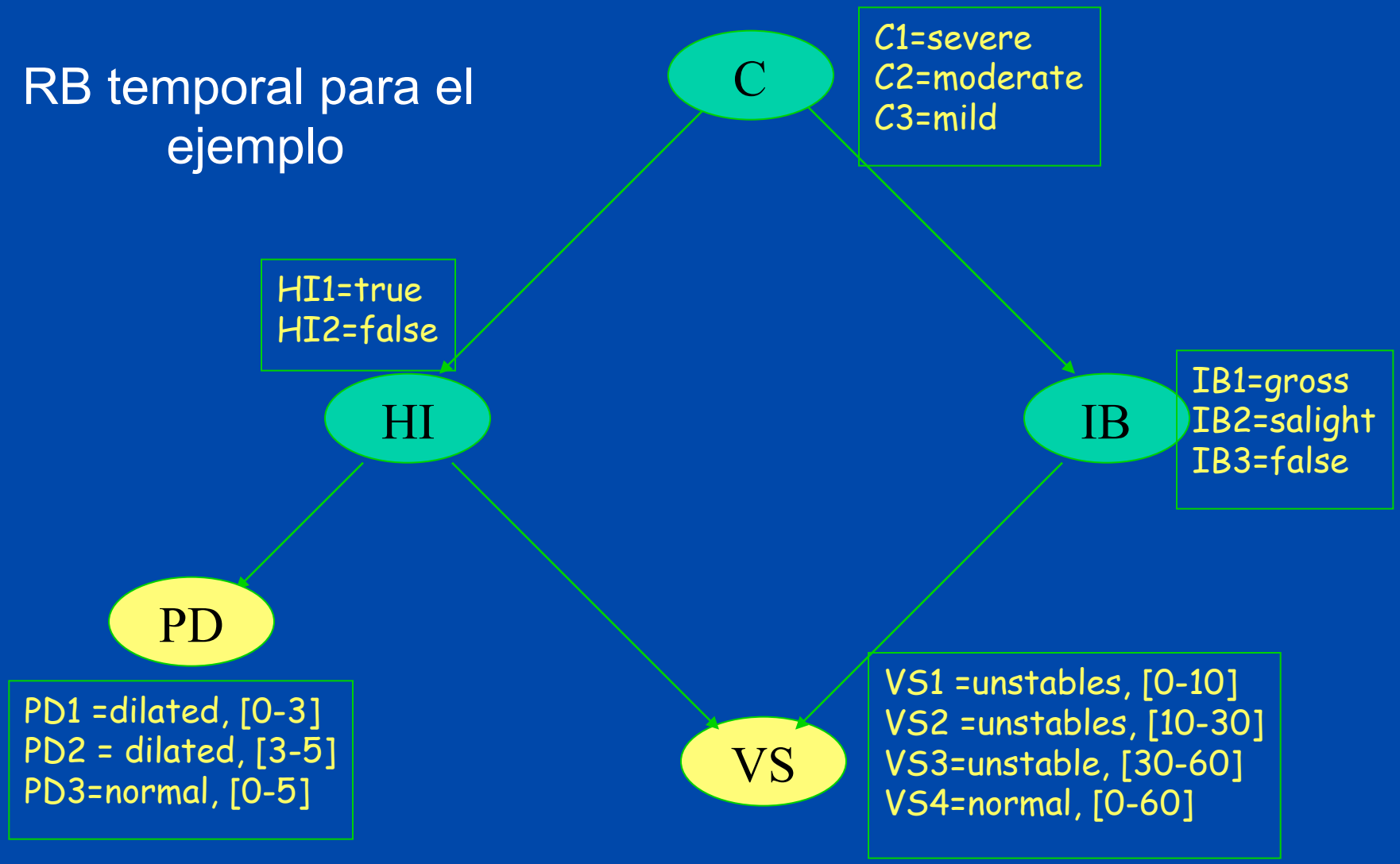
Redes con Nodos Temporales

- Permiten una representación más compacta de ciertos dominios que las redes dinámicas

- Ejemplo (accidente automovilístico):



RB temporal para el ejemplo



TNBN

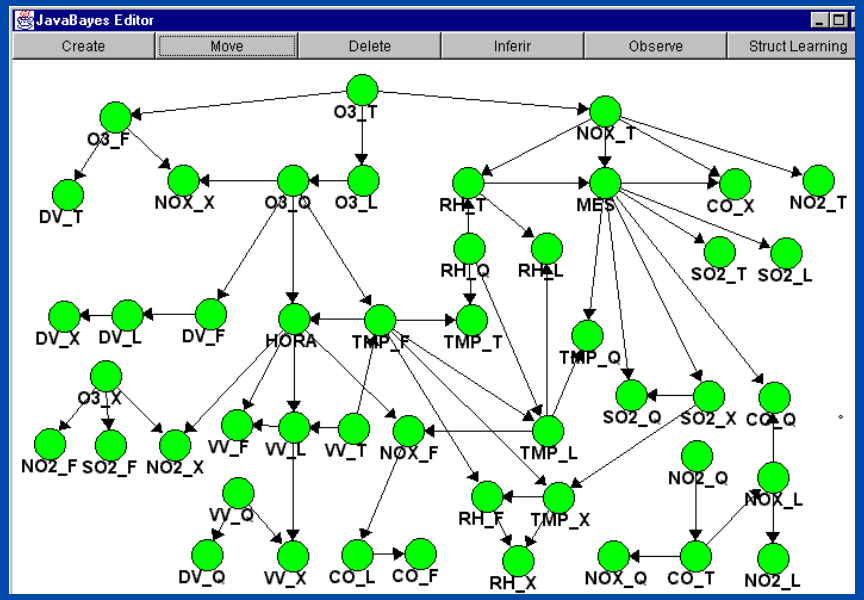
- Para cada nodo temporal se definen un conjunto de valores que corresponden a intervalos de tiempo y las probabilidades asociadas

- La propagación se hace de la misma manera que en redes estáticas

Aprendizaje de TNBN

- El aprendizaje de TNBN implica 3 aspectos:
  - Aprender los intervalos temporales (similar a un proceso de discretización)
  - Aprender la estructura
  - Aprender los parámetros

7: Aplicaciones

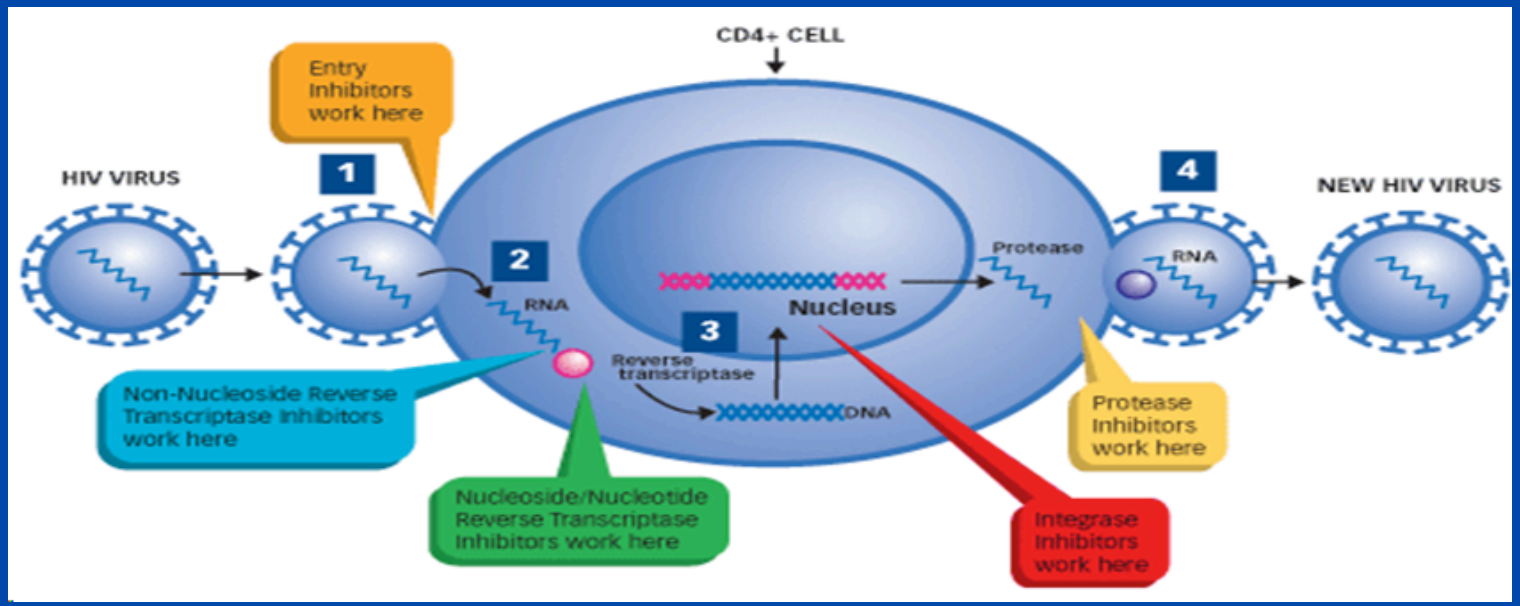


Aplicaciones

- Modelado de la evolución del VIH
  - Red temporal
- Endoscopía
  - RB (mejora estructural)
- Reconocimiento de gestos
  - RBD (clasificador bayesiano dinámico)
- Predicción de ozono
  - RB (aprendizaje estructural)

VIH

- El VIH es uno de los organismos que más rápidamente evoluciona, entre otras causas como respuesta a los fármacos
- ¿Es posible modelar la evolución del virus para predecir las mutaciones futuras y poder diseñar una mejor estrategia de control?



