Modelos Gráficos Probabilistas

L. Enrique Sucar
INAOE

Sesión 12:
Redes Bayesianas – Aprendizaje

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

Aprendizaje de Redes Bayesianas

- Introducción
- Aprendizaje paramétrico
  - Incertidumbre en las probabilidades
  - Datos faltantes / nodos ocultos
- Aprendizaje estructural
  - Árboles
  - Poliárboles
  - Redes multiconectadas
- Combinación de conocimiento y datos

Aprendizaje

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

En redes bayesianas se divide en 2 aspectos:

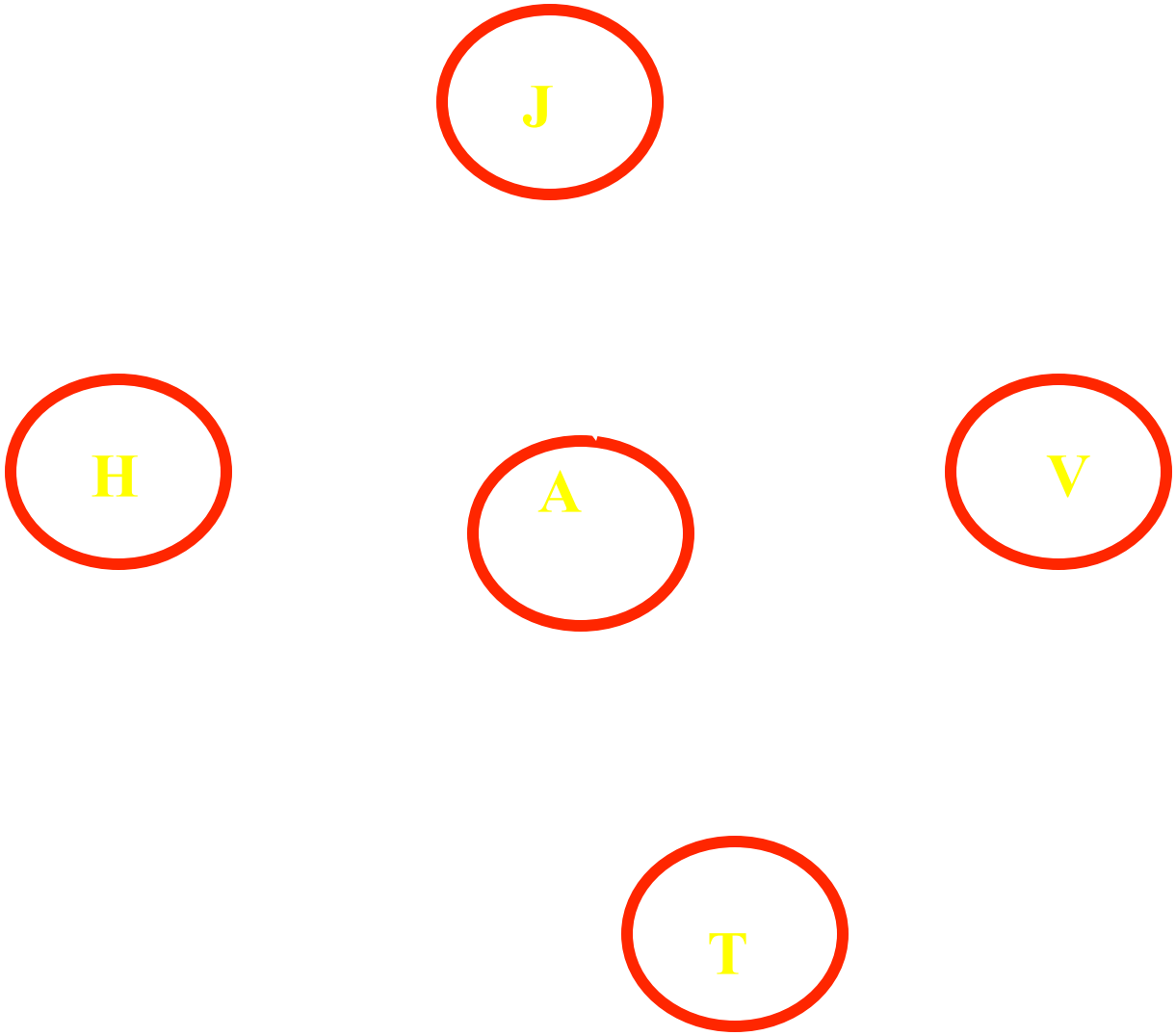
- Obtener la estructura de la red –
- Obtener las probabilidades asociadas –

Aprendizaje Paramétrico

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

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


J
H
A
V
T

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.

Suavizado

- Cuando se tienen pocos datos (o muchas variables-valores) se pueden tener probabilidades igual a cero, lo que ocasiona problemas
- Para ello se pueden "suavizar" las estimaciones de las probabilidades
- Existen varios métodos de suavizado, el más sencillo y común el de *Laplace*

Suavizado

- El suavizado de Laplace consiste en inicializar todas las probabilidades en forma uniforme, y después *incrementarlas* con los datos
- Ejemplo:

  – Inicial:
  – Dato
  – Dato
  – Dato

Incertidumbre en las probabilidades

- Normalmente hay incertidumbre en las probabilidades, ya sea estimadas de datos o por expertos
- Se puede representar mediante una distribución de probabilidad, ventajas:
  - Representación explícita
  - Combinar información de expertos con datos
  - Propagar la incertidumbre en las probabilidades

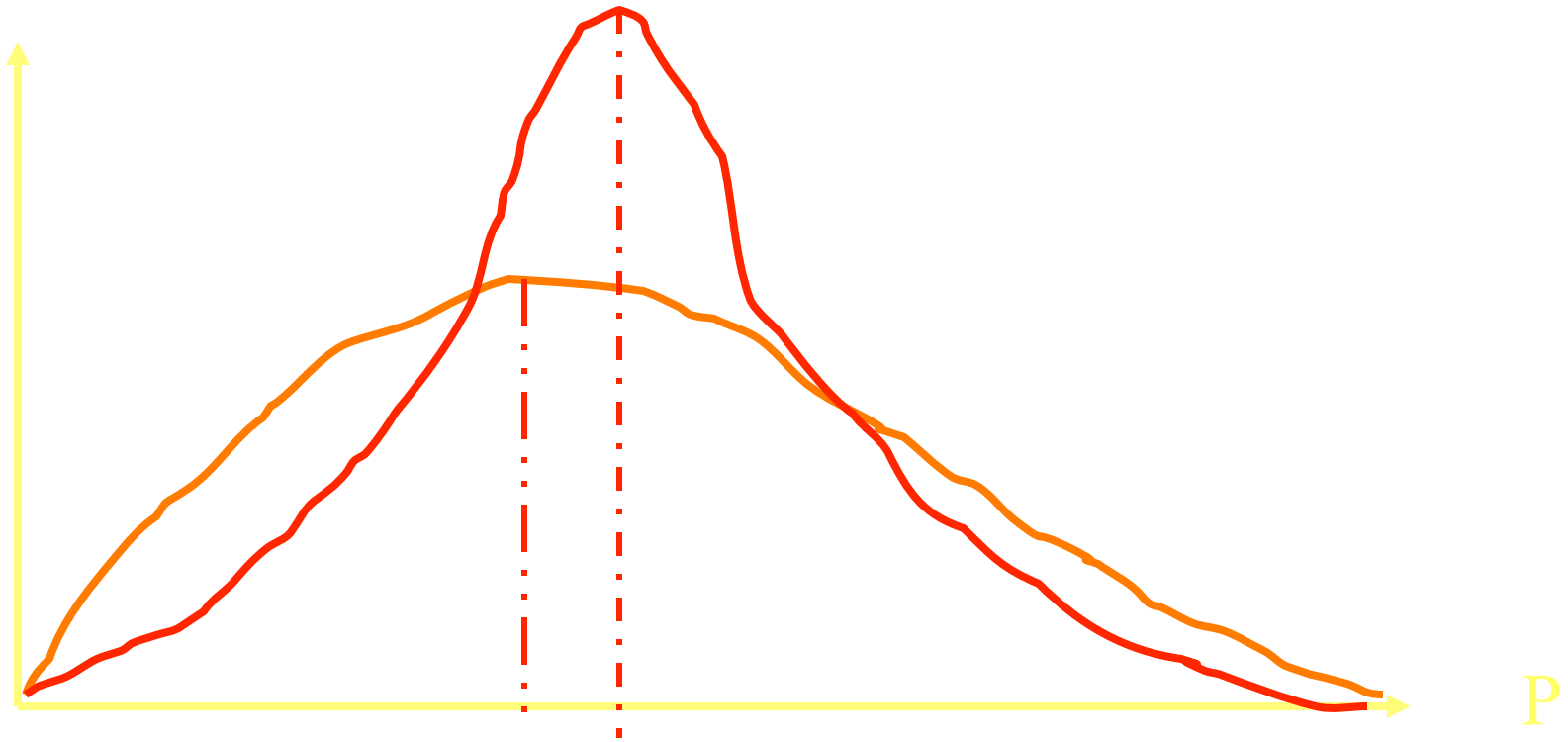
Incertidumbre en las probabilidades

- Variables binarias – distribución Beta

$$\beta(a,b) = \frac{(a+b+1)!}{a!b!} x^a (1-x)^b$$

- Valor promedio (esperado):

Distribución Beta



Incertidumbre en las probabilidades

- Modelación de estimación de expertos mediante valores de a y b:
  - Ignorancia completa: a=b=0
  - Poco confidente: a+b pequeño (10)
  - Medianamente confidente: a+b mediano (100)
  - Muy confidente: a+b grande (1000)
- Combinación de experto y datos:
  - P(x1) =
  - Datos:
  - Experto:

Incertidumbre en las probabilidades

- Variables multivaluadas – se utiliza la generalización de la Beta → distribución *Dirichlet*

$$\mathrm{Dir}(a_1, a_2, \ldots a_n) = \frac{(b_i + a_i + t - 1)!}{a_i!(b_i + t - 2)!} x^{a_i} (1 - x)^{(b_i + t - 1)}$$

  - Donde:
- Valor esperado:

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 –
  - Faltan todos los valores de una variable –

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

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:

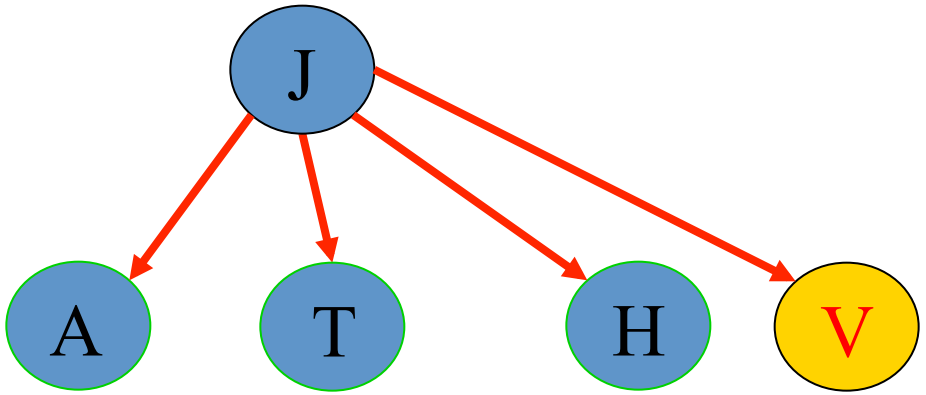
  : se estiman los datos faltantes en base a los parámetros (P) actuales

  : 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

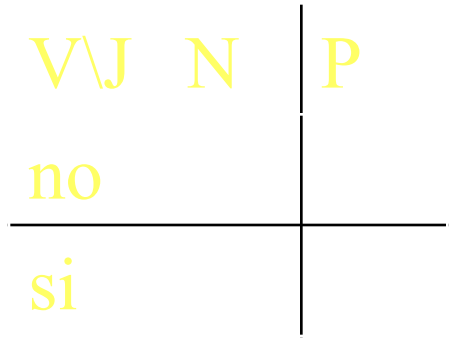
Ejemplo



- 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

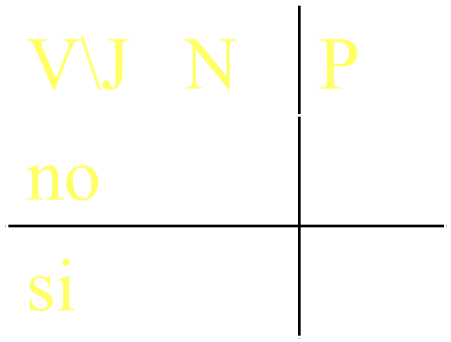


EM: paso E

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

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



EM

- Limitaciones:
  - Puede caer en máximos locales (depende del valor inicial)
  - Complejidad computacional

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 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 es la causa o padre de .

