Modelos Gráficos Probabilísticos

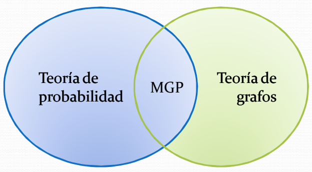
Descubrimiento de conocimiento a Partir de Datos

Dr. Marcelo G. Armentano
ISISTAN, Fac. de Cs. Exactas, UNICEN

Agenda

- Introducción
- Redes de Bayes
- Modelos de Markov

Modelos Gráficos Probabilísticos



- Forma natural de hacer frente a:
  - Incertidumbre
  - Complejidad

Modelos Gráficos Probabilísticos

- Representación
  - Como representamos de manera compacta una distribución de probabilidades conjunta
- Inferencia
  - Como inferimos los estados ocultos de un sistema basándonos en observaciones parciales
- Aprendizaje
  - Como estimamos los parámetros y la estructura del modelo

Redes de Bayes

Agenda

- Introducción
- Redes de Bayes
  - Definición
  - Ejemplos
  - Conceptos de Probabilidades
  - Utilización
  - Inferencia Probabilística
  - Construcción
  - Aprendizaje
- Modelos de Markov

Redes de Bayes

- Las Redes de Bayes se usan para modelar dominios que contengan incertidumbre.
- Se conocen también como:
  - Redes Bayesianas
  - Redes probabilísticas causales
  - Redes de creencias
  - Redes de creencias bayesianas

Incertidumbre: fuentes

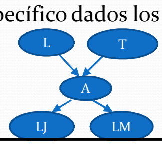
- Conocimiento incompleto del dominio
- Entendimiento/comprensión incorrecta del dominio
- Relaciones en el dominio de naturaleza no determinística (ej: enfermedades y síntomas)
- Términos involucrados muy vagos (grande, hermosa, dolor)
- Realización de algún tipo de abstracción
- Naturaleza aleatoria de los mecanismos que rigen el comportamiento del dominio (ej. modelado de usuarios)

Conceptos

- Una Red de Bayes consiste de un conjunto de **nodos** y de un conjunto de **arcos dirigidos** entre estos nodos.
- Los arcos reflejan relaciones **causa-efecto** dentro del dominio. Estos efectos normalmente no son determinísticos (ej: enfermedad → síntoma).
- La "fuerza" de un efecto es modelada como una **probabilidad**.
- La base matemática está dada por el **Teorema de Bayes**.

Definición

- Una **Red Bayesiana** es un **grafo dirigido acíclico** (GDA) donde cada nodo representa una **variable aleatoria** y los arcos entre los nodos representan **dependencias probabilísticas** entre las variables. Cada nodo contiene los **estados** de la variable que representa y una **tabla de probabilidad condicional** (CPT). La CPT de un nodo contiene las probabilidades de que el nodo esté en un estado específico dados los estados de sus padres.



Nodos

- Un nodo representa una variable aleatoria **discreta** con un número finito de estados o una variable aleatoria **continua** (distribución Gaussiana).
- Si un nodo no tiene padres, el nodo contendrá una **tabla de probabilidad marginal**.
  - Si el nodo es discreto, contiene una distribución de probabilidad sobre los estados de la variable que representa.
  - Si el nodo es continuo, contiene una función de densidad Gaussiana (dada por la media y la varianza) para la variable aleatoria que representa.

Tablas de probabilidad

- Si el nodo tiene padres, contiene una tabla de probabilidad condicional (CPT).
  - Si el nodo es **discreto**, cada celda en la CPT de un nodo contiene una probabilidad condicional para cada estado en que pueda estar el nodo dada una configuración específica de los estados de sus padres.
  - Si el nodo es **continuo**, la CPT contiene una media y una varianza para cada configuración de los estados de sus padres discretos y un coeficiente de regresión por cada padre continuo.

Ejemplo: Apple Jack

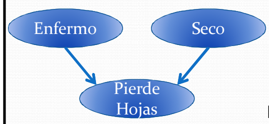
- Un día Apple Jack descubre que su mejor árbol de manzanas está perdiendo sus hojas. Entonces quiere saber qué está pasando. Él sabe que si el árbol está seco es muy común que pierda sus hojas. Por el otro lado, perder las hojas puede ser un indicio de que el árbol esté enfermo.

¿variables?

¿estados?

¿relaciones?

Ejemplo: Apple Jack



P(seco)

Seco=si	Seco=no
0.1	0.9

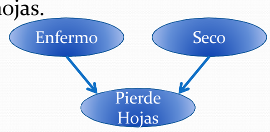
P(enfermo)

Enfermo=si	Enfermo=no
0.1	0.9

p(Pierde/ Seco, Enfermo)	Seco=si Enfermo=si	Seco=si Enfermo=no	Seco=no Enfermo=si	Seco=no Enfermo=no
Pierde=si	0.95	0.85	0.90	0.02
Pierde=no	0.05	0.15	0.10	0.98

Dependencias Causales

- Cuando existe una **dependencia causal** de un nodo A a un nodo B, se espera que cuando A está en un cierto estado esto tiene un **impacto** en el estado de B.
- En el ejemplo, existe un vínculo causal de “Enfermo” a “Pierde hojas” porque cuando un árbol está enfermo puede causar que pierda las hojas.