Datos

- A partir de datos históricos de pacientes se aprende un modelo de la evolución del virus basado en TNBN
- BD de U. Stanford, más de 2000 pacientes

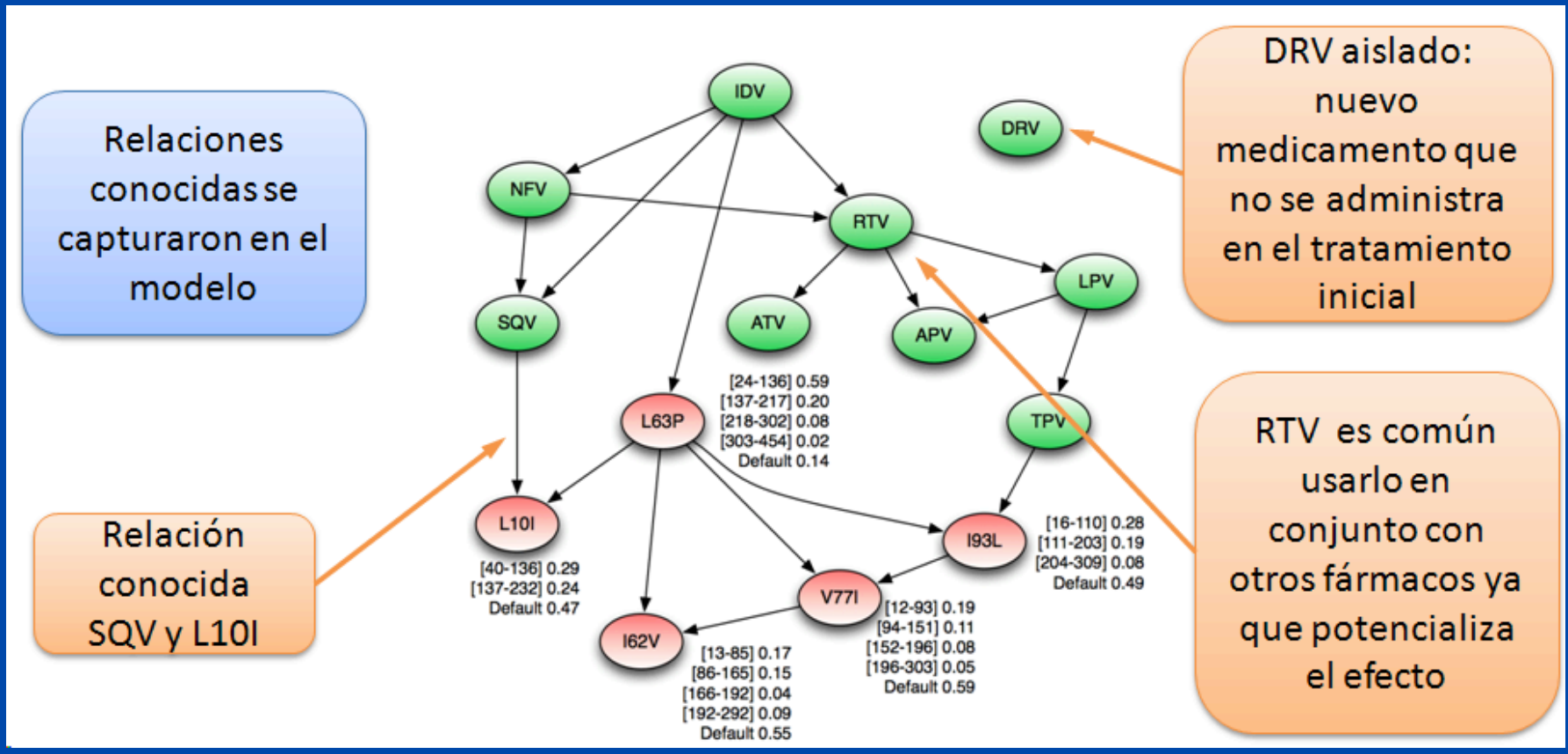
Paciente	Tratamiento	Lista de Mutaciones	Semanas
P1	LPV, FPV, RTV	L63P, L10I, V77I, I62V	15 30 10
P2	NFV, RTV, SQV	L10I V77I	25 45

Alma Ríos-Flores (INAOE) 12

Aprendizaje

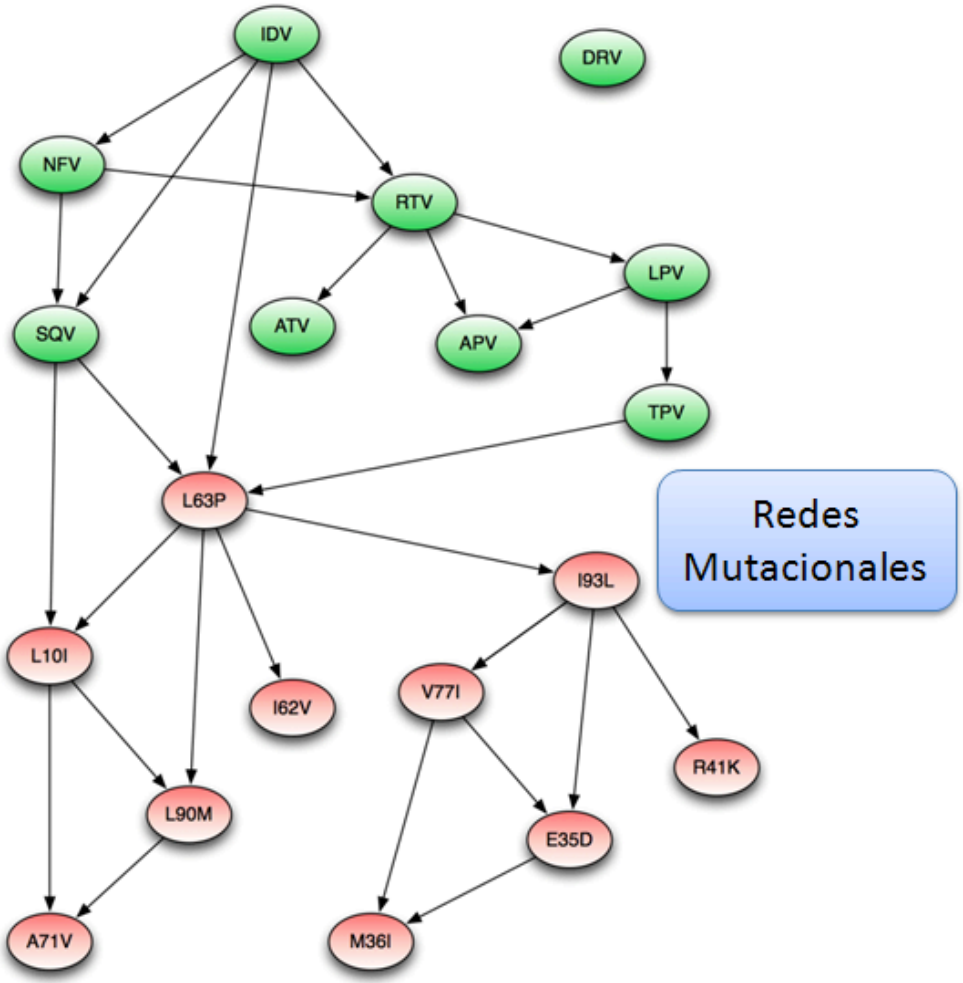
- A partir de los datos se aprende una TBNB incluyendo la estructura, los intervalos temporales y los parámetros
- Se construyeron dos modelos considerando los medicamentos y mutaciones más probables

Modelo 1


Relaciones conocidas se capturaron en el modelo
Relación conocida SQV y L10I
DRV aislado: nuevo medicamento que no se administra en el tratamiento inicial
RTV es común usarlo en conjunto con otros fármacos ya que potencializa el efecto
IDV
NFV
RTV
SQV
ATV
APV
LPV
DRV
L63P
[24-136] 0.59
[137-217] 0.20
[218-302] 0.08
[303-454] 0.02
Default 0.14
TPV
L10I
[40-136] 0.29
[137-232] 0.24
Default 0.47
I93L
[16-110] 0.28
[111-203] 0.19
[204-309] 0.08
Default 0.49
V77I
[12-93] 0.19
[94-151] 0.11
[152-196] 0.08
[196-303] 0.05
Default 0.59
I62V
[13-85] 0.17
[86-165] 0.15
[166-192] 0.04
[192-292] 0.09
Default 0.55