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 ( ) y la aproximada ( ):

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

- El objetivo es minimizar .

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).

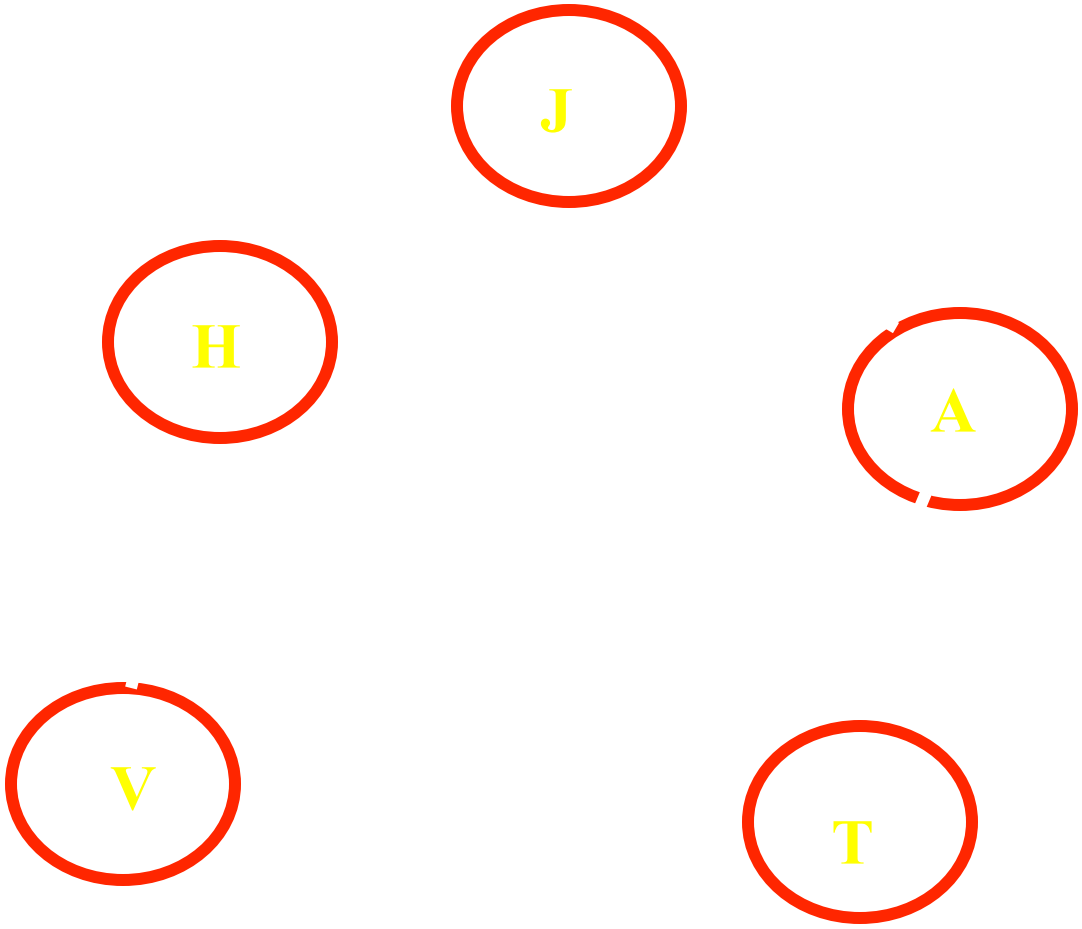
- El algoritmo NO provee la dirección de los arcos, por lo que ésta se puede asignar en forma arbitraria o utilizando semántica externa (experto).

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.	…

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:
  -

- Los primeros dos casos son indistinguibles, pero el tercero es diferente, ya que las dos variables "padre" son marginalmente independientes.

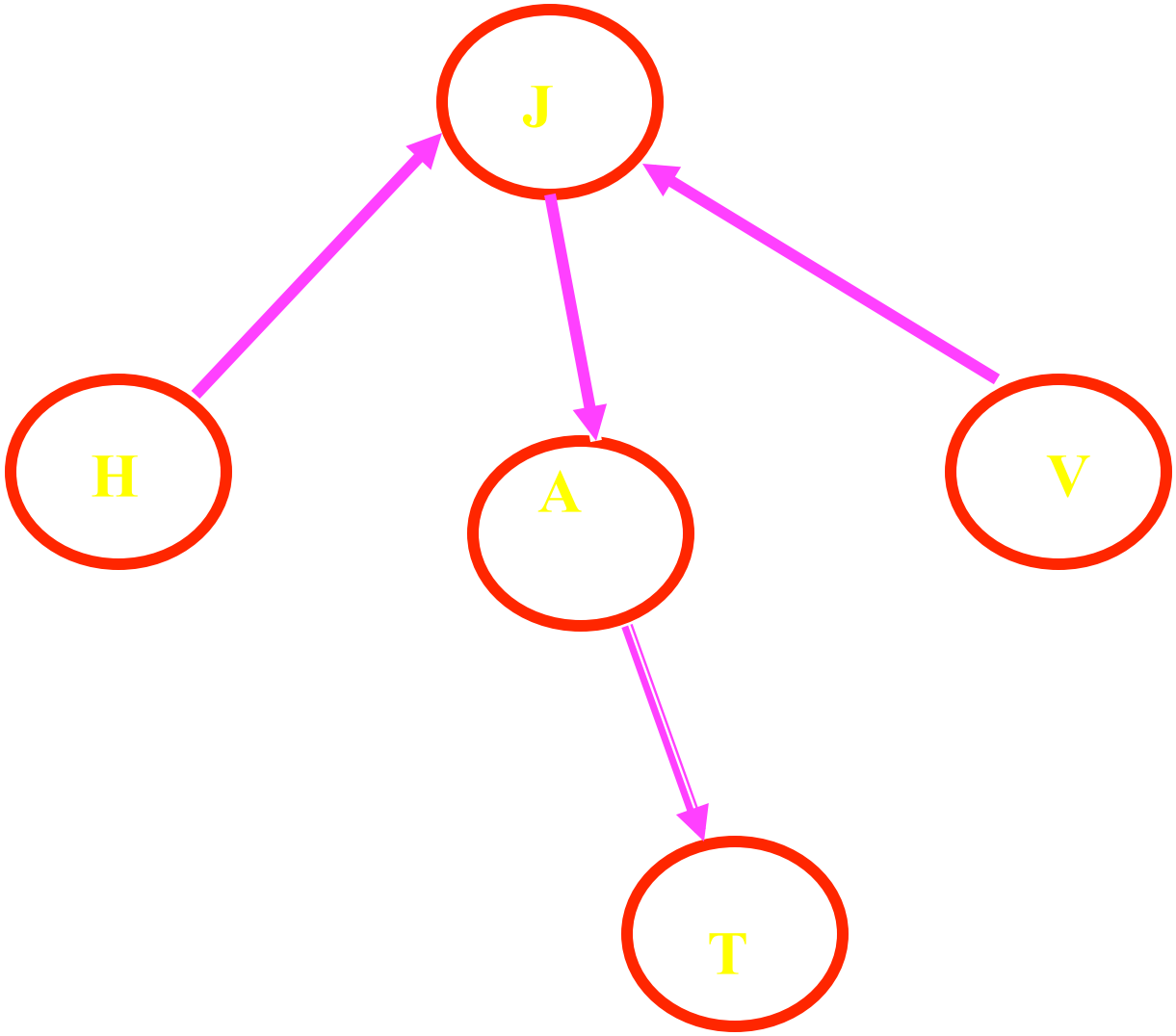
Prueba de Tripletas

- Tripleta de variables:
- Si son independientes dado , entonces pueden ser secuenciales o divergentes
- Si no son independientes dado , entonces son arcos convergentes

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



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)

Medida Bayesiana

- Maximizar la probabilidad de la estructura dados los datos:

- En términos relativos:

- Considerando variables discretas y que los datos son independientes, las estructuras se pueden comparar en función del número de ocurrencias (frecuencia) de los datos predichos por cada estructura

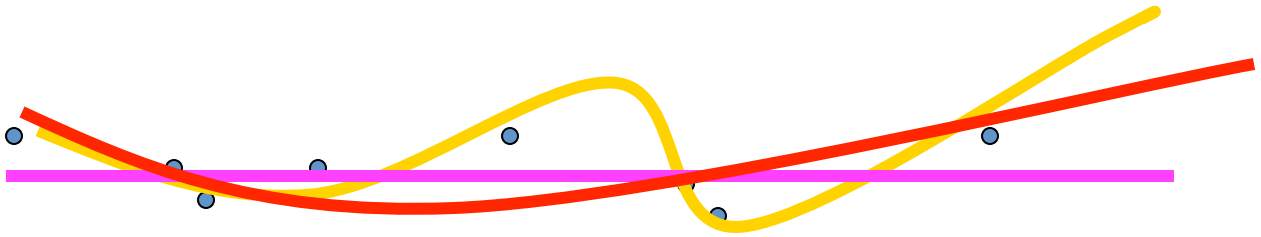
MDL

- 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:

Ejemplo – ajustar un polinomio a un conjunto de puntos:



MDL

Para redes bayesianas:

Complejidad:

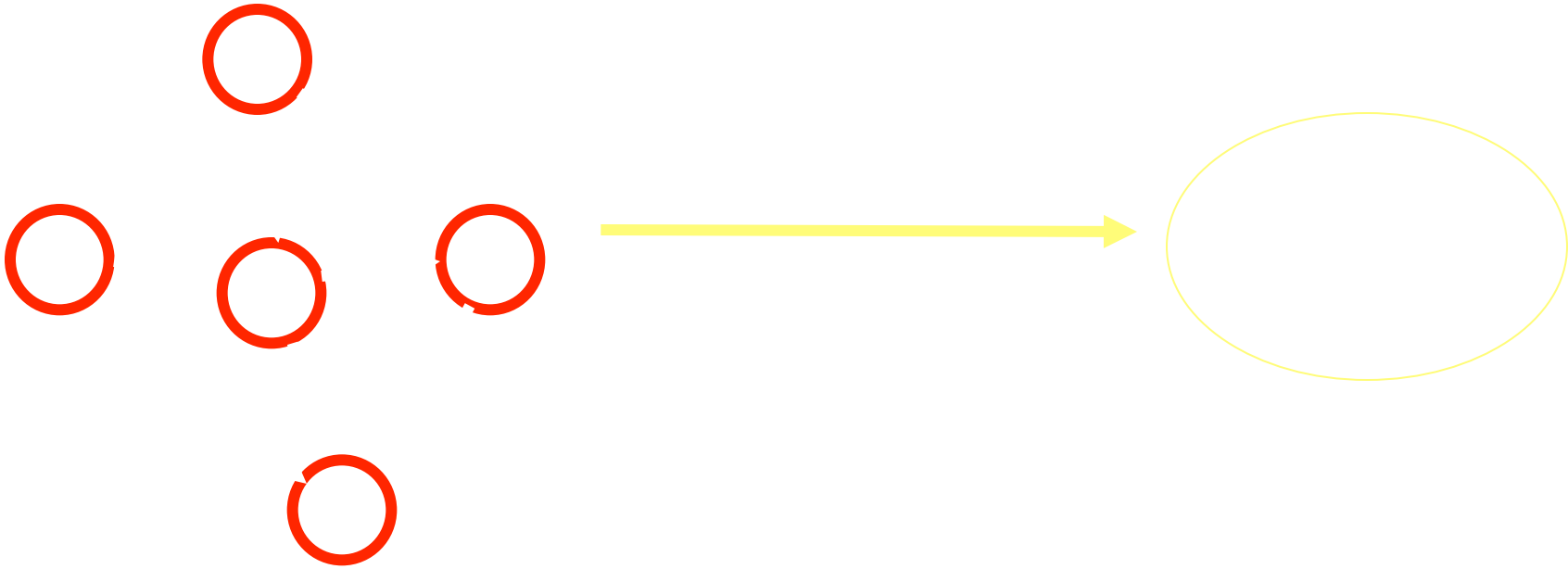
n-# de nodos, k-# padres por nodo, Si-# de valores por variable, Fi-conj. de padres, d-# de bits