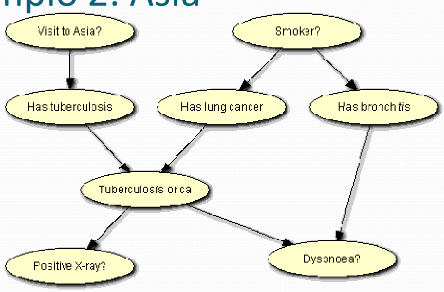


Independencia condicional

- Dos conjuntos de variables A y B son (condicionalmente) independientes de un tercer conjunto C si cuando los valores de C son conocidos, el conocimiento acerca de B no provee más información sobre los valores de A.

p(A/B,C) = p(A/C)

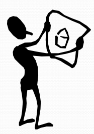
Ejemplo 2: Asia



p(Bronchitis/Visit Asia,Smoker) = p(Bronchitis/Smoker)

Distribución de probabilidad conjunta

Dado un conjunto de variables V1,V2,...,Vk cuyos valores se denotan v1,v2,....,vk, una expresión de la forma p(V1,V2,...,Vk) se denomina función de probabilidad conjunta sobre las variables V1,V2,.....,Vk.



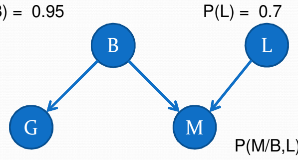
Ejemplo 3: Robot

- Un robot es capaz de levantar una bandeja si la bandeja es liviana y si las baterías de la fuente de energía del robot están cargadas. Si estas condiciones se satisfacen, entonces cuando el robot trata de levantar la bandeja, mueve sus brazos. Se determina si la batería está cargada mediante un medidor.

¿variables?
¿estados?
¿relaciones?

Ejemplo 3: Robot

P(B) = 0.95   P(L) = 0.7



P(G/B) = 0.95
P(G/¬B) = 0.1

P(M/B,L) = 0.9
P(M/B,¬L) = 0.05
P(M/¬B,L) = 0.0
P(M/¬B,¬L) = 0.0

Ejemplo 3: Robot

- P(B,M,L,G)
- 4 variables binarias binarias → 16 probabilidades

(B,M,L,G)	Prob. Conj.
(V,V,V,V)	0.5686
(V,V,V,F)	0.0299
(V,V,F,V)	0.0135
(V,V,F,F)	0.0007
…	…

Ejemplo 3: Robot

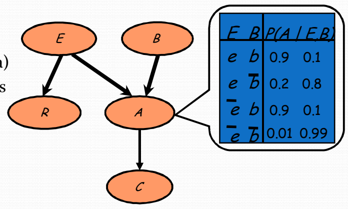
- P(A,B,D,L,S,X,T)
- 7 variables binarias → 128 Probs. Conjuntas!!!!

Problema de la probabilidad conjunta

- Se necesitan en el orden de 2^k valores para definir la función de probabilidad conjunta.
- Solución
  - Las relaciones de independencia condicional entre las variables contribuyen a la disminución de la complejidad del problema

Redes de Bayes

- Representación compacta de distribuciones de probabilidades via independencia condicional
- Parte cualitativa:
  - Grafo dirigido acíclico (gda)
  - Nodos - variables aleatorias
  - Arcos – influencia directa
- Parte cuantitativa:
  - Conjunto de distribuciones de probabilidades condicionales

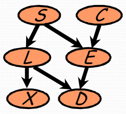


E	B	P(A | E,B)	
e	b	0.9	0.1
e	$\bar{b}$	0.2	0.8
$\bar{e}$	b	0.9	0.1
$\bar{e}$	$\bar{b}$	0.01	0.99

- Juntos definen una única distribución conjunta de manera factorizada

$$P(B,E,A,C,R) = P(B)P(E)P(A \mid B,E)P(R \mid E)P(C \mid A)$$

Semántica de las Redes de Bayes



Independencias condicionales en la estructura de la red + Modelos de probabilidades locales = Distribución de probabilidad conjunta sobre un dominio

$$P(s,\bar{c},l,\bar{e},\bar{x},d) = P(s)\,P(\bar{c})\,P(l\mid s)P(e\mid s,\bar{c})\,P(\bar{x}\mid l)\,P(d\mid l,\bar{e})$$

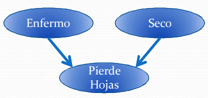
- Representación natural y compacta

Agenda

- Definición
- Ejemplos
- Conceptos de Probabilidades
- **Utilización**
- **Inferencia Probabilística**
- Construcción
- Aprendizaje

Utilización de Redes de Bayes

- Obtener los valores de probabilidades de ciertas variables de interés a partir de la información (**evidencia**) del estado de otras variables.
- Ej: Sabemos que el árbol está perdiendo sus hojas,
  - ¿Cuál es la probabilidad de que el árbol esté enfermo?
  - ¿Cuál es la probabilidad de que el árbol esté seco?



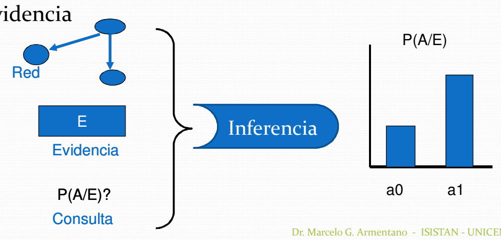
Inferencia

Inferencia probabilística

- Consiste en calcular probabilidades de interés a partir de una distribución de probabilidades conjunta.
- La estructura de la red de Bayes permite calcular cualquier probabilidad de interés; donde la independencia condicional simplifica el cálculo

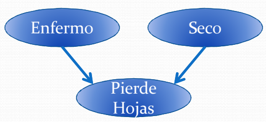
Inferencia en Redes de Bayes

- La inferencia consiste en el cálculo de los resultados a una consulta (distribución de probabilidades) con respecto a una red dada y a la presencia de cierta evidencia



Ejemplo de inferencia

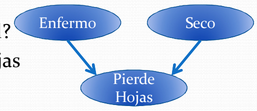
Consulta: ¿Está enfermo el árbol?