Modelo 2

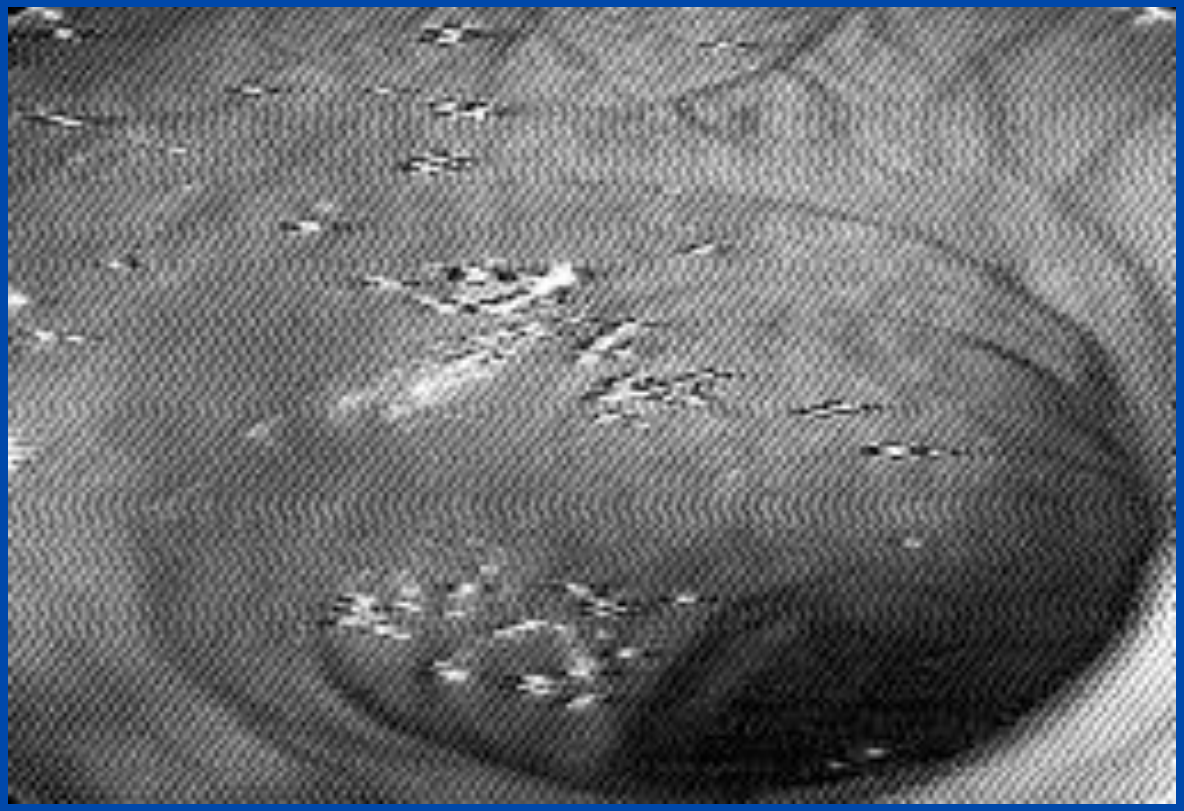
- Mutacion L63P
  - es tan frecuente que está sesgando el modelo
  - Tal vez es clave para desencadenar las demás mutaciones
- Dos redes han sido identificadas.



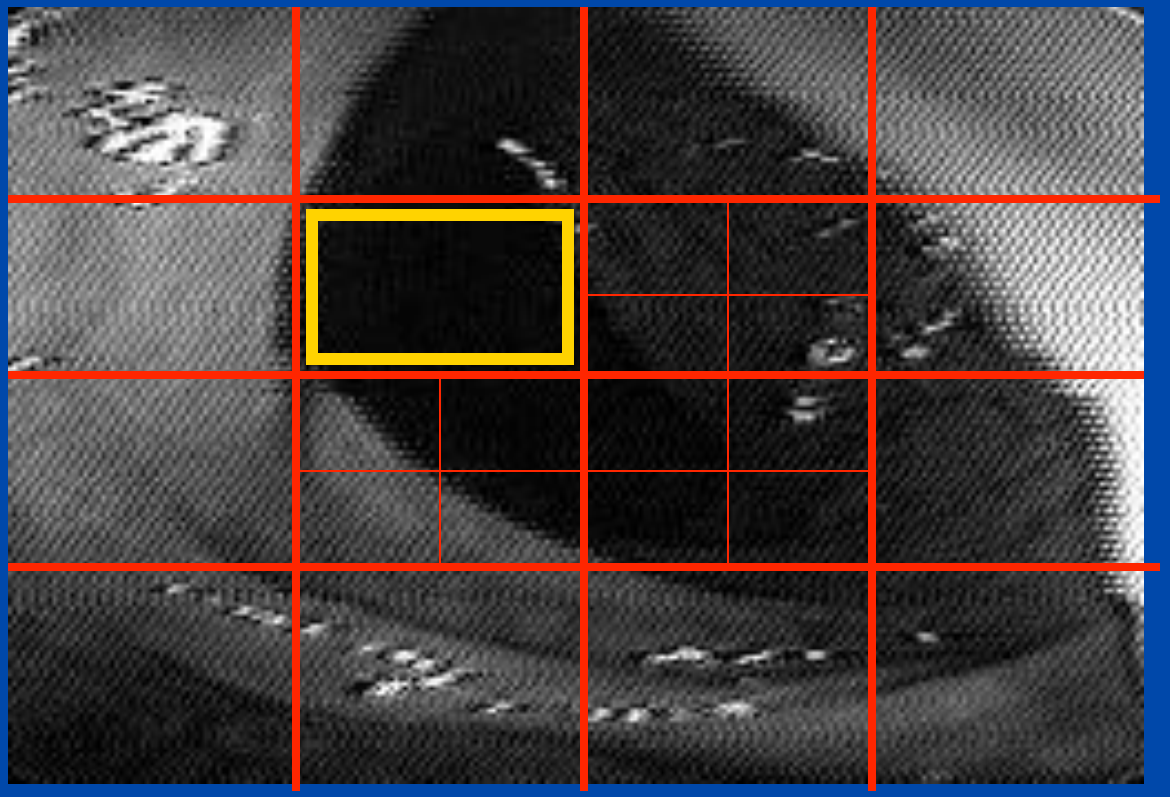
Endoscopía

- Herramienta para la observación directa del tubo digestivo
- El objetivo es ayudar al endoscopista a guiar el endoscopio en el colón (navegación semiautomática)
- Para ello se requieren reconocer los principales "objetos", como son:
  - *Lumen* – el "centro" del tubo
  - *Divertículos* – "bolsas" en las paredes que se pueden confundir con el colón

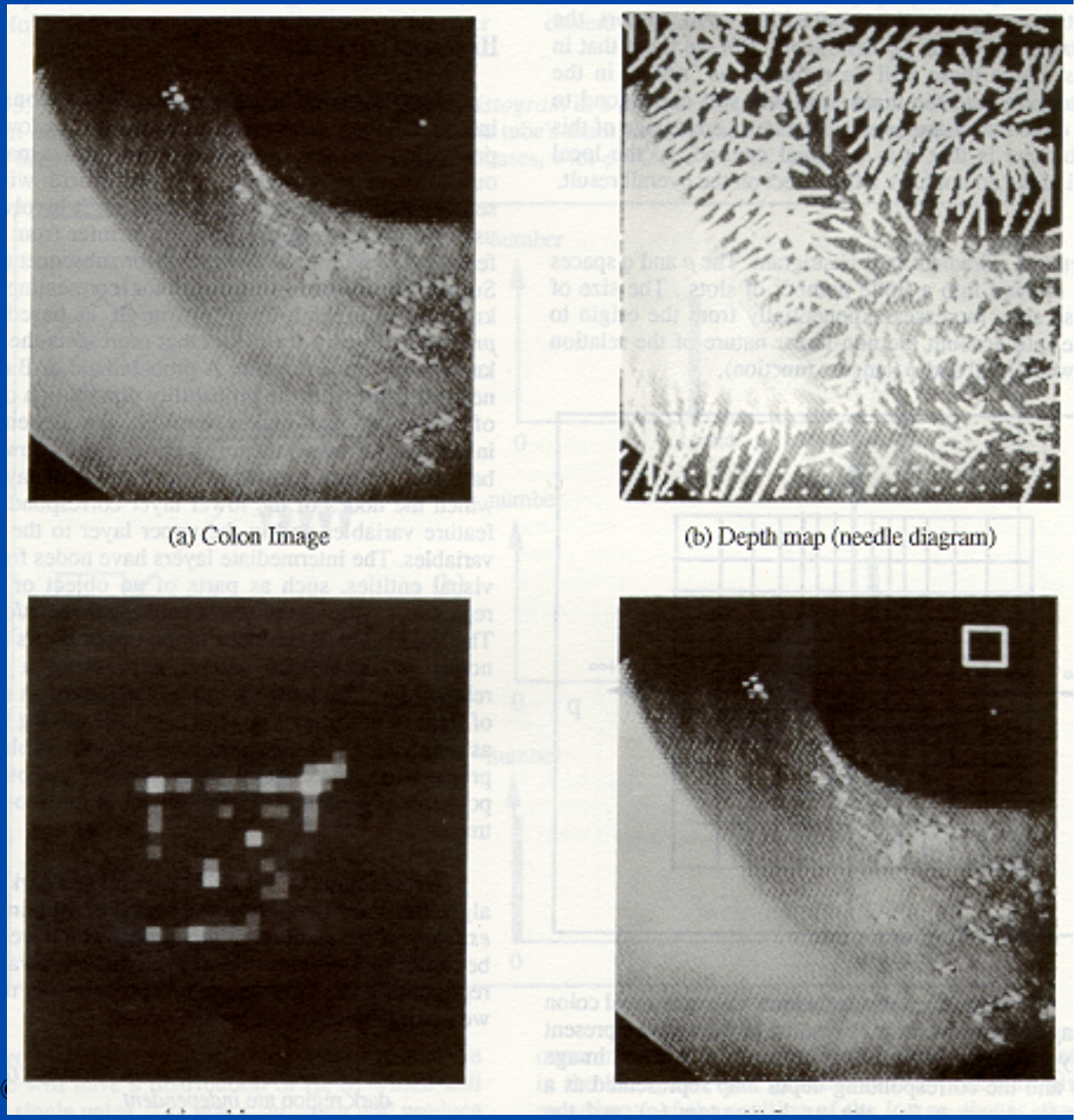
Ejemplo de imagen



Segmentatción – “región obscura”