Exactitud:

Buscando la mejor estructura

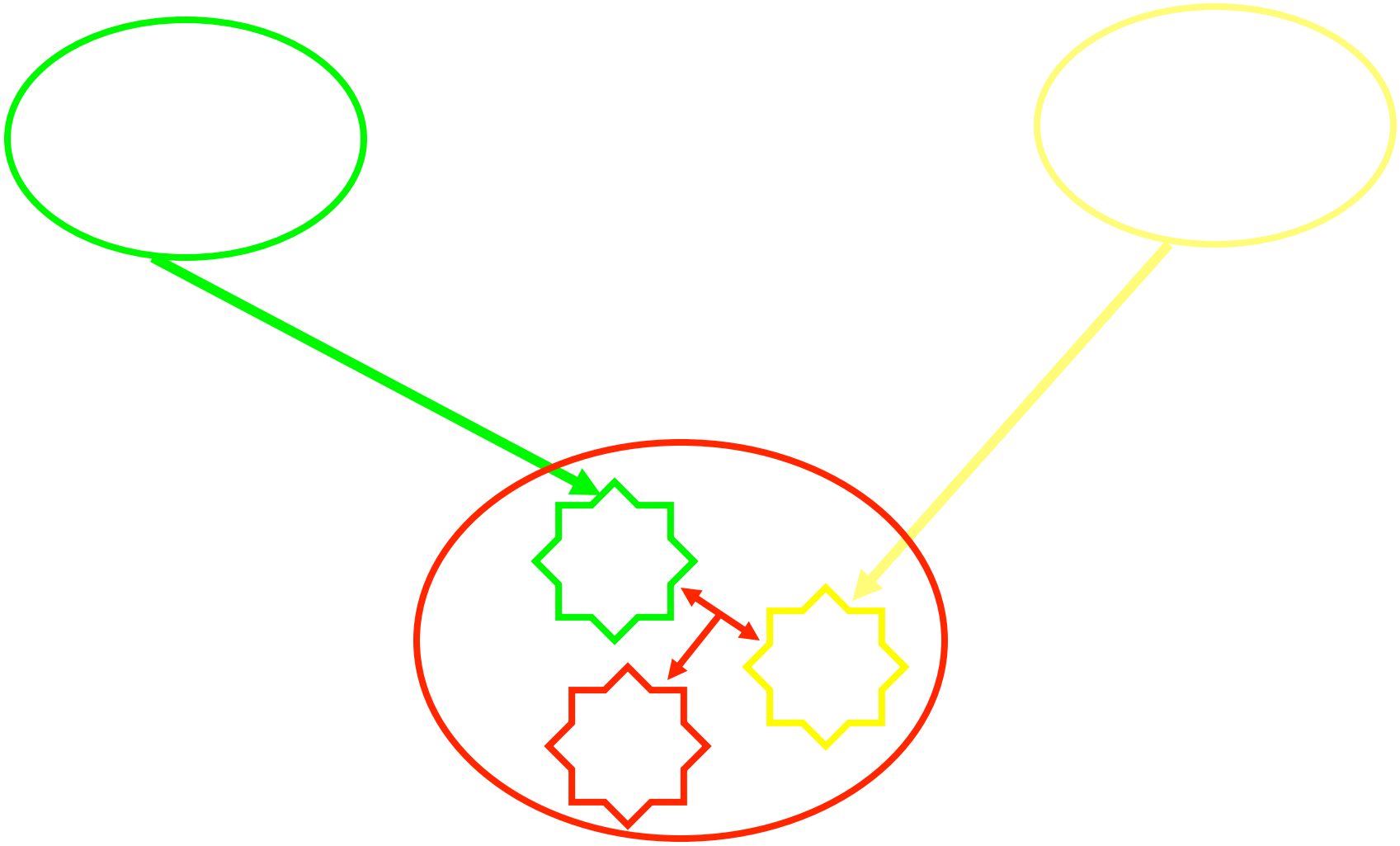
- Búsqueda de ascenso de colinas (*hill climbing*)
- Se inicia con una estructura simple (árbol) y se van agregando arcos hasta llegar a un *mínimo local*

Buscando la mejor estructura



- Se puede iniciar con una estructura compleja (máximo número de arcos) y se van eliminando arcos hasta llegar a un *mínimo local*

Búsqueda bidirecional



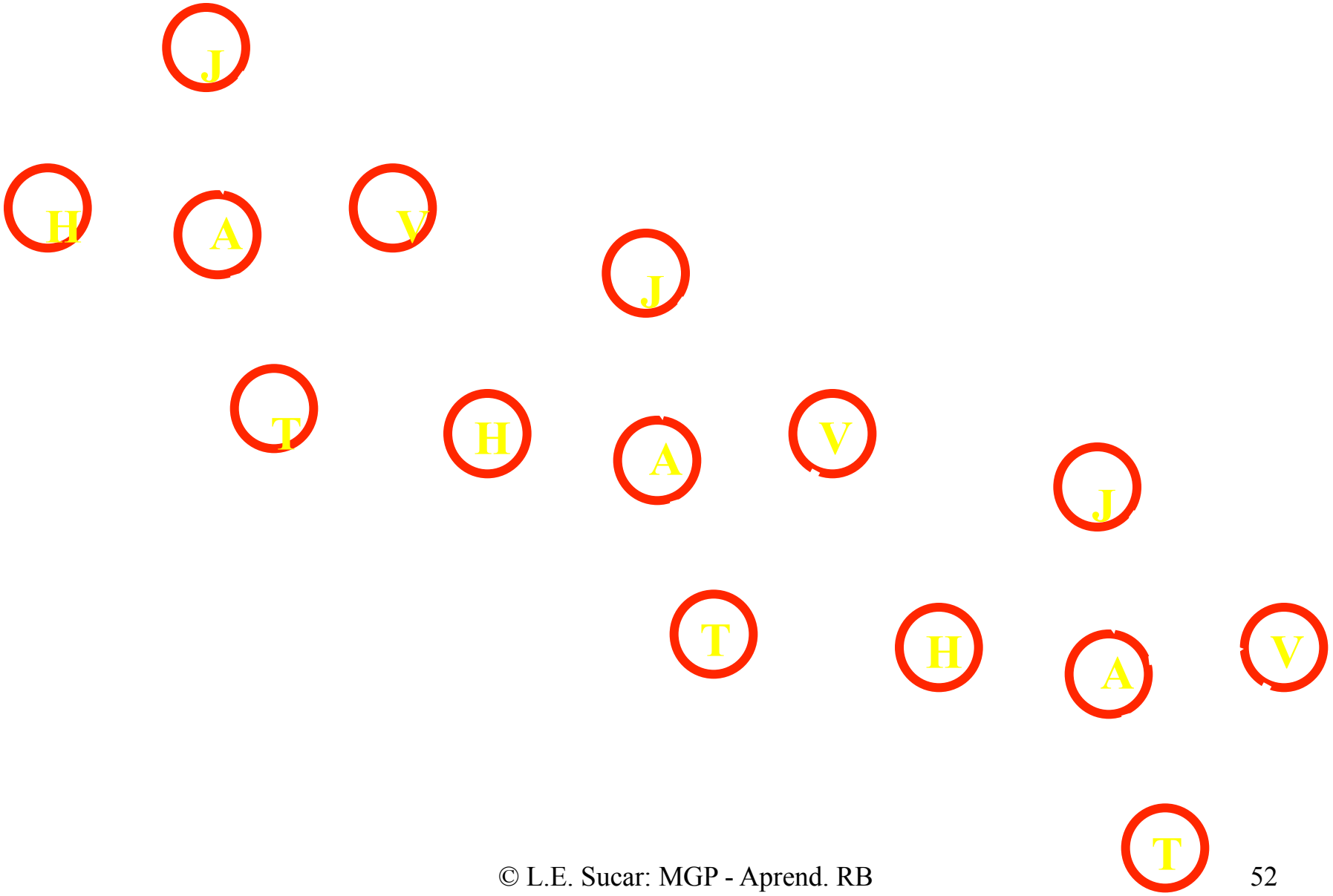
Algoritmo

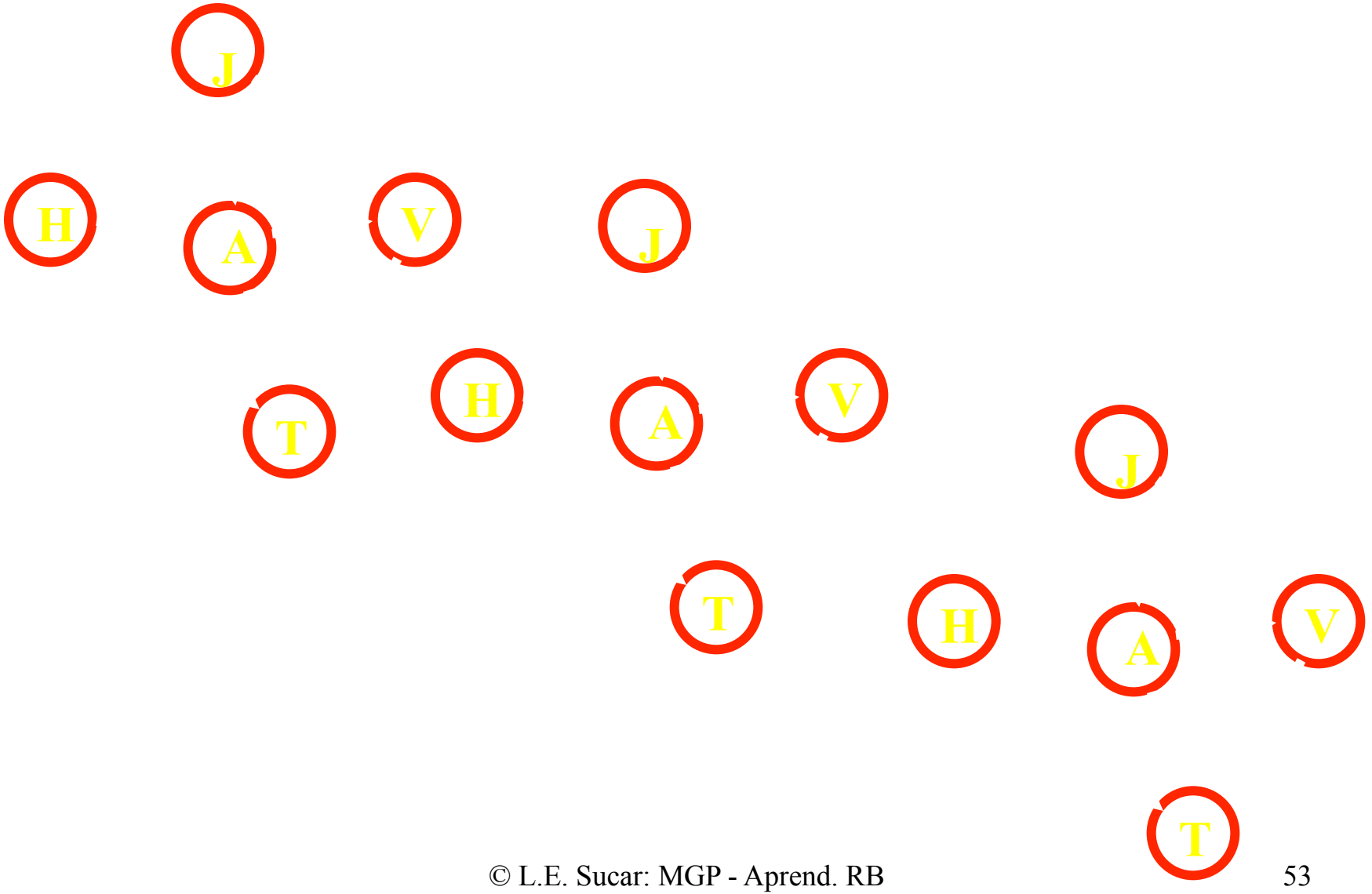
Parámetros

- Máximo número de padres
- Orden causal (opcional)
- Tamaño del *haz* en la última etapa