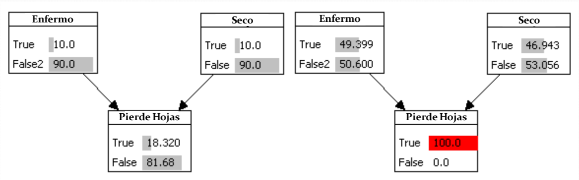


Evidencia: el árbol pierde las hojas

Ejemplo de inferencia

- **Consulta**: ¿Está enfermo el árbol?
- **Evidencia**: el árbol pierde las hojas





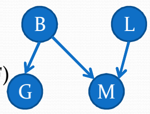
Teorema de Bayes

- El teorema de Bayes nos dice como obtener una probabilidad a posteriori en una hipótesis Vi luego de la observación de alguna evidencia Vj, dadas la probabilidad a priori de Vi y la probabilidad de observar Vj dado Vi.

$$p(Vi/Vj) = \frac{p(Vj/Vi)\,p(Vi)}{p(Vj)}$$

Ejemplo del Teorema de Bayes

- Si una persona ve que un brazo del robot se está moviendo, se quiere obtener la probabilidad de que esto haya ocurrido cuando la batería está descargada.
  - Evidencia: p(M = V)
  - Probabilidad a priori: p(B = F)
  - Probabilidad condicional: p(M = V / B = F)



- P(B = F / M = V) = p(M = V / B = F) p(B = F) / p(M = V)

Inferencia en Redes de Bayes

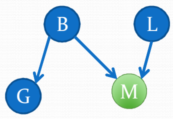
- Inferencia causal o top-down
- Inferencia de diagnóstico o bottom-up

Inferencia causal

- Probabilidad de que el brazo se mueva dado que la bandeja es liviana

P(M/L)

Consulta Evidencia



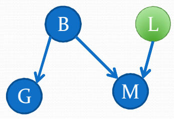
p(M/L) = p(M,B/L) + p(M, ¬B/L)   (expansión)
= p(M/B,L) P(B/L)+ p(M/ ¬B,L) P(¬B/L)
= p(M/B,L) P(B)+ p(M/ ¬B,L) P(¬B)

Inferencia de diagnóstico

- Probabilidad de que la bandeja esté pesada dado que el brazo no se mueve

P(¬L/¬M)

Causa Efecto



p(¬L/¬M) = p(¬M/¬L) p(¬L) / p(¬M)   Bayes

Se resuelve p(¬M/¬L) usando razonamiento causal

Inferencia exacta

- A pesar de poder explotar las propiedades de la red, la inferencia probabilística exacta en una red de Bayes arbitraria es NP-hard.
- Para muchas aplicaciones, donde las redes son pequeñas o pueden simplificarse, la complejidad puede no ser fatal.

Algoritmos de Inferencia

- Existen numerosos algoritmos de inferencia implementados
- Centrados en operaciones algebraicas
  - Variable Elimination [Zhang96]
  - Bucket Elimination [Dechter99]
  - SPI [D'Ambrosio94]
- Centrados en propiedades gráficas
  - Junction trees [Jensen96]

Agenda

- Definición
- Ejemplos
- Conceptos de Probabilidades
- Utilización
- Inferencia Probabilística
- **Construcción**
- **Aprendizaje**

Construcción de la red

- Ingeniería del Conocimiento
  - Un experto en el dominio identifica los aspectos cualitativos (y a veces cuantitativos) del problema
  - Codificar el conocimiento existente de expertos en una red
  - Usar una base de datos para actualizar este conocimiento
    - refinar el conocimiento experto original e identificar nuevas relaciones

Construcción de la red

- Determinar las **variables** que intervienen en el modelo
- Determinar cada uno de los **estados** o valores que pueden tomar las variables
- Determinar las relaciones (**dependencias**) entre las variables
- Construir un GDA que codifique las aserciones de independencia condicional

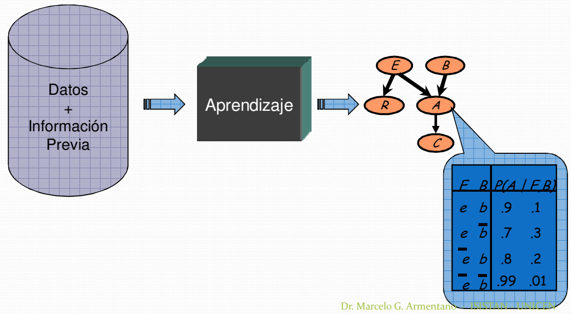
Construcción de la red

- Cómo obtener los valores de las probabilidades?
  - Conocimiento de expertos en el dominio
  - Estudios estadísticos
  - Derivados analíticamente
  - Aprenderlos a partir de los datos crudos