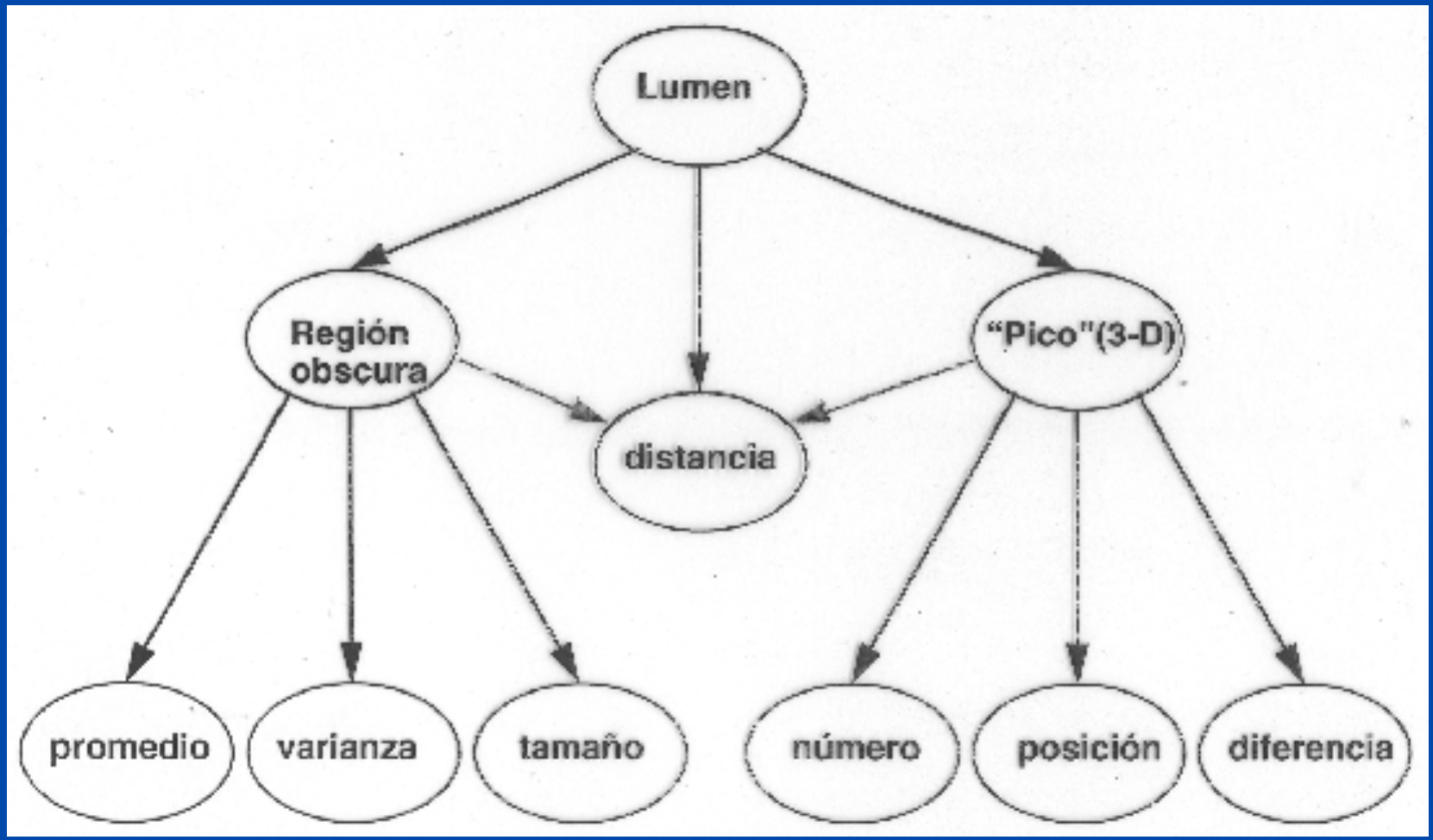
Información de 3D – *histograma pq*



(a) Colon Image

(b) Depth map (needle diagram)

RB para endoscopía (parcial)