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


J
H
A
V
J
T
H
A
V
J
T
H
A
V
T

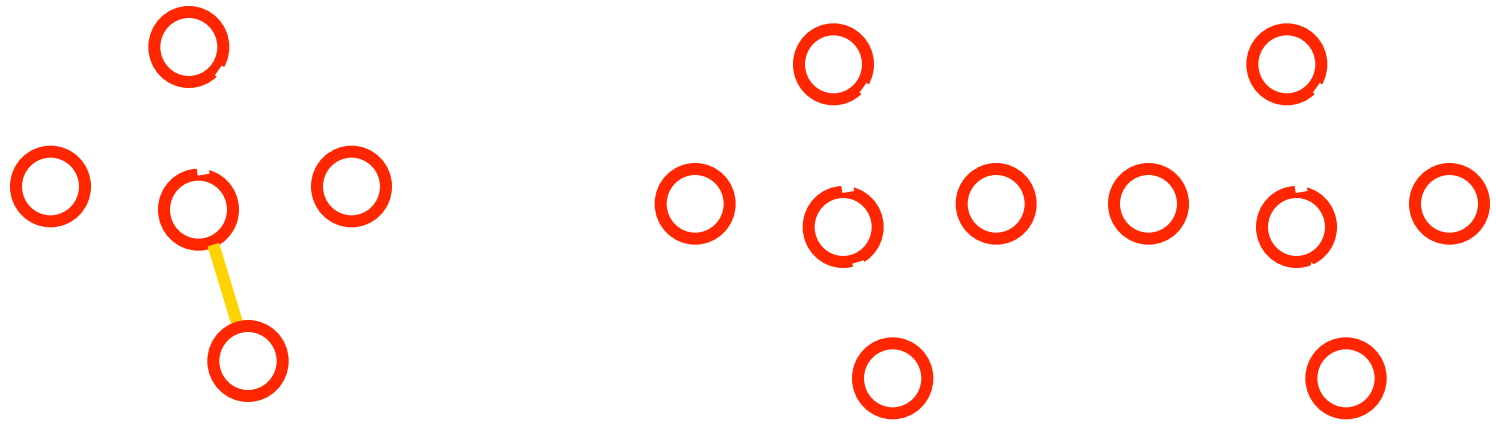

J
H
A
V
J
T
H
A
V
J
T
H
A
V
T

Variantes

- Utilizar otros métodos de búsqueda:
  - Algoritmos genéticos
  - "Beam search"
  - Etc.
- Considerar sólo estructuras que sean diferentes estadísticamente, buscando sobre estructuras equivalentes (se llega a una estructura parcial)

Estructuras Equivalentes

- Cuando ciertos arcos no se pueden determinar por pruebas estadísticas, por ejemplo:



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:

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

Prueba de Independencia

- Para probar si son independientes dado se utiliza la entropía cruzada condicional:

- 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)