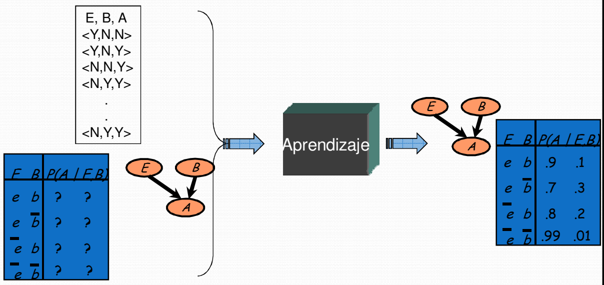
Construcción de la red

- Cómo obtener la estructura de la red?
  - A partir del conocimiento de un experto
  - Algoritmos de aprendizaje en Redes de Bayes
    - Datos completos o incompletos
    - Estructura conocida o desconocida

Aprendizaje de Redes de Bayes

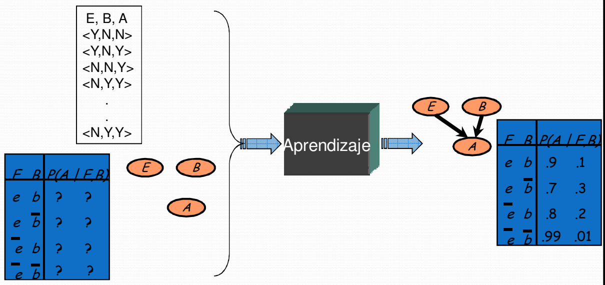


Estructura conocida, datos completos



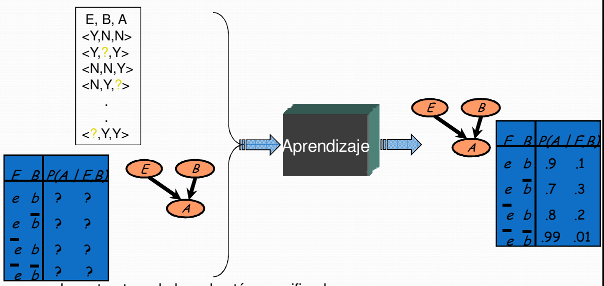
- La estructura de la red está especificada
  - Se necesitan estimar los parámetros (probabilidades)
- Los datos no contienen valores faltantes

Estructura desconocida, datos completos



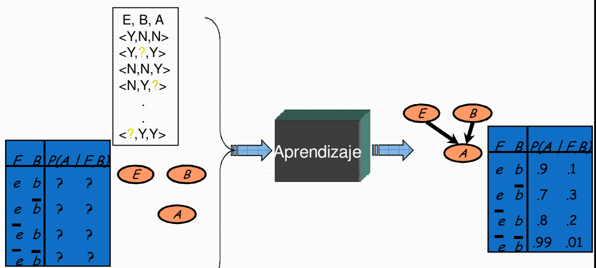
- La estructura de la red no está especificada
  - Se necesitan seleccionar los arcos y estimar los parámetros
- Los datos no contienen valores faltantes

Estructura conocida, datos incompletos



- La estructura de la red está especificada
- Los datos contienen valores faltantes
  - Se necesitan considerar asignaciones a los valores faltantes

Estructura desconocida, datos incompletos



- La estructura de la red no está especificada
  - Se necesitan seleccionar los arcos y estimar los parámetros
- Los datos contienen valores faltantes
  - Se necesitan considerar asignaciones a los valores faltantes

Aprendizaje de Redes de Bayes

- Aprendizaje paramétrico
  - Aprender los parámetros que describen el grado de dependencia entre las variables del dominio
- Aprendizaje estructural
  - Aprender las dependencias entre las variables que intervienen en el problema

Aprendizaje paramétrico

- Aprendizaje Adaptativo
  - Se utiliza cuando se quieren adaptar todas o algunas de las tablas de probabilidad condicional a partir de un nuevo conjunto de datos.
- Aprendizaje Batch
  - Se utiliza cuando se quieren generar las tablas de probabilidad condicional a partir de los datos. Ej: EM (Expectation Maximization)

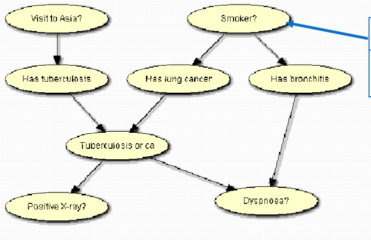
Aprendizaje adaptativo

- La adaptación es el proceso de refinar las probabilidades condicionales especificadas en la Red de Bayes teniendo en cuenta los resultados de experimentos o casos reales.
- Se utilizan tablas de experiencia y/o tablas de desvanecimiento.

Aprendizaje adaptativo

- Un paso del proceso de aprendizaje adaptativo consiste de:
  - ingresar evidencia,
  - propagar la evidencia,
  - actualizar (adaptar) las tablas de probabilidad condicional y las tablas de experiencia

Aprendizaje adaptativo con tablas de experiencia



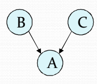
	Yes	No
Cantidad de experiencias	10	5

P(Smoker=yes) = #yes)/ (#yes + #no) = 10 / 15 = 0.667

P(Smoker=no) = #no)/ (#yes + #no) = 5 / 15 = 0.333

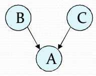
Fractional Updating

- Utiliza tablas de experiencia
- Suposiciones
  - Independencia Global
  - Independencia Local
- Sean:
  - $p(A|b_i,c_j)=(x_1,x_2)$
  - (n_1,n_2) las experiencias de A
  - $s = n_1 + n_2$ cantidad total de experiencias
  - $p(A\,|\,b_i,c_j)=(\frac{n_1}{s},\frac{n_2}{s})$