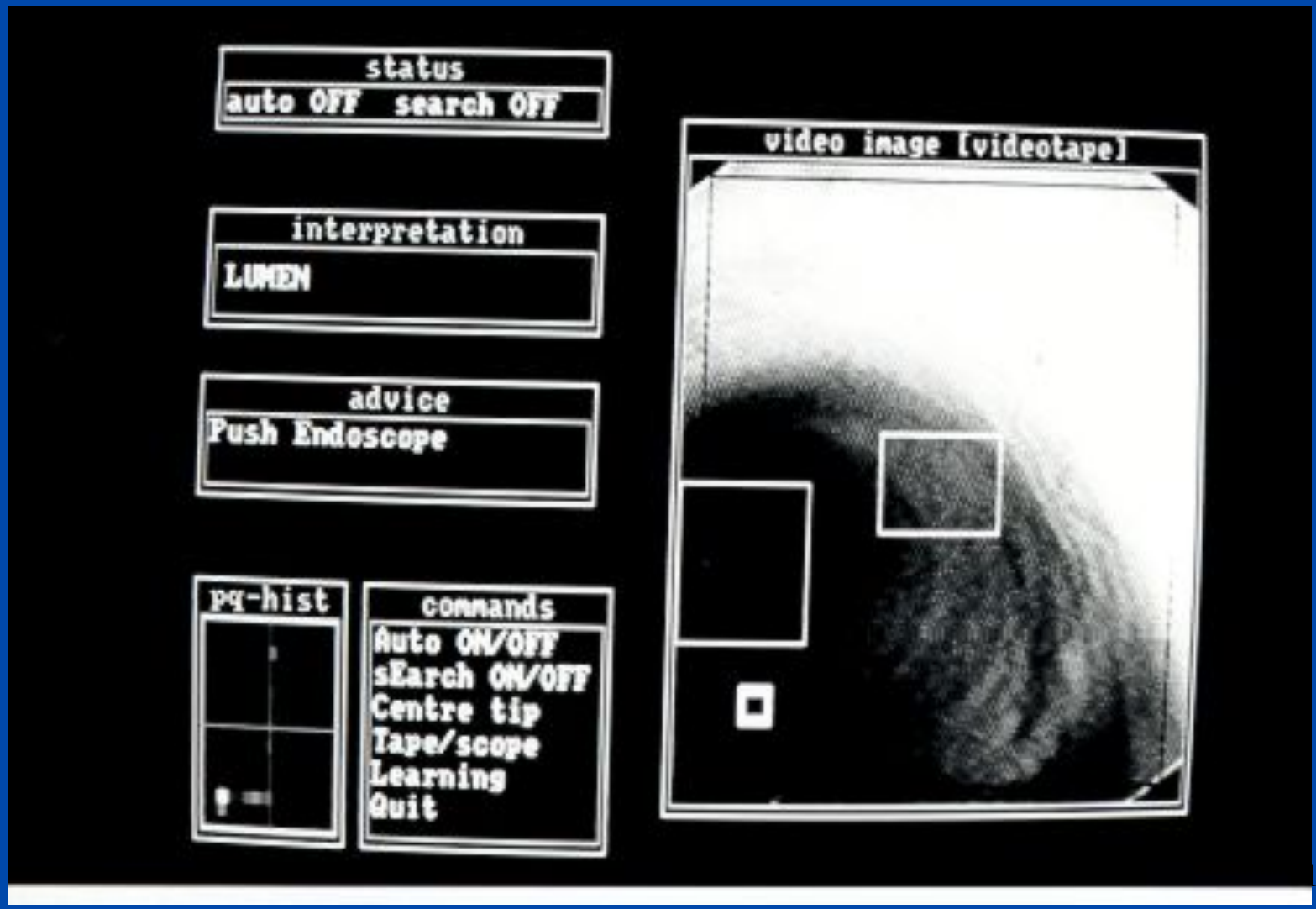


Metodología

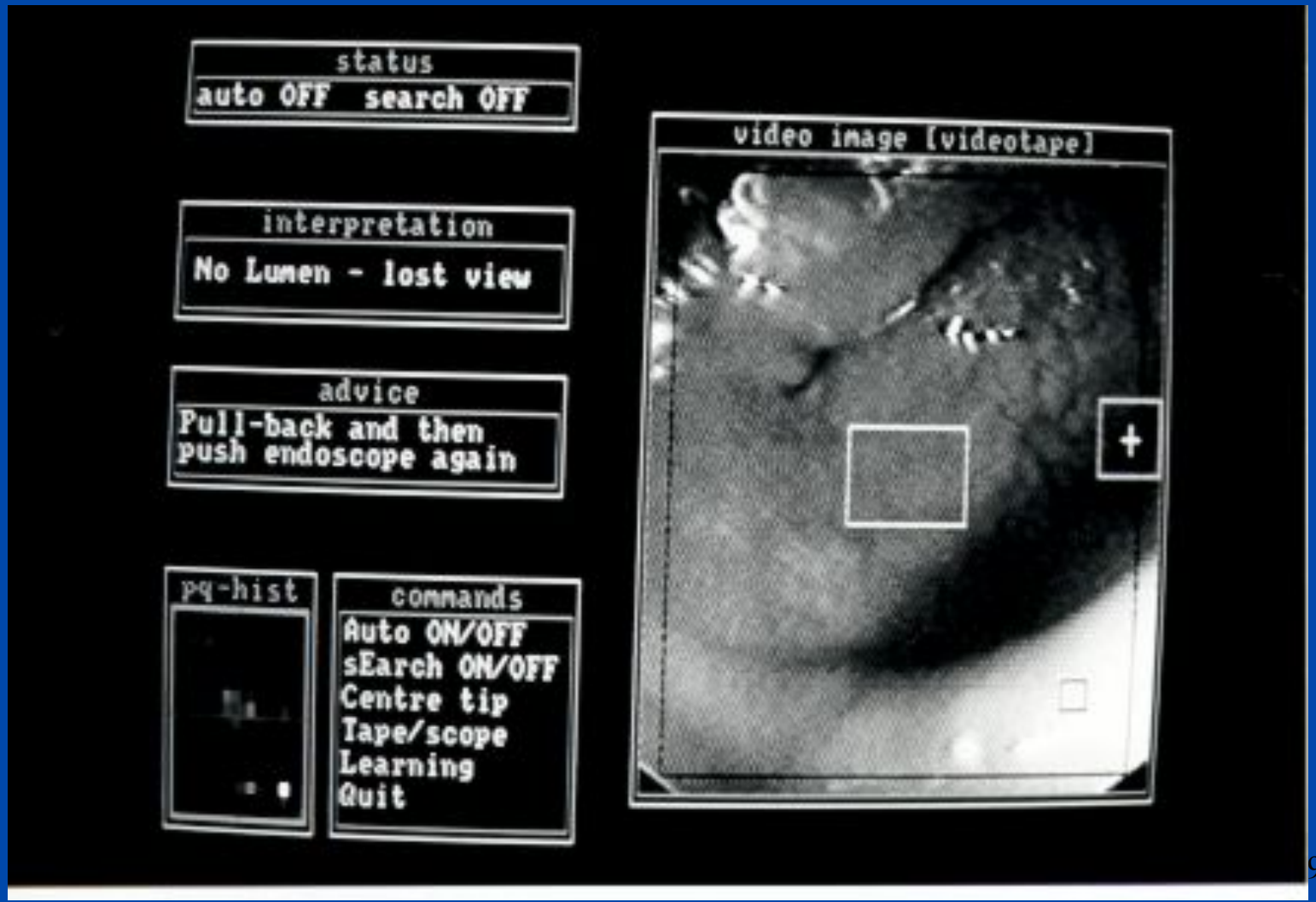
- La estructura de la RB se construyó con la ayuda de un experto
- Los parámetros de obtuvieron de datos (videos de endoscopías)
- Posteriormente se mejoró la estructura utilizando pruebas de independencia

Endoscopio semi-automático

Sistema de ayuda



Sistema de ayuda



Reconocimiento de gestos

- Reconocimiento de gestos orientados a comandar robots
- Inicialmente 5 gestos
- Reconocimiento con RBD

Come

attention

go-right

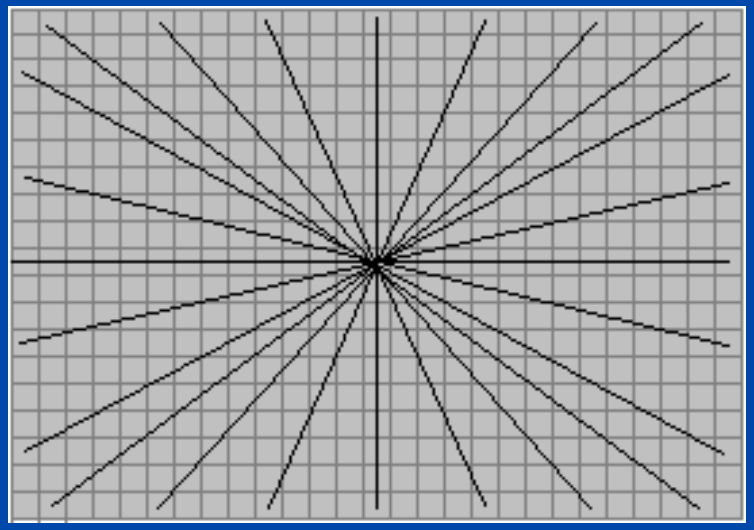
go-left

stop

Extracción de características

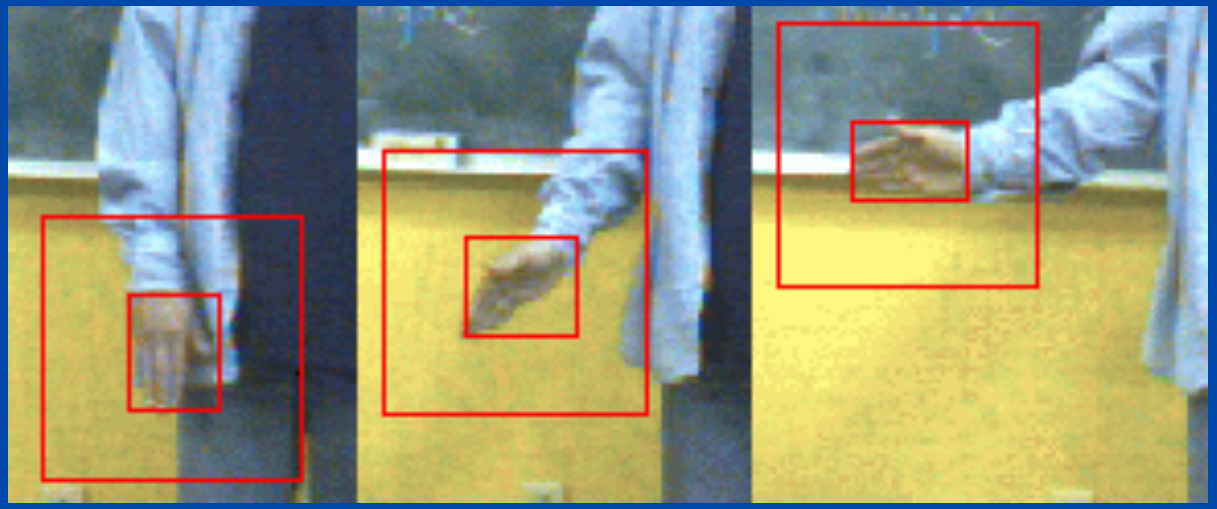
- Detección de piel
- Segmentación de cara y mano
- Seguimiento de la mano
- Características de movimiento