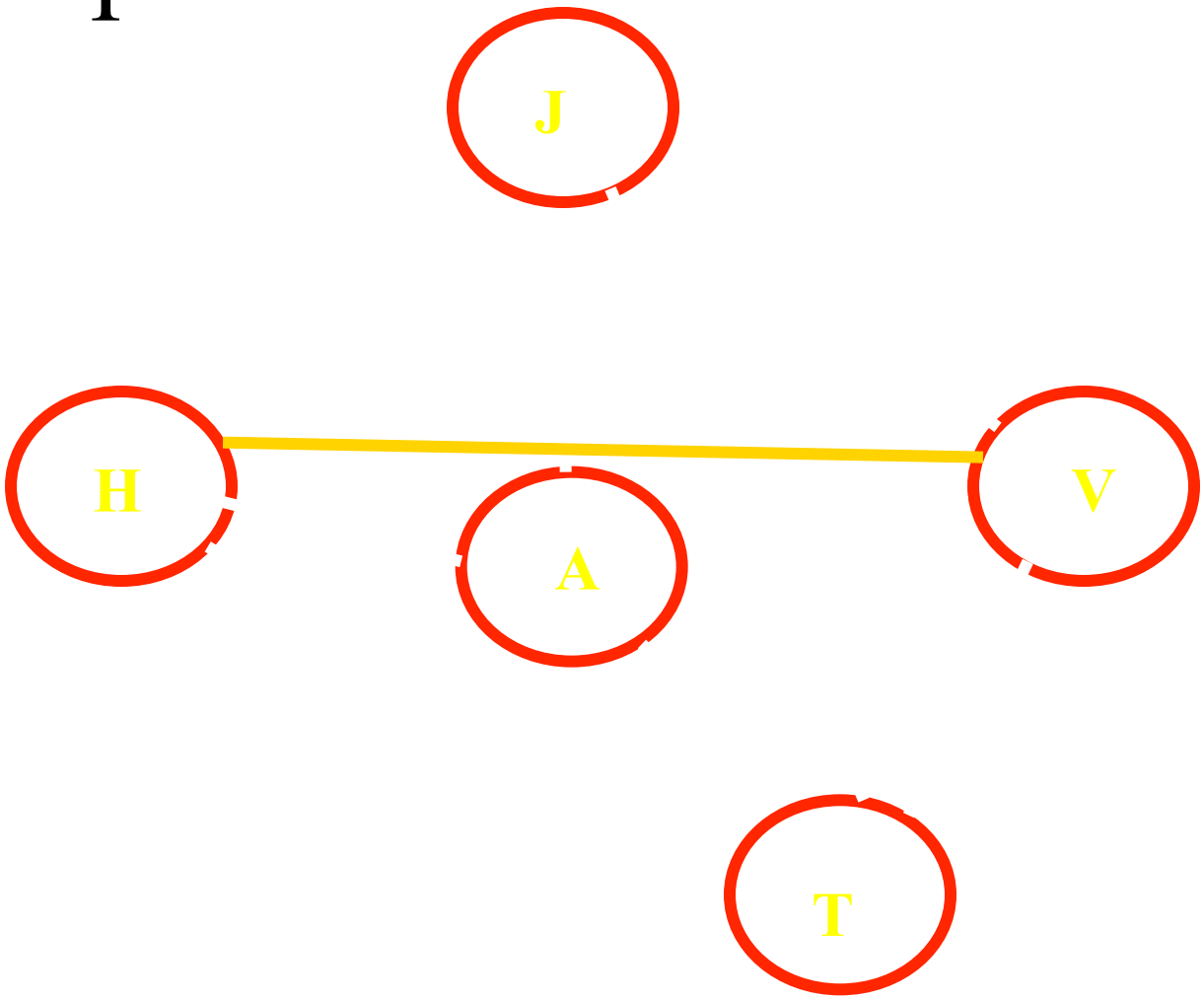
Algoritmo

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

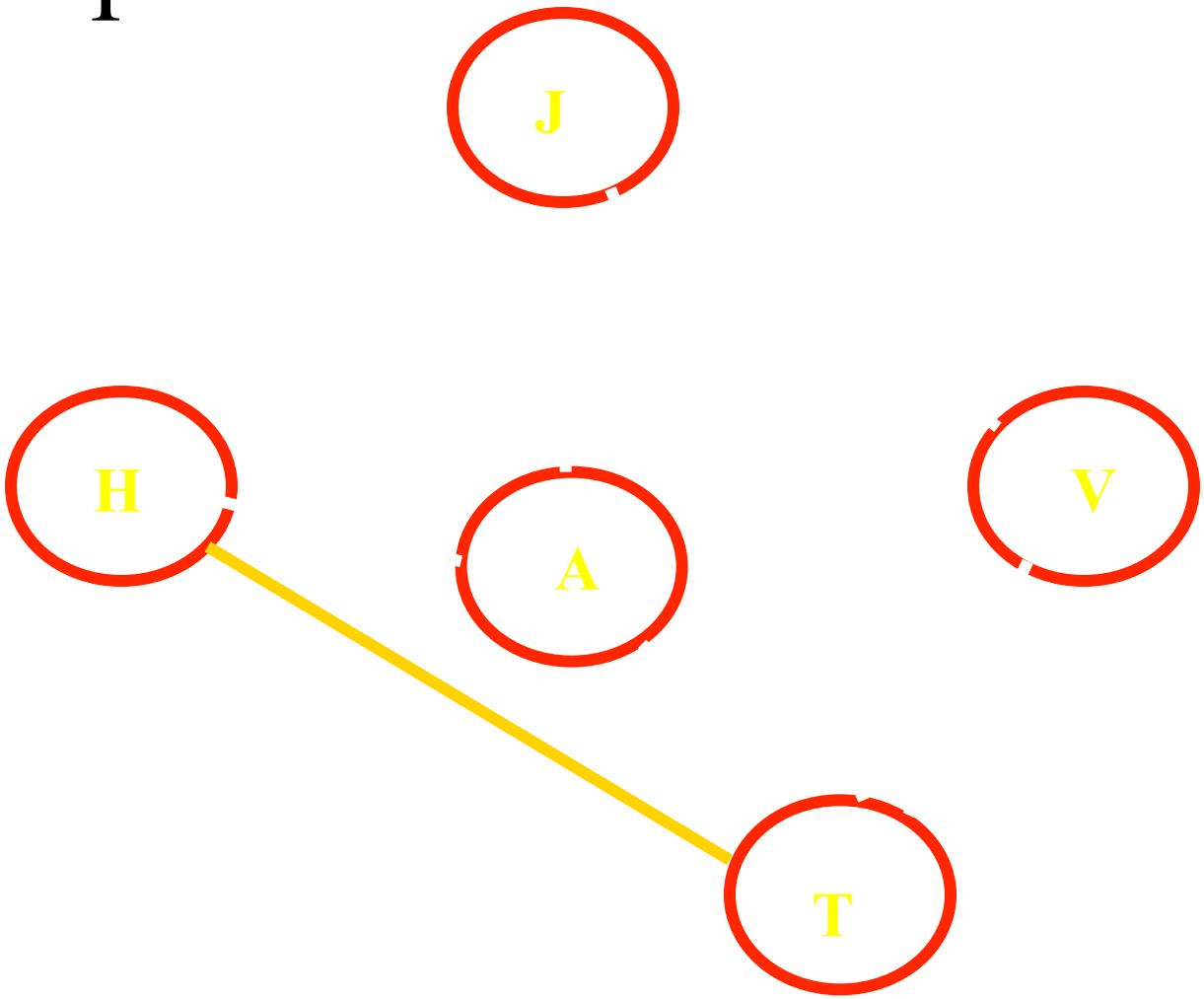
Esqueleto

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

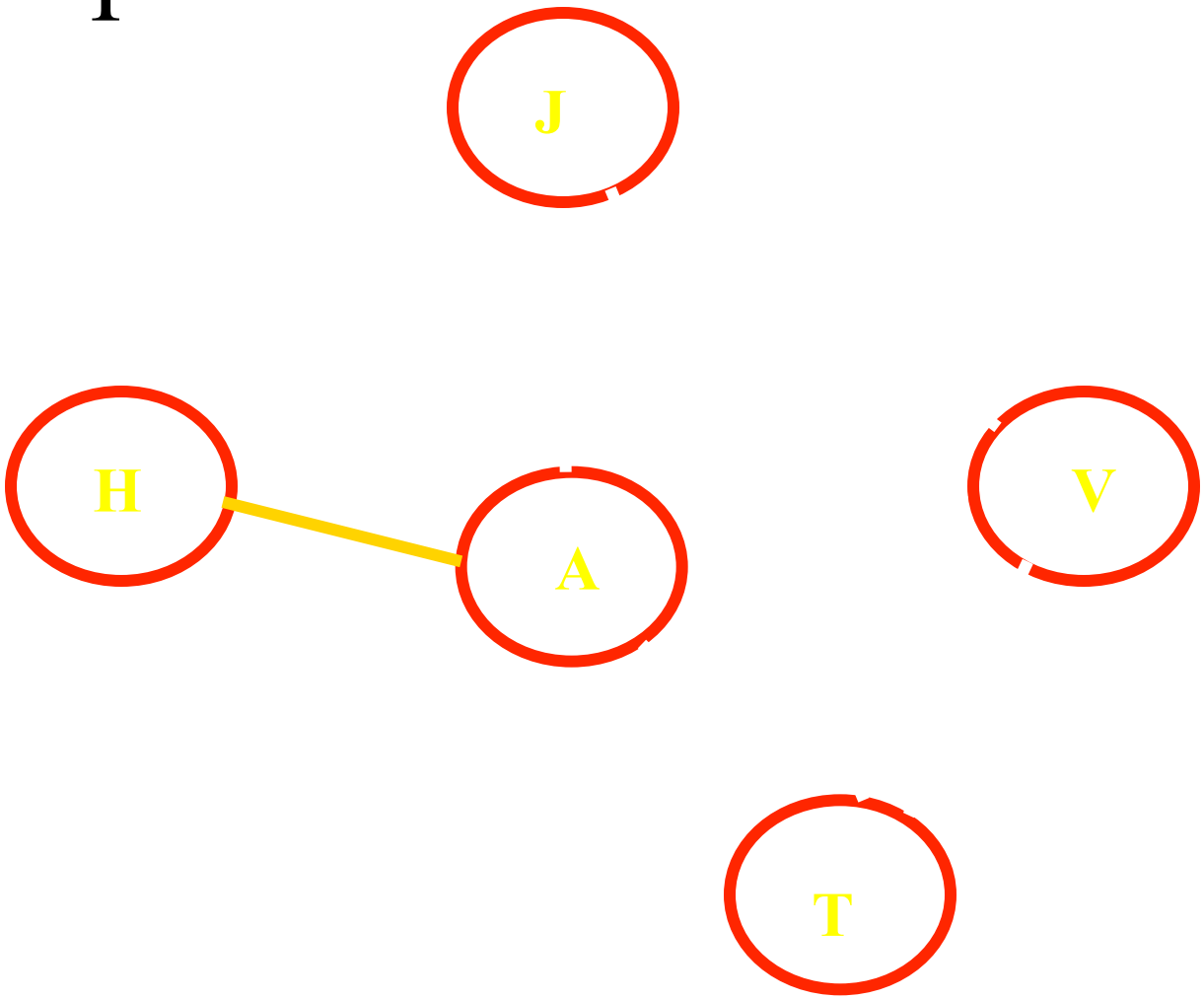
Ejemplo



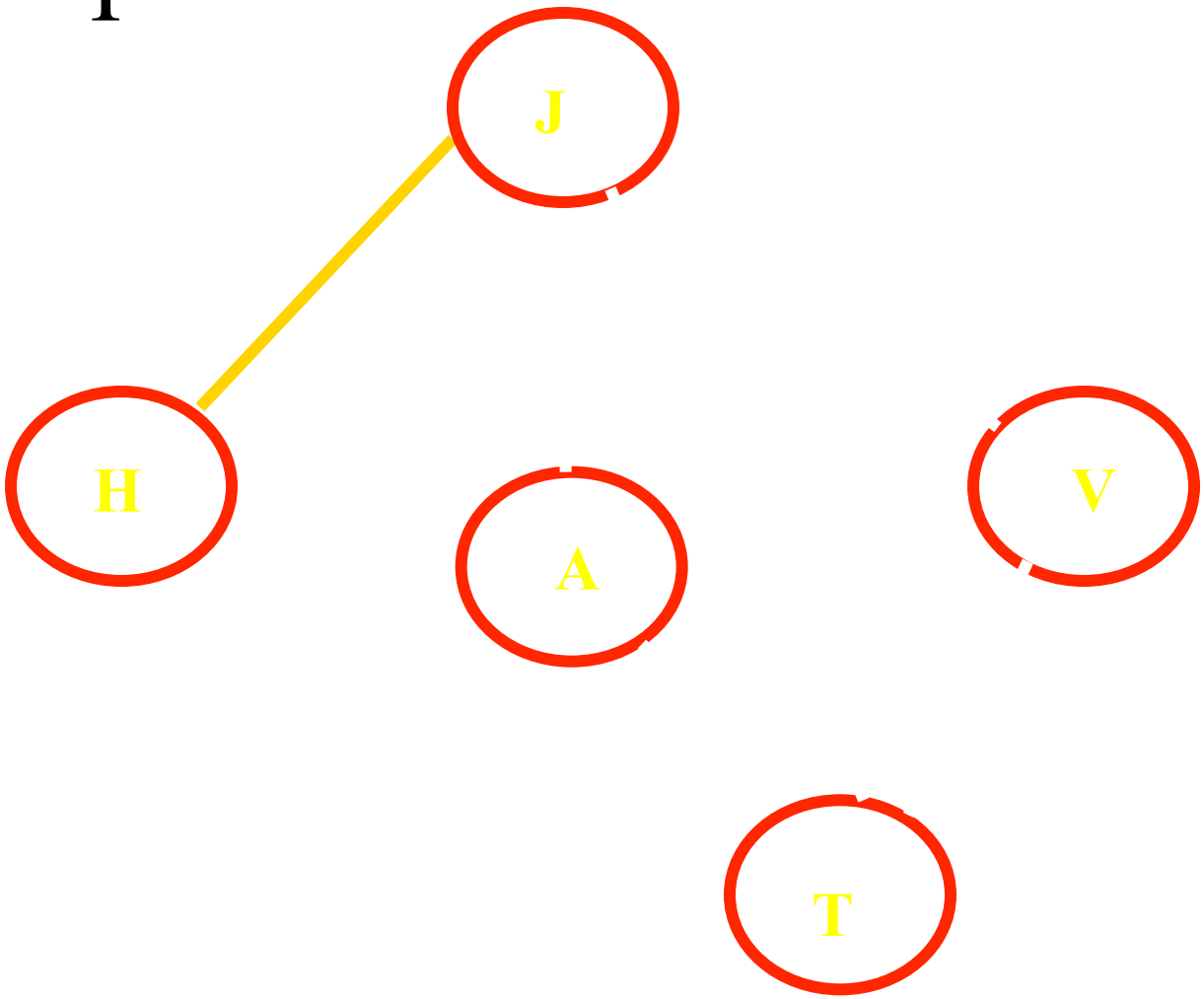