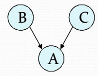


Fractional Updating

- Caso 1
  - Tenemos una nueva experiencia e con:
    - $B=b_i$,
    - $C=c_j$
    - $A=a_1$
  - Luego,
    - $n_1=n_1+1$
    - $s=s+1$
    - $p(A \mid b_i, c_j) = (\frac{n_1+1}{s+1}, \frac{n_2}{s+1})$

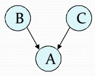


Fractional Updating



- Caso 2
  - Tenemos una nueva experiencia e con:
    - $B=b_i$,
    - $C=c_j$
    - de A sólo sabemos $p(A \mid e) = p(A \mid b_i, c_j, e) = (y_1, y_2)$
  - Luego,
    - $n_k=n_k+y_k$
    - $s=s+1$
    - $x_k = \frac{n_k + y_k}{s+1}$

Fractional Updating



- Caso general
  - Tenemos una nueva experiencia e con:
    - $p(b_i, c_j \mid e) = z$
    - $p(A \mid b_i, c_j, e) = (y_1, y_2)$
  - Luego,
    - $n_k=n_k+z.y_k$
    - $s=s+z$
    - $x_k = \frac{n_k + y_k.z}{s+z}$

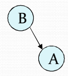
Fractional Updating

- Ejemplo

	B	
	true	false
true	0.6	0.3
false	0.4	0.7
Experiences	10	10

S=20

n_1 n_2



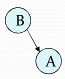
Como P(A=true|B=true) = 0.6
Tenemos
•N(A=true|B=true) = 0.6 * 10 = 6
•N(A=true|B=false)= 0.3 * 10 = 3
El total de las cuentas para la variable A:
•N(true) = 6 + 3 = 9
•N(false) = 4 + 7 = 11.

Fractional Updating

$p(b_i \mid e) = z$

$p(A \mid b_i, e) = (y_1, y_2)$

$x_k = \frac{n_k + y_k.z}{s+z}$



Después de una experiencia con:
• A=true
• B=true
Obtenemos

$p(A=true \mid B=true) = \frac{6+1.0*1.0}{11} = 0.63$

$p(A=false \mid B=true) = \frac{4+0.0*1.0}{11} = 0.37$

$p(A=true \mid B=false) = \frac{3+1.0*0.0}{10} = 0.3$

$p(A=false \mid B=false) = \frac{7+0.0*0.0}{10} = 0.7$

	B	
	true	false
true	0.63	0.3
false	0.37	0.7
Experiences	11	10

Fractional Updating

$p(b_i \mid e) = z$

$p(A \mid b_i, e) = (y_1, y_2)$

$x_k = \frac{n_k + y_k.z}{s+z}$

Si solo tenemos evidencia de A=true

Obtenemos

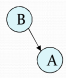
p(B=true|e)

$p(A=true \mid B=true) = \frac{6+1.0*0.67}{10.67} = 0.625$

$p(A=false \mid B=true) = \frac{4+0.0*0.67}{10.67} = 0.375$

$p(A=true \mid B=false) = \frac{3+1.0*0.33}{10.33} = 0.32$

$p(A=false \mid B=false) = \frac{7+0.0*0.33}{10.33} = 0.68$

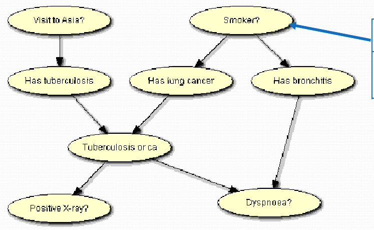


	B	
	true	false
true	0.625	0.32
false	0.375	0.68
Experiences	10.67	10.33

Aprendizaje adaptativo con tablas de desvanecimiento

- A veces, las observaciones viejas no son tan importantes como las nuevas en el proceso de adaptación. Entonces, tenemos que “desaprender” u “olvidar” algunas de ellas.
- Las observaciones nuevas son más importantes que las viejas, y por lo tanto se les debe dar más peso durante la adaptación.
- Se introduce un factor de desvanecimiento, que indica la tasa con la cual se olvidan las observaciones previas
  - 0 no hay adaptación
  - 1 no hay desvanecimiento.

Aprendizaje adaptativo con tablas de desvanecimiento



	Yes	No
Cantidad de experiencias	10	5

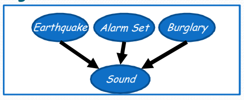
P(Smoker=yes) = #yes)/ (#yes + #no) = 10 / 15 = 0.667

P(Smoker=no) = #no)/ (#yes + #no) = 5 / 15 = 0.333

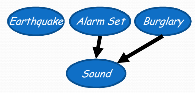
Aprendizaje adaptativo con tablas de desvanecimiento

- Smoker?=yes
- Fading factor(s) = 0.5
- #(Smoker=yes) = factor * #(yes-previos) + 1= 0.5 * 10 + 1 = 6
- #(Smoker=no) = factor * #(no-previos) = 0.5 * 5 = 2.5
- #(experiencias) = 6 + 2.5 = 8.5
- P(Smoker?=yes) = 6 / 8.5 = 0.7059
- P(Somker?=no) = 2.5 / 8.5 = 0.2941

Aprendizaje estructural

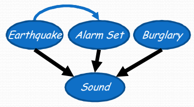


Si falta un arco



- No puede ser solucionado ajustando los parámetros
- Suposiciones erróneas acerca del dominio