Segmentación

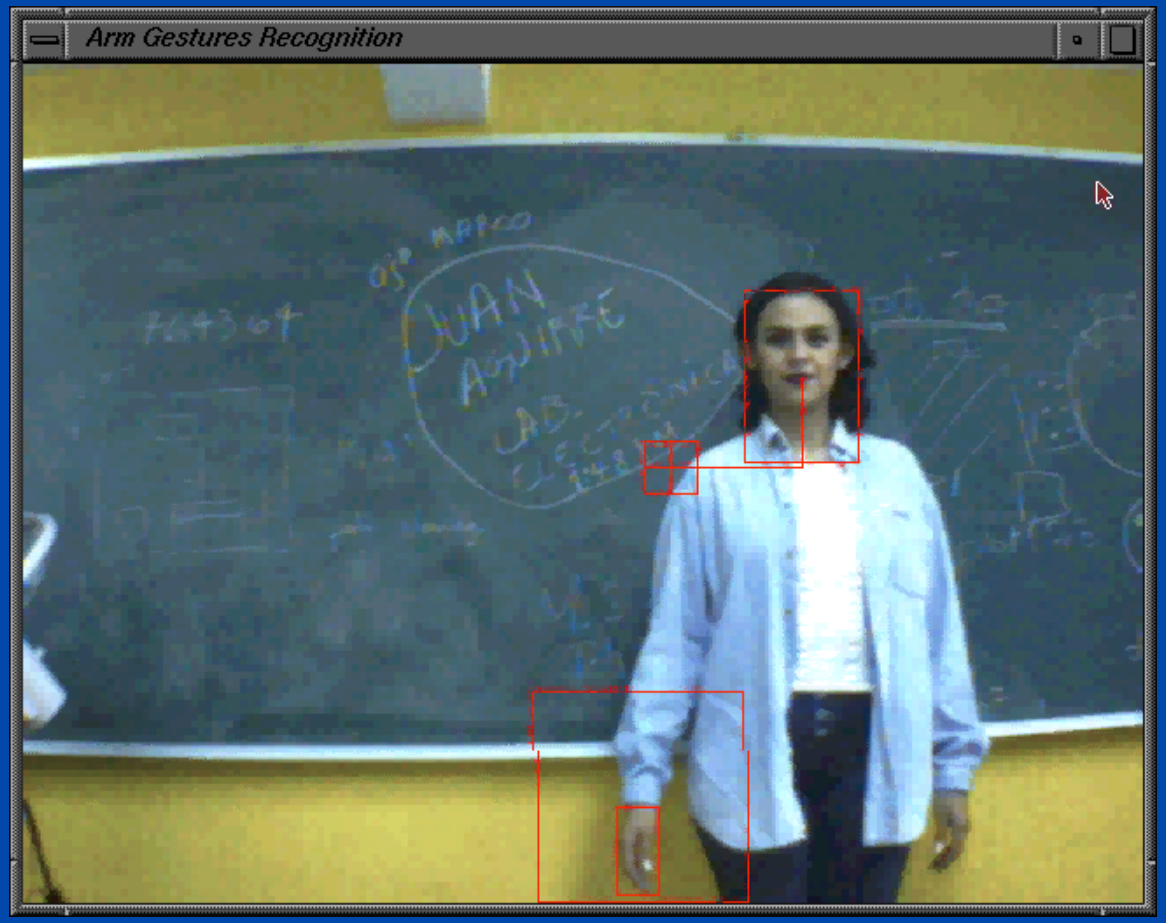


Agrupamiento de pixels de piel en muestreo radial

Seguimiento



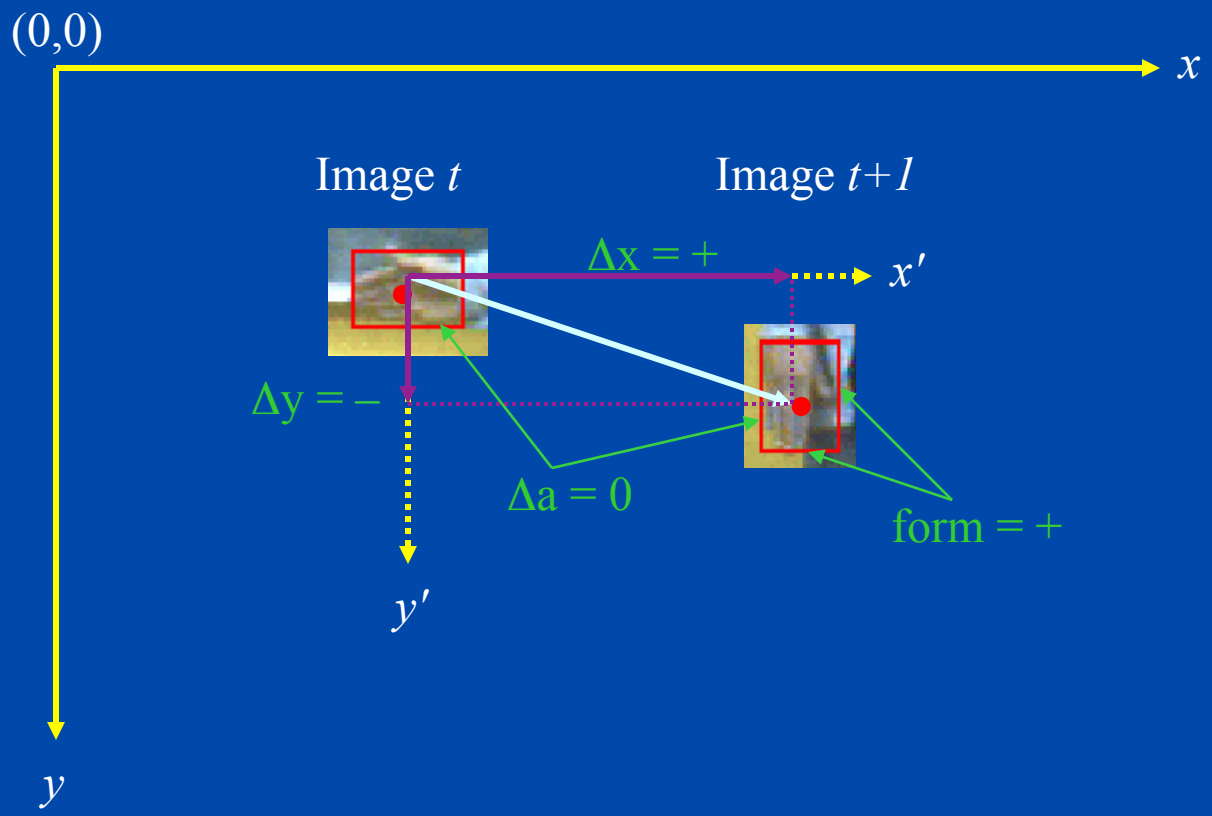
Seguimiento



Entrenamiento y Reconocimiento

- Los parámetros para la RBD se obtienen de ejemplos de cada gesto usando el algoritmo EM
- Para reconocimiento, se obtiene la probabilidad de cada modelo mediante propagación

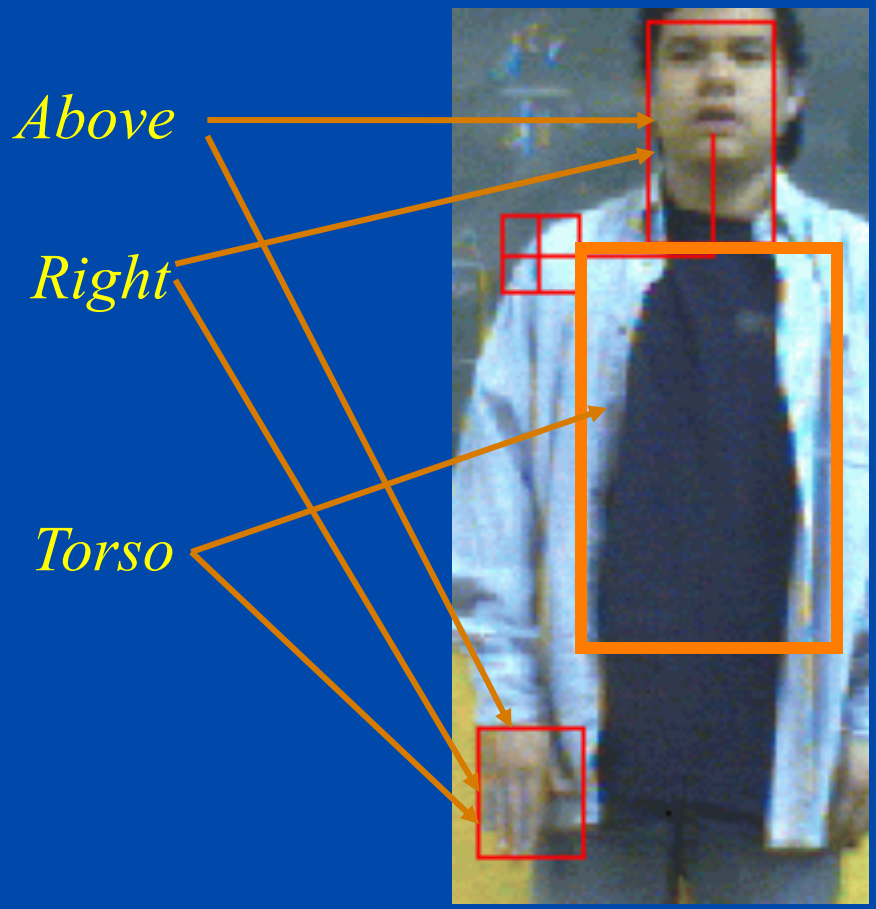
Atributos de Movimiento



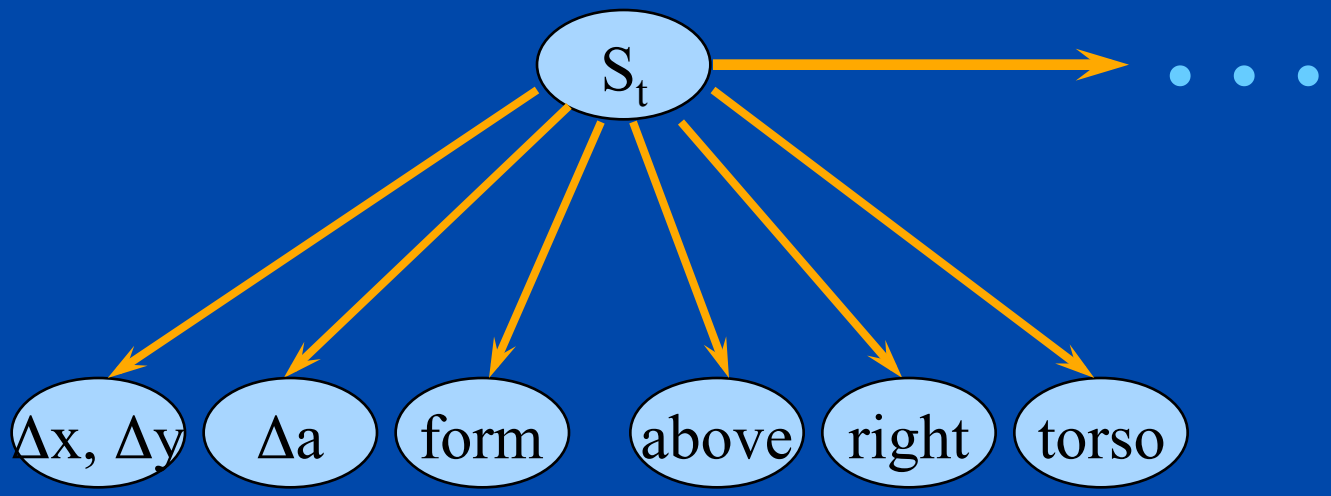
Atributos de postura

Relaciones espaciales (binarias) entre la mano y otras partes del cuerpo:

- *Derecha*
- *Arriba*
- *Torso*



Clasificador bayesiano dinámico



Experimentos

- 150 muestras de cada gesto
- Ambiente de laboratorio con condiciones variables de iluminación
- Distancia entre 3.0 m y 5.0 m
- Se varía el número de muestras para entrenamiento

Resultados: DNBCs sin información de postura

	come	attention	go-right	go-left	stop
come	98 %			2 %	
attention	3 %	87 %	10 %		
go-right			100 %		
go-left				100 %	
stop	4 %	39 %		1 %	56 %

Promedio 87.75 %

© L.E. Sucar: MGP - RB Ext y Apli

Resultados: DNBCs con información de postura

	come	attention	go-right	go-left	stop
come	100 %				
attention		100 %			
go-right			100 %		
go-left				100 %	
stop	11 %	6 %			83 %

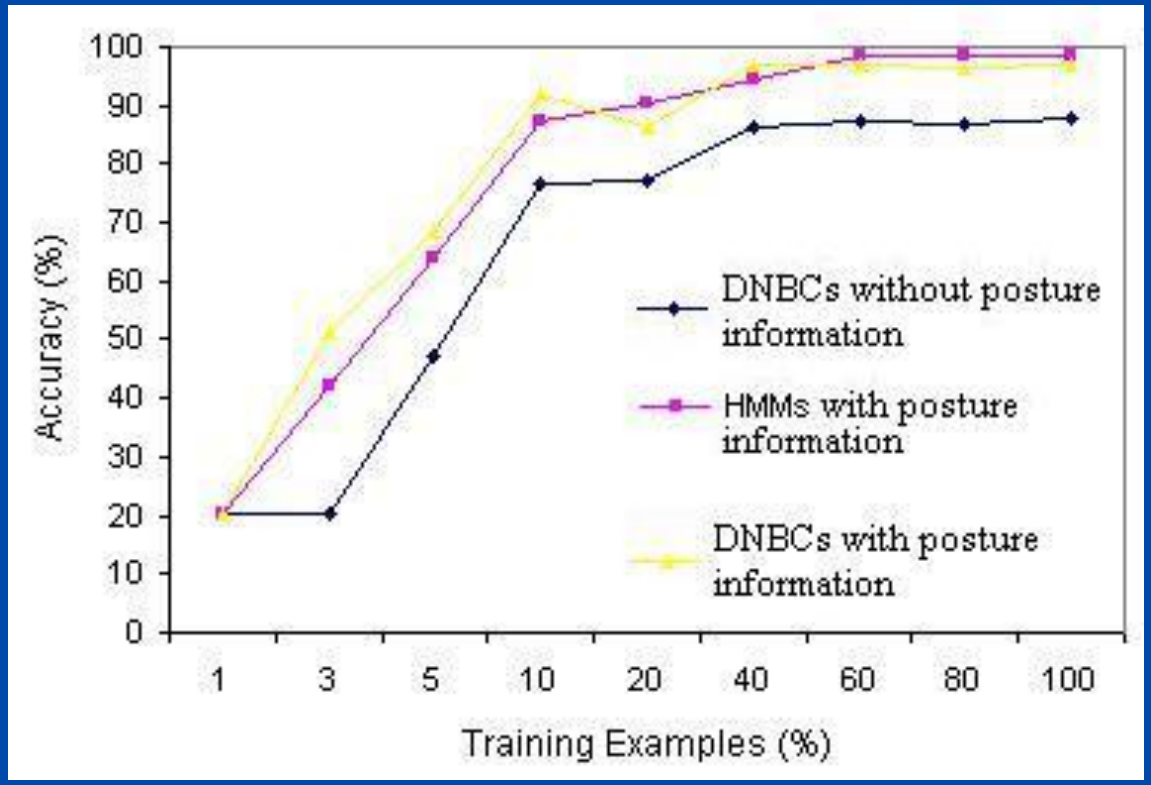
Promedio 96.75 %

Resultados: HMMs con información de postura

	come	attention	go-right	go-left	stop
come	100 %				
attention		100 %			
go-right			100 %		
go-left				100 %	
stop	8 %				92 %

Promedio 98.47 %

Exactitud vs tamaño de la muestra



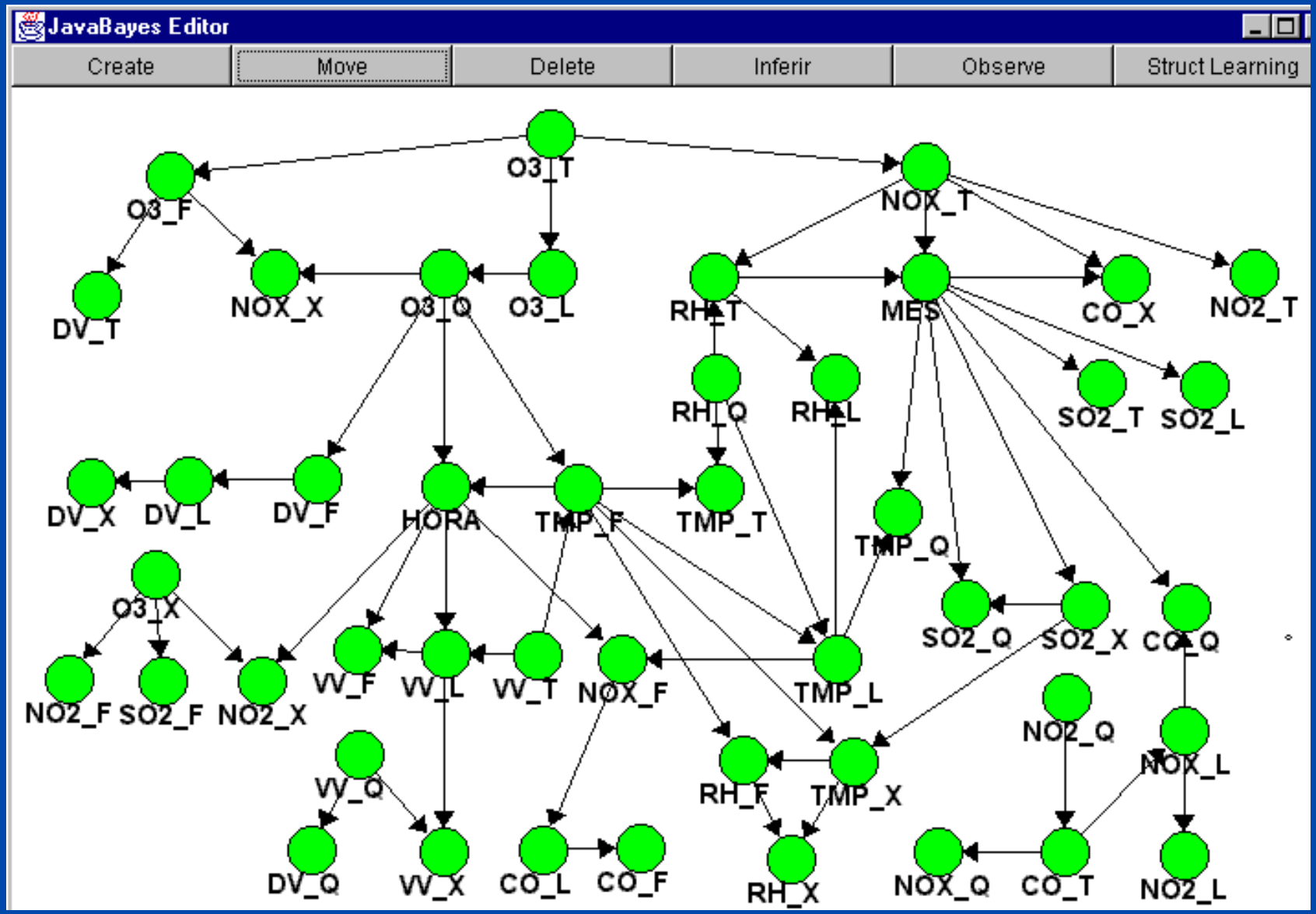
Predicción de Ozono en la Ciudad de México