Ejemplo



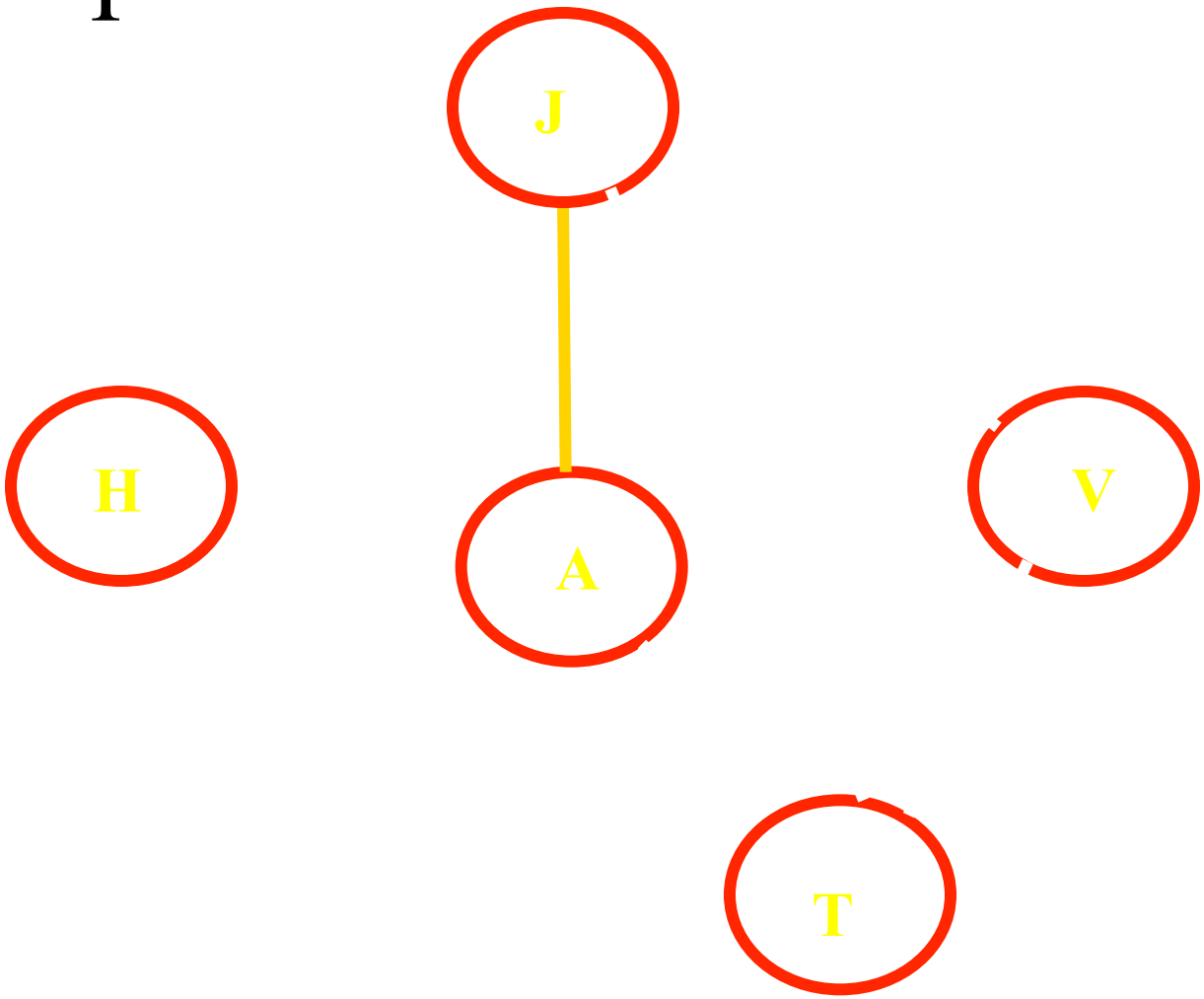
Ejemplo



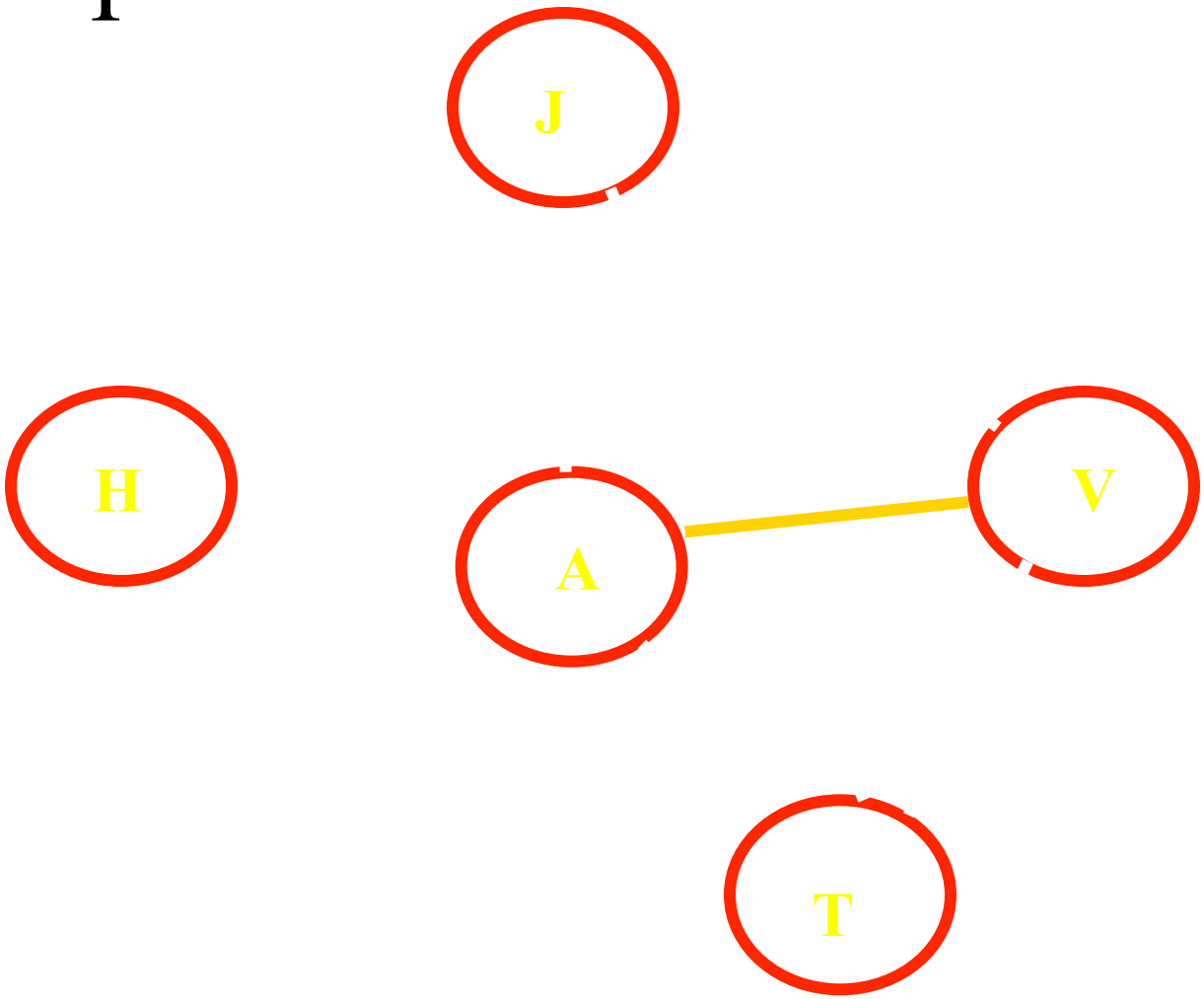
Ejemplo



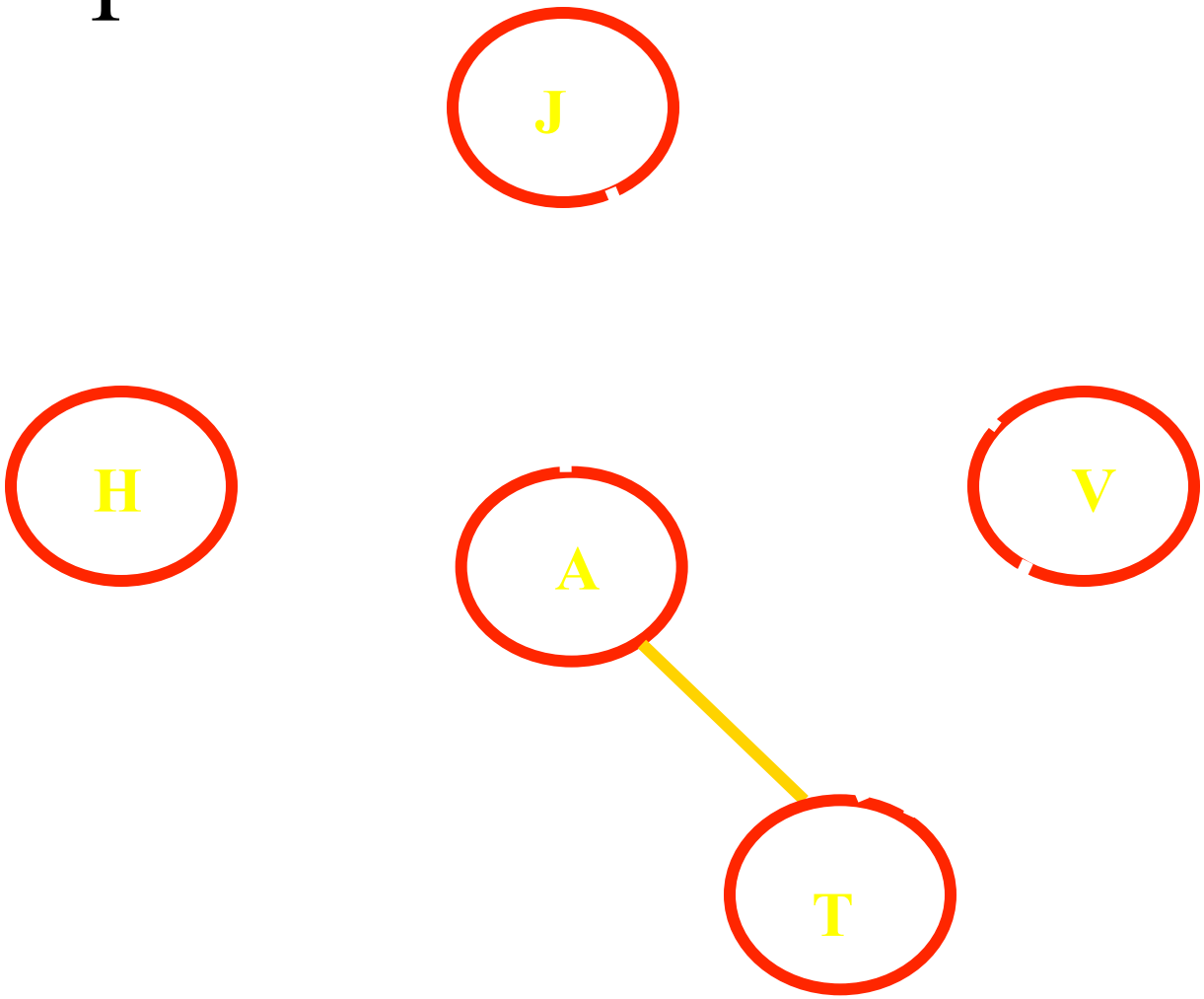
Ejemplo



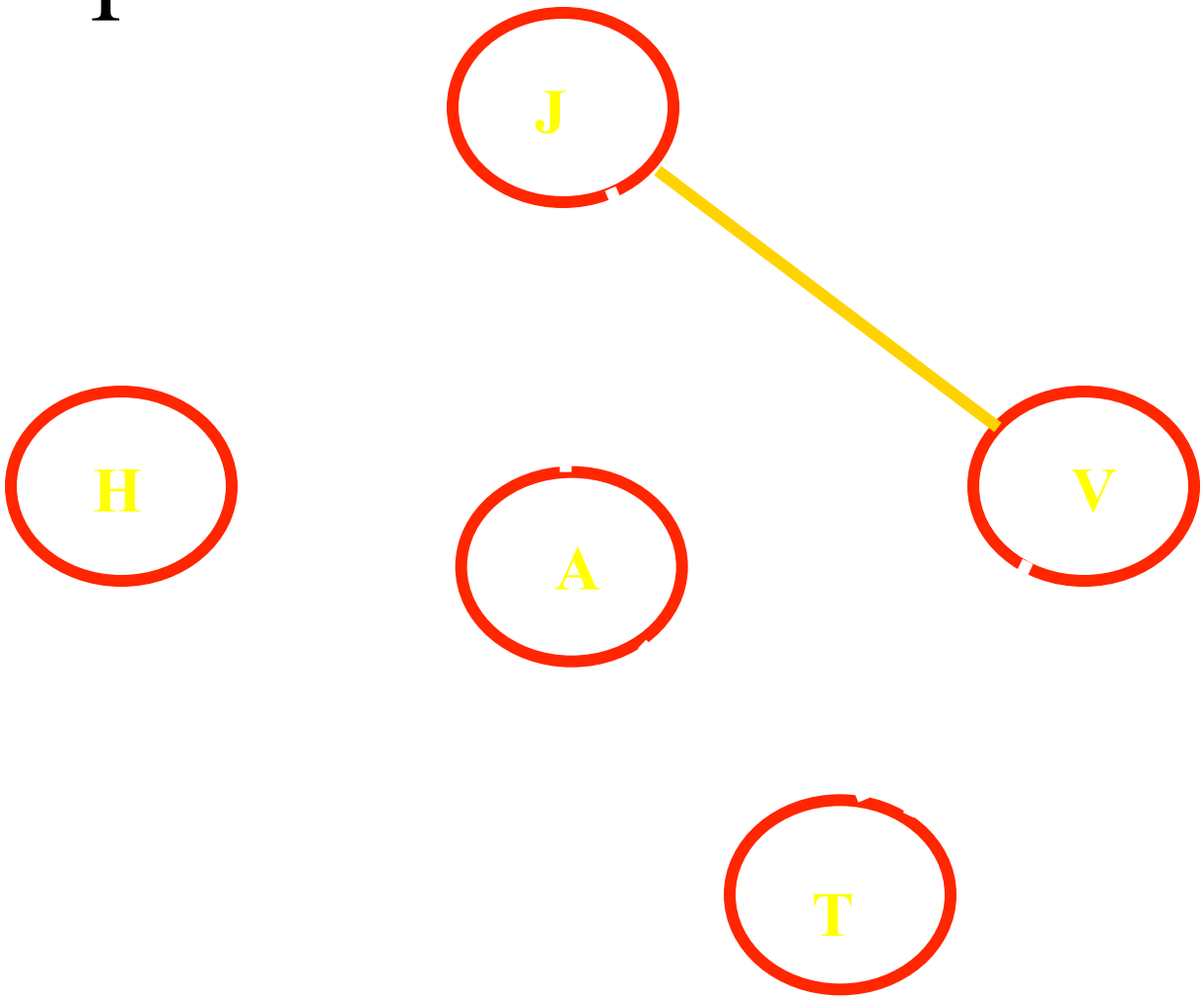
Ejemplo



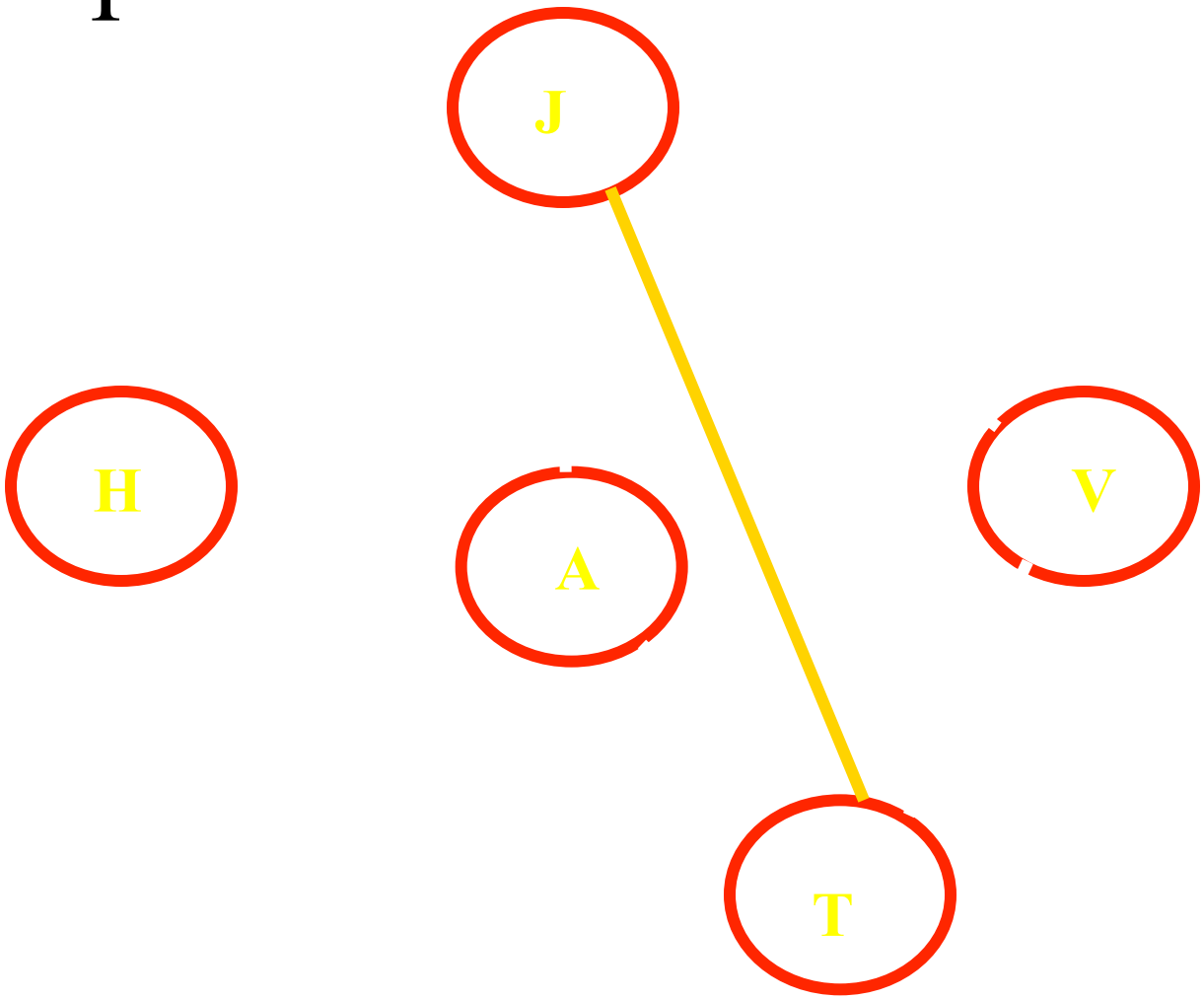
Ejemplo