Si sobra un arco



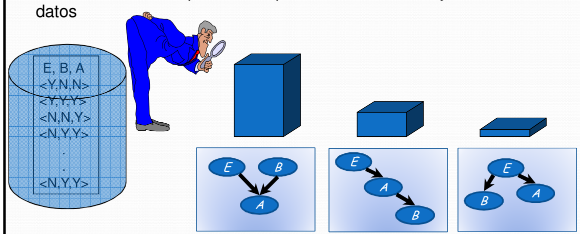
- Aumenta el número de parámetros a ser estimados
- Suposiciones erróneas acerca del dominio

Aprendizaje estructural

- Search and Score
  - Métrica de puntuación (Score)
    - Mide la calidad de la estructura gráfica
    - Se utiliza para decidir cual es la mejor solución
  - Algoritmo de búsqueda
    - Selecciona un pequeño subconjunto de las posibles redes que tengan una calidad alta
    - Se selecciona la “mejor”

Basado en optimizaciones

Definir una función que evalúe qué tan bien se asemeja el modelo a los datos



Buscar una estructura que maximice esa función

Aprendizaje estructural

- Tests de independencia
  - Obtienen las dependencias condicionales y las independencias utilizando tests de independencias entre triplas de variables
  - Por ejemplo tests estadísticos

Algoritmo K2

- Score and Search
- Búsqueda Greedy
- Muy eficiente
- Asume que las variables están ordenadas
- Comienza con una red sin arcos
- En cada iteración, para cada variable X_i, el nodo cuyo orden es previo a X_i y que incrementa la calidad de la red se agrega a $=Pa(X_i)$
- Se repite el proceso hasta que
  - No hay incremento en la calidad de la red
  - Se llega a una red completa
- La cantidad de padres puede limitarse

Algoritmo B

- Score and Search
- No requiere un ordenamiento previo de las variables
  - Controlar ciclos
- Comienza con un conjunto vacío de padres
- En cada iteración agrega un arco tal que
  - No produce ciclos
  - Maximiza la calidad de la red
- El proceso se detiene cuando
  - No se incrementa la calidad de la red, o
  - Se llega a una red completa

Algoritmo PC

- Tests de Independencia
- Dos fases
  - Construcción del esqueleto
  - Encontrar las direcciones de los arcos mediante tests de independencia
  - Orientar el resto de los arcos sin producir ciclos

Algoritmo PC

- Construcción del esqueleto
  - Comienza con un grafo no dirigido totalmente conectado
  - Mediante tests de independencia va eliminando arcos que no cumplen la siguiente condición:
    - Dos nodos X e Y están conectados si y sólo si no existe un subconjunto Sxy del conjunto de vértices V tal que I(X,Y|Sxy) sea verdadero