- Construcción de un modelo basado en redes bayesianas para predecir el Ozono
- 47 variables:
  - 9 mediciones en 5 estaciones
  - Hora del día
  - Mes del año

Experimentos

- Aprendizaje estructural:
  - 400 casos
  - Orden causal
- Predicción:
  - Nivel de ozono en una estación una hora después
  - Diferentes porcentajes de variables conocidas

Modelo


JavaBayes Editor
Create
Move
Delete
Inferir
Observe
Struct Learning
O3_T
O3_F
NOX_T
DV_T
NOX_X
O3_Q
O3_L
RH_T
MES
CO_X
NO2_T
RH_Q
RH_L
SO2_T
SO2_L
DV_X
DV_L
DV_F
HORA
TMP_F
TMP_T
TMP_Q
O3_X
SO2_Q
SO2_X
CO_Q
NO2_F
SO2_F
NO2_X
VV_F
VV_L
VV_T
NOX_F
TMP_L
NO2_Q
NOX_L
VV_Q
RH_F
TMP_X
DV_Q
VV_X
CO_L
CO_F
RH_X
NOX_Q
CO_T
NO2_L

Comparación con una red neuronal

Predicción de ozono: BN vs HNN

% Datos	Método	Error relativo	Error de inter.
100%	BN	21%	12%
	HNN	26%	12%
85%	BN	21%	12%
	HNN	26%	9%
70%	BN	25%	8%
	HNN	32%	12%
50%	BN	21%	12%
	HNN	37%	29%

Otras aplicaciones

- Predicción del precio del petróleo
- Modelado de riesgo en accidentes de automóviles
- Diagnóstico médico
- Validación de sensores
- Modelado de usuarios (ayudantes Microsoft Office)
- Modelado del estudiante (tutores inteligentes)
- Diagnóstico de turbinas (General Electric)
- Reconocimiento de objetos en imágenes
- Reconocimiento de voz
- ...

Conclusiones

- El enfoque bayesiano permite combinar el conocimiento previo con los datos para tomar las *mejores decisiones* bajo incertidumbre
- Los modelos gráficos probabilistas aprovechan las relaciones de independencia para lograr una implementación computacionalmente eficiente del enfoque bayesiano
- Bajo el paradigma de los MGPs hay una variedad de técnicas que permiten modelar diversos problemas que involucran incertidumbre, tiempo y decisiones / utilidades

Retos

- Aumentar la expresividad de los modelos gráficos probabilistas
  - Modelos gráficos relacionales
- Descubrir relaciones “causales”
  - Modelos gráficos causales

Modelos Relacionales Probabilistas

Combinan lógica y probabilidad

Ejemplo – gramática visual para reconocimiento de objetos (caras):

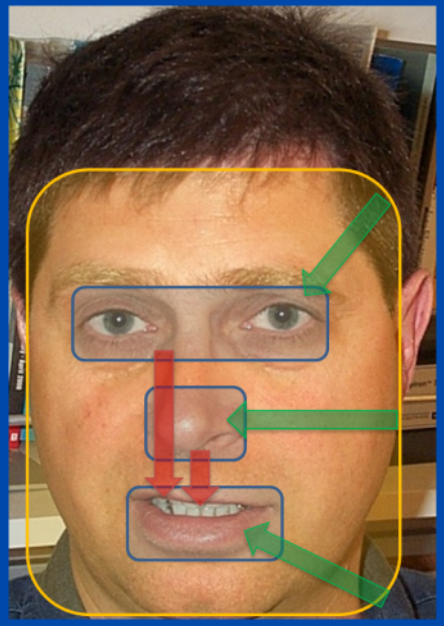
Above(Eyes, Mouth)

Above(Nose, Mouth)

Within(Eyes, Skin)

Within(Nose, Skin)

Within(Mouth, Skin)



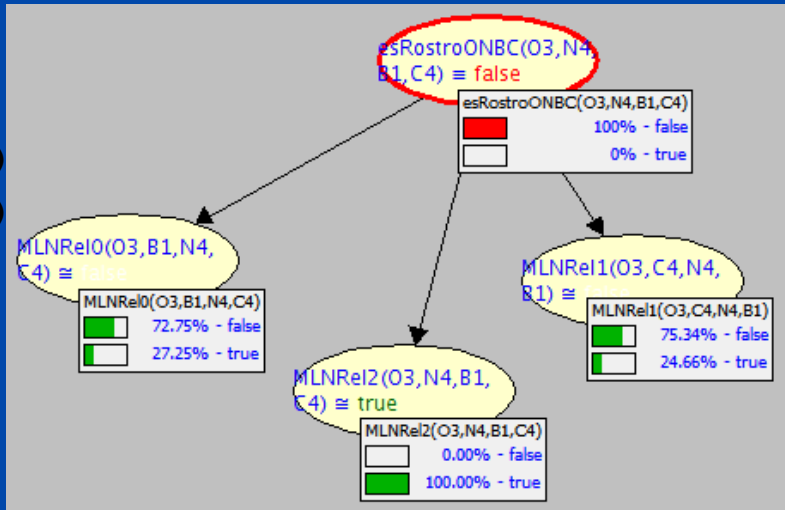
Reglas lógicas con dichas relaciones:

2.3 $\text{Above}_{EM}(e,m)$ v $\text{Above}_{NM}(n,m)$ => Rostro(e,n,m,s)

1.8 $\text{Within}_{ES}(e,s)$ v $\text{Within}_{NS}(n,s)$ => Rostro(e,n,m,s)

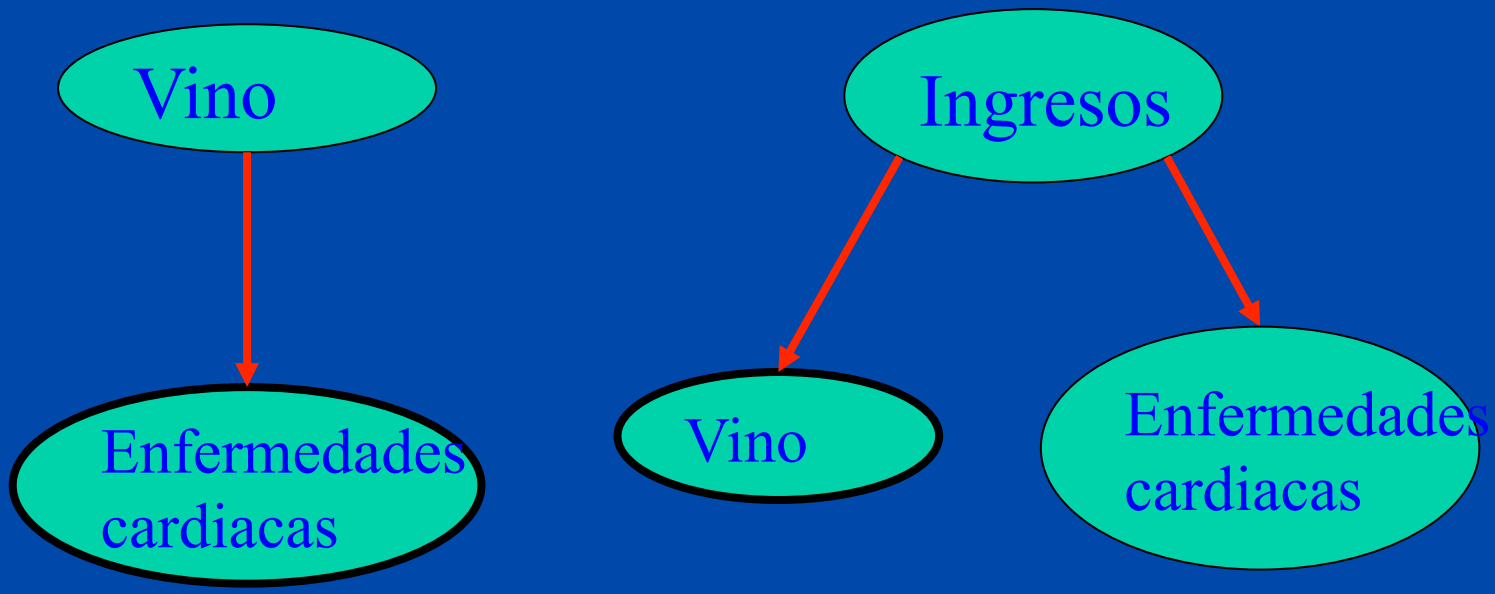
0.9 Rostro(e,n,m,s) => $\text{Within}_{MS}(m,s)$

El **peso** indica una incertidumbre asociada a tales reglas



Modelos Gráficos Causales

- Buscar representar relaciones causales; y realizar inferencias basadas en estos modelos
- Reto: aprender estos modelos sin necesidad de hacer intervenciones



Información Adicional

- Página del curso en INAOE:

http://ccc.inaoep.mx/~esucar/Clases-pgm/pgm.html

- Libro:

L. E Sucar, "Probabilistic Graphical Models: Principles and Applications", Springer-Verlag (por publicarse)