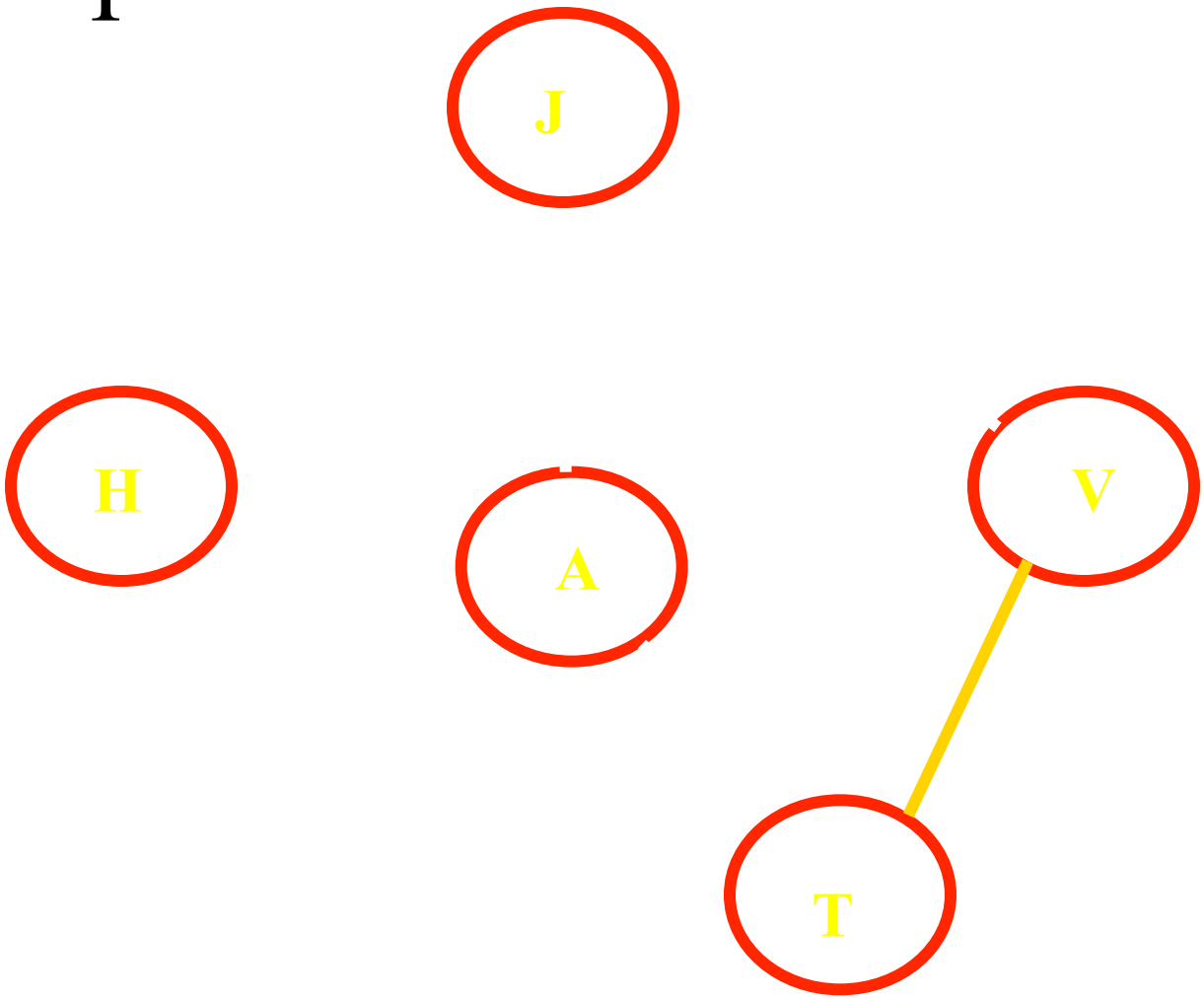
Ejemplo



Ejemplo



Ejemplo

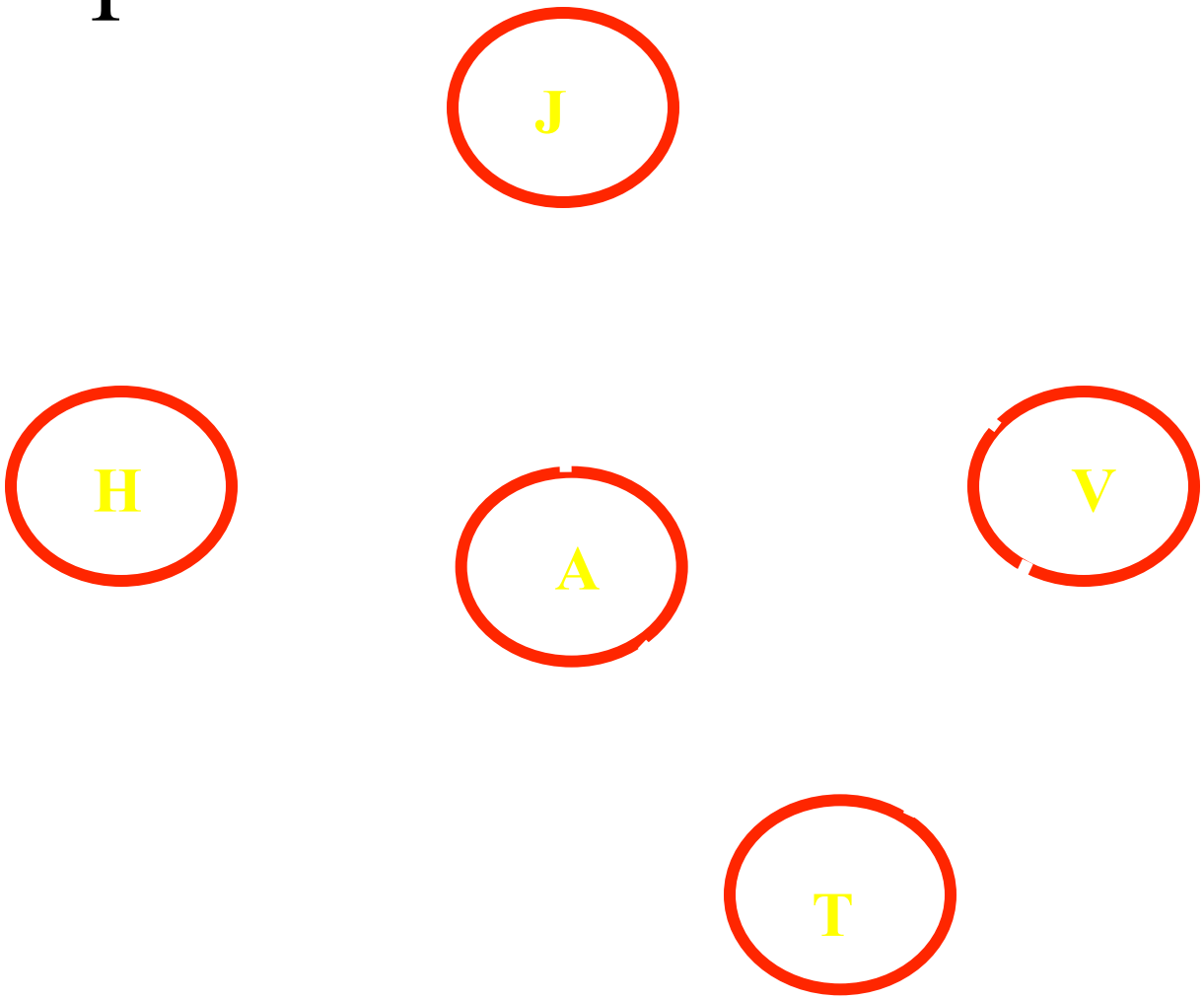


Arcos convergentes

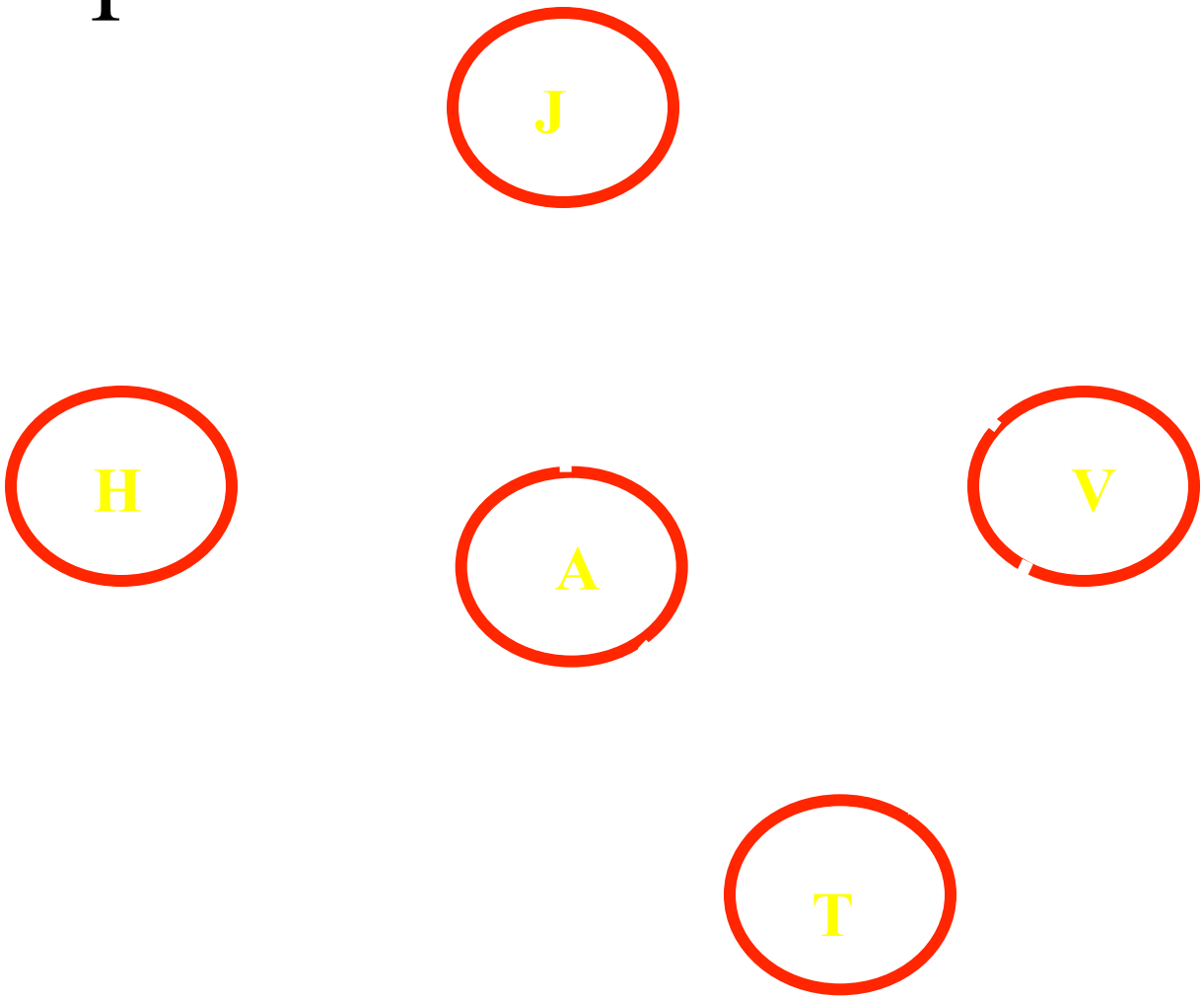
- Se verifica cada tripleta de variables para encontrar arcos convergentes mediante pruebas de independencia:

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

Ejemplo



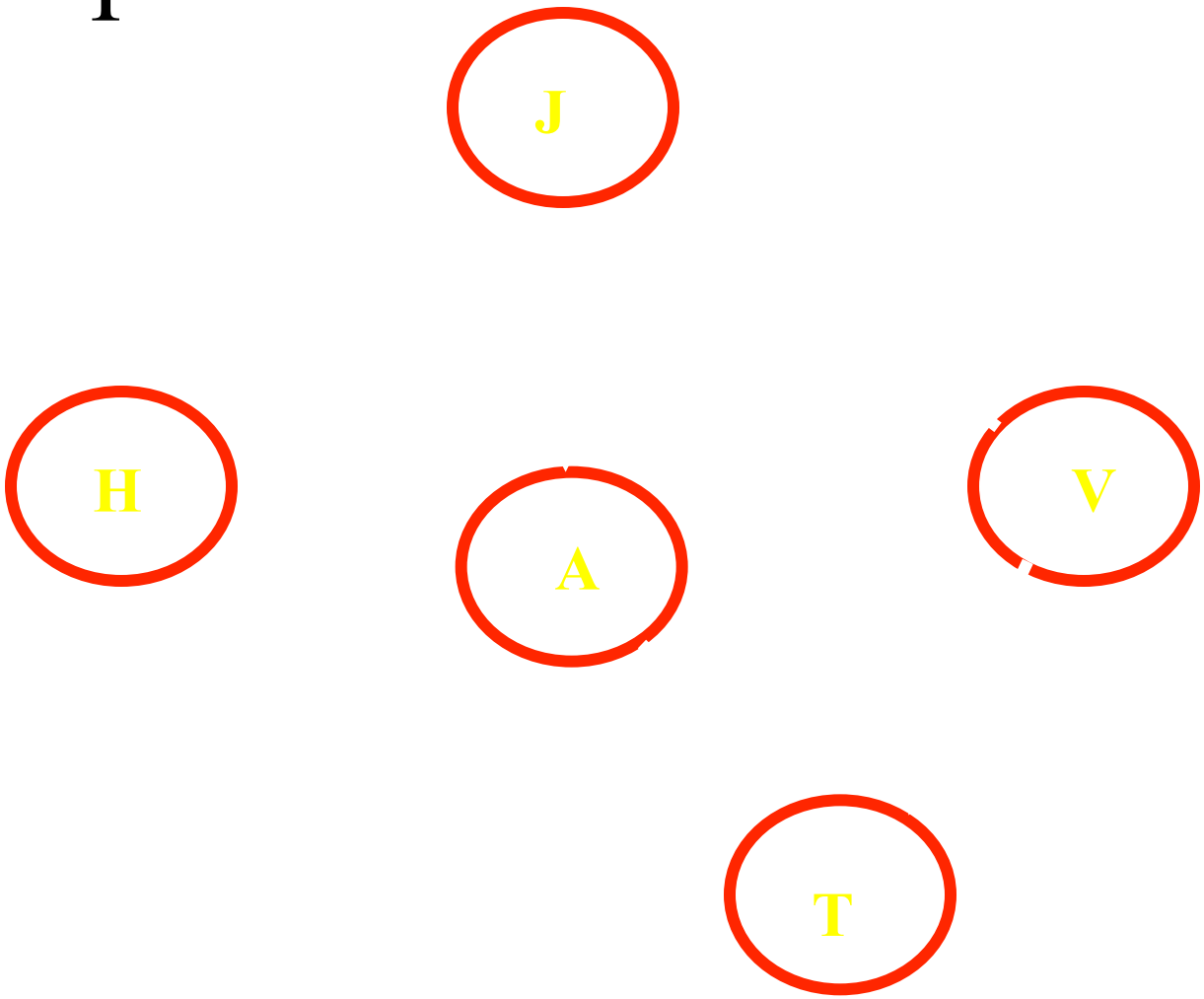
Ejemplo



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



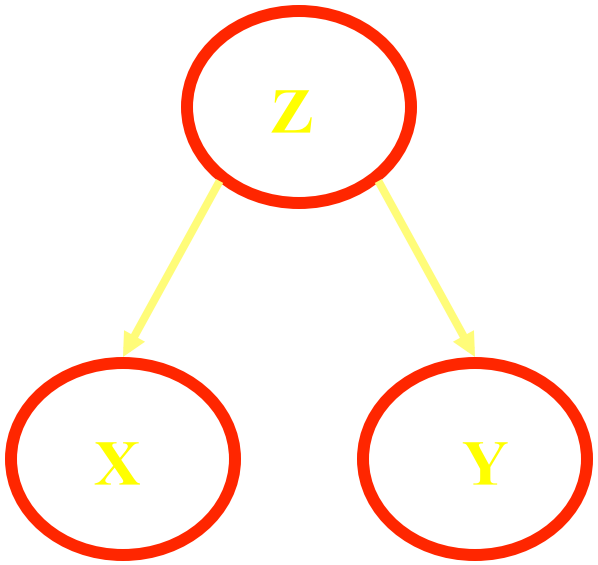
Combinación de conocimiento y datos

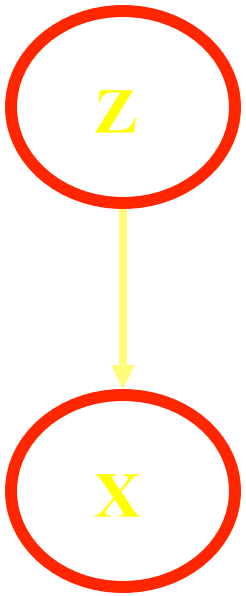
- Restricciones:
  - Se incorpora conocimiento previo a los algoritmos de aprendizaje estructural
  - Por ejemplo:
    - Orden de las variables (orden causal)
    - Dependencias conocidas
    - Independencias conocidas

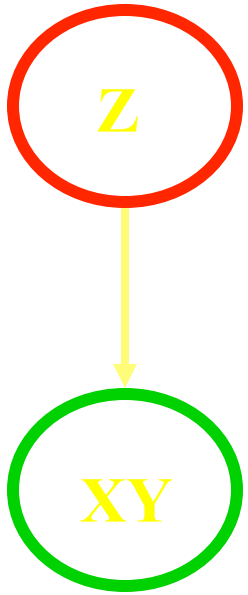
Combinación de conocimiento y datos

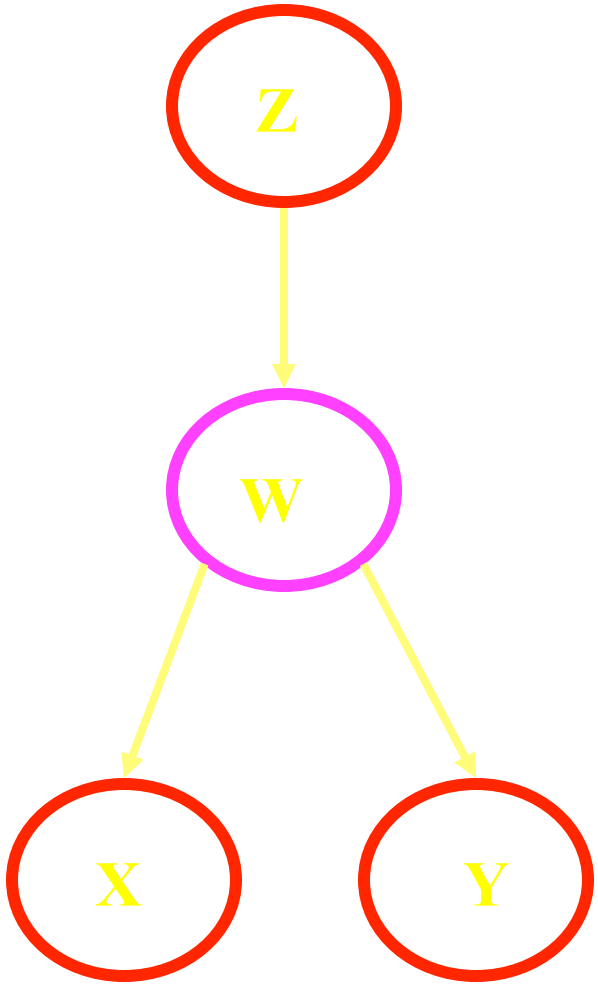
- Mejora:
  - Se parte de una estructura dada por un experto (subjetiva) y se mejora con datos
  - Por ejemplo, verificando relaciones de independencia y alterando la estructura:
    - Eliminar nodos
    - Combinar nodos
    - Insertar nodos

Mejora Estructural









Aprendizaje por Transferencia

- Todos los métodos de aprendizaje de RB requieren “suficientes” datos
- En ocasiones hay *pocos* datos para un dominio, pero *muchos* datos para otros dominios similares
- Entonces podemos tratar de usar los datos de dominios cercanos para aprender un mejor modelo para el dominio de interés

Algoritmo PC con Transferencia

- Se desarrolló una variante del algoritmo PC que incorpora transferencia de conocimiento
- Para ello las medidas de independencia condicional combinan los datos del dominio objetivo con los datos de dominios similares
- Se realiza una suma pesada de dichas medidas, donde el pesos depende de la "cercanía" al dominio objetivo y la cantidad de datos en cada dominio
- Esto da mejores resultados a simplemente juntar todos los datos de todos los dominios y aplicar PC

Referencias

- Pearl 88 – Cap. 8
- Neapolitan 90 – Cap. 10
- Koller & Friedman - Cap. 17, 18
- T. Mitchell, Machine Learning, McGraw-Hill, 1997 – Cap. 6
- Borglet & Kruse, Graphical Models, Wiley – Cap. 5 (EM)

Referencias

- W. Lam, F. Bacchus, "Learning Bayesian Belief Networks: An Approach based on the MDL Princlple", Computational Intelligence, Vol. 10 (1994) 269-293.
- G. Cooper, E. Herskovits, “A Bayesian method for the induction of probabilistic networks from data”, Machine Learning, Vol 9, 1992.
- G. Cooper, E. Herskovits, “A Bayesian method for the induction of probabilistic networks from data”, Machine Learning, Vol 9, 1992.
- W. Buntine, “A guide to the literature on learning probabilistic networks form data”, IEEE TKDE.

Referencias

- R. Neapolitan, “Learning Bayesian Networks”, Prentice-Hall, 2004.
- L. E. Sucar, D. F. Gillies, D. A. Gillies, “Objective Probabilities in Expert Systems”, Artificial Intelligence Journal, Vol. 61 (1993) 187-208.
- R. Luis, L. E. Sucar, E. F. Morales, “Inductive Transfer for Learning Bayesian Networks”, Machine Learning Journal, Vol. 79, 2010.