Algoritmo PC

Comienza con un grafo completo no dirigido *gp*

$i = 0$

Repeat

    For each $X \in V$

        For each $Y \in ADJ_X$

            Determinar si existe $S \subseteq ADJ_X - \{Y\}$ *con* $|S| = i$ *y* $I(X,Y|S)$

            Si este conjunto existe

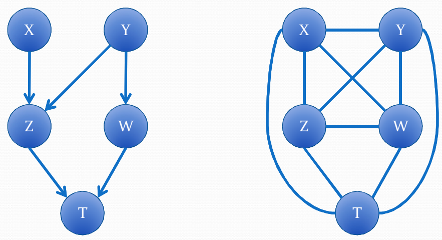
                $S_{XY} = S$

                Eliminar el arco X-Y

            $i = i+1$

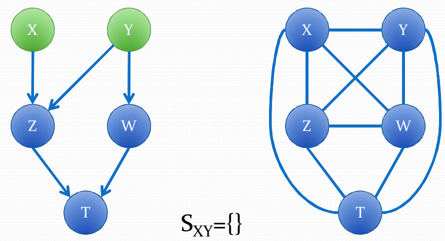
Until $|ADJ_X| \leq i; \ \forall X$

Algoritmo PC



Algoritmo PC

SI

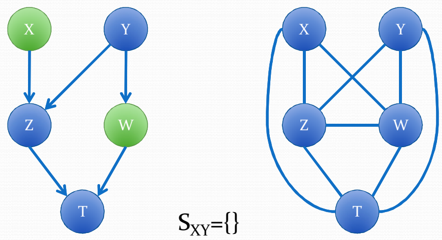


S_{XY}={}

i=0

Algoritmo PC

SI

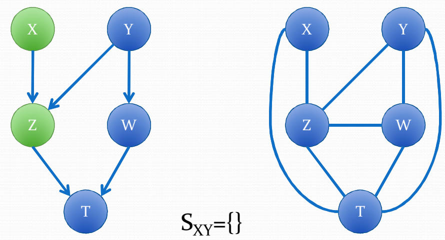


S_{XY}={}
S_{XW}={}

i=0

Algoritmo PC

NO

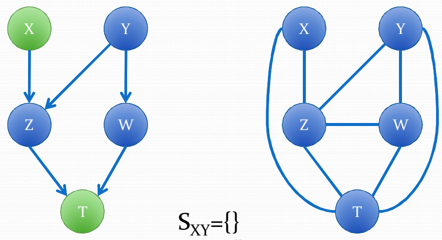


S_{XY}={}
S_{XW}={}

i=0

Algoritmo PC

NO

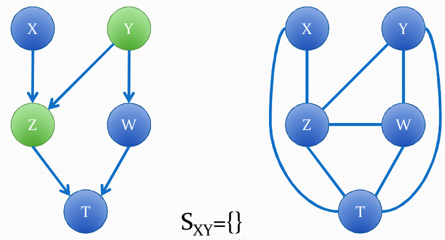


S_{XY}={}
S_{XW}={}

i=0

Algoritmo PC

NO

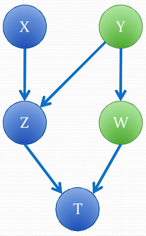


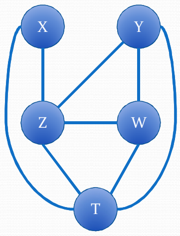
S_{XY}={}
S_{XW}={}

i=0

Algoritmo PC

NO



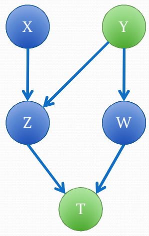


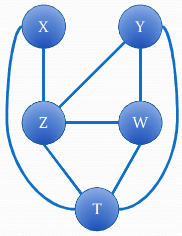
S_{XY}={}
S_{XW}={}

i=0

Algoritmo PC

NO



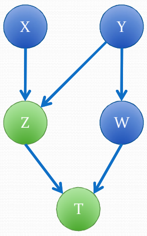


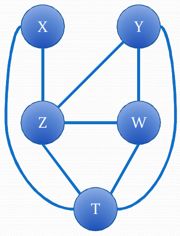
S_{XY}={}
S_{XW}={}

i=0

Algoritmo PC

NO



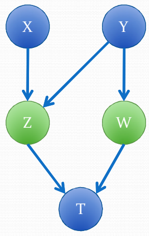


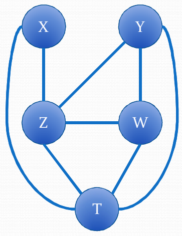
S_{XY}={}
S_{XW}={}

i=0

Algoritmo PC

NO



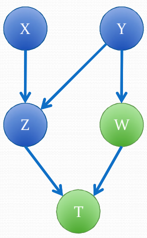


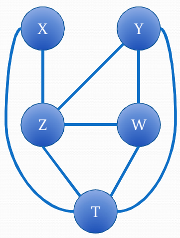
S_{XY}={}
S_{XW}={}

i=0

Algoritmo PC

NO



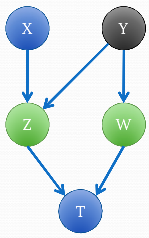


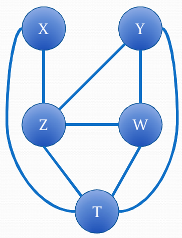
S_{XY}={}
S_{XW}={}

i=0

Algoritmo PC

SI

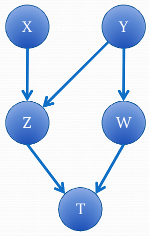


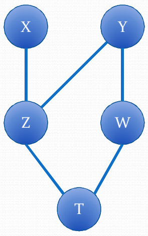


S_{XY}={}
S_{XW}={}
S_{ZW}={Y}

i=1

Algoritmo PC





S_{XY}={}
S_{XW}={}
S_{ZW}={Y}
S_{XT}={Z,W}
S_{YT}={Z,W}

No hay nodos con más de 3 adyacentes

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo PC

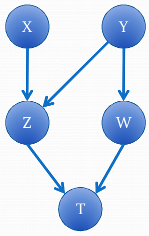
- Encontrar las direcciones de los arcos mediante tests de independencia (tomando las variables de a tres)
  - Para cada variable Z con dos adyacentes X e Y comprueba que no Z no esté en Sxy para conectar a X e Y como padres de Z

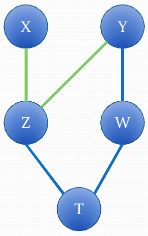
Y — Z — X

Y → Z ← X

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo PC



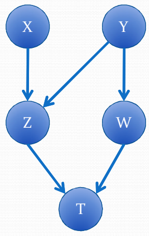


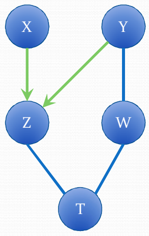
Z$\notin$ S_{XY}

S_{XY}={}
S_{XW}={}
S_{ZW}={Y}
S_{XT}={Z,W}
S_{YT}={Z,W}

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo PC



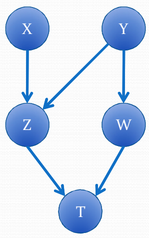


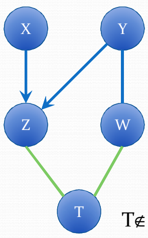
Z$\notin$ S_{XY}

S_{XY}={}
S_{XW}={}
S_{ZW}={Y}
S_{XT}={Z,W}
S_{YT}={Z,W}

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo PC



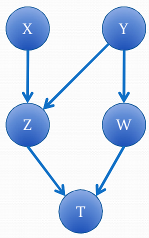


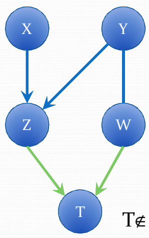
T$\notin$ S_{ZW}

S_{XY}={}
S_{XW}={}
S_{ZW}={Y}
S_{XT}={Z,W}
S_{YT}={Z,W}

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo PC



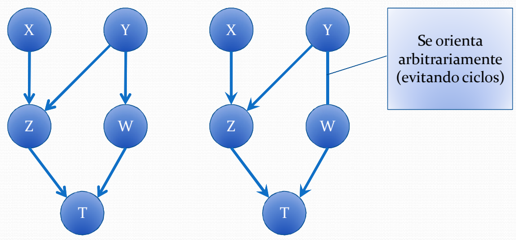


T$\notin$ S_{ZW}

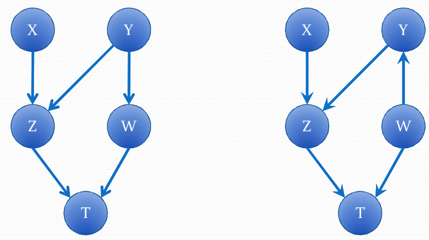
S_{XY}={}
S_{XW}={}
S_{ZW}={Y}
S_{XT}={Z,W}
S_{YT}={Z,W}

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo PC



Algoritmo PC



Herramientas

- Hugin: http://www.hugin.com/Products_Services/Products/Demo/
- Bayesware: http://www.bayesware.com/
- JavaBayes: http://www-2.cs.cmu.edu/~javabayes/
- Página de Finn Jensen: http://www.cs.aau.dk/~fvj

Modelos de Markov

Agenda

- Introducción
- Redes de Bayes
- Modelos de Markov
  - Introducción
  - Cadenas de Markov
  - Modelos Ocultos de Markov

Introducción

- Modelar dependencias entre símbolos de una secuencia
  - Temporales:
    - En reconocimiento del habla: fonemas en una palabra, palabras en una oración (reconocimiento sintáctico)
  - Espaciales
    - Reconocimiento de escritura
    - Secuencias de genes

Procesos aleatorios de Markov

- Una secuencia aleatoria tiene la propiedad de markov si su distribución de probabilidades está determinada únicamente por el estado actual. Un proceso aleatorio que cumple con esta propiedad se llama un proceso aleatorio de Markov
- Cuando los estados son observables → cadenas de Markov
- Cuando los estados no son observables → modelos ocultos de markov (HMM, hidden Markov models)

Regla de Bayes

$$P(q_t, q_{t-1}, \ldots, q_1) = P(q_t \mid q_{t-1}, \ldots q_1) P(q_{t-1}, \ldots q_1)$$

$$P(q_t, q_{t-1}, \ldots q_1) = P(q_t \mid q_{t-1}, \ldots q_1) P(q_{t-1} \mid q_{t-2}, \ldots q_1) P(q_{t-2}, \ldots q_1)$$

$$P(q_t, q_{t-1}, \ldots q_1) = P(q_1) \prod_{i=2}^{t} P(q_i \mid q_{i-1}, \ldots q_1)$$

Propiedad de Markov

$$P(q_i \mid q_{i-1}, \ldots q_1) = P(q_i \mid q_{i-1}) \ \ for\ \ i > 1$$

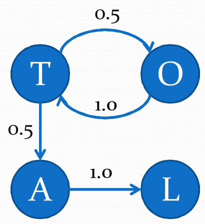
$$P(q_t, q_{t-1}, \ldots q_1) = P(q_1) \prod_{i=2}^{t} P(q_i \mid q_{i-1}) = P(q_1) P(q_2 \mid q_1) \ldots P(q_t \mid q_{t-1})$$

Cadenas de Markov

- Pueden verse como un grafo de transición de estados con probabilidades en sus arcos
- Ejemplo: Generación de Texto Aleatorio
  - Un nodo por cada letra única observada
  - Si luego de la letra X observo la letra Y agrego un arco de X a Y
    - Si el arco ya existe incremento la cantidad de observaciones asociada al arco

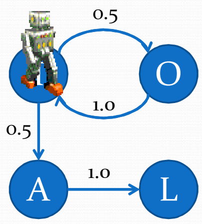
Cadenas de Markov

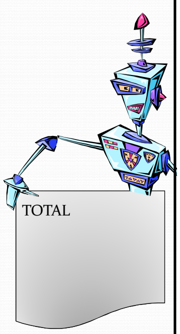
TOTAL



Cadenas de Markov

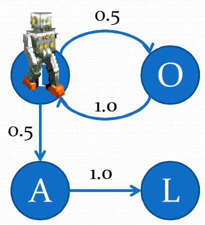
TOTAL

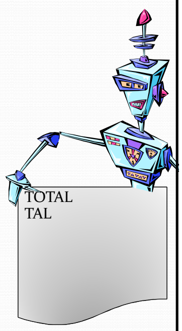




Cadenas de Markov

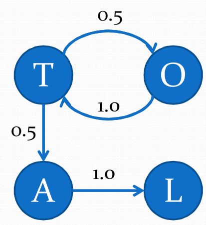
TOTAL

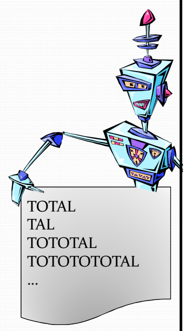




Cadenas de Markov

TOTAL





Cadenas de Markov

- Texto aleatorio generado a partir de un modelo entrenado con el primer capítulo de “100 años de soledad” de Gabriel García Marquez

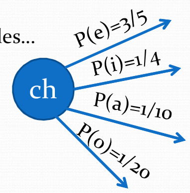
Mapl idoca **Uno** domen quespiate mé s abrdíaton quadelaros **del y en** ióra púllade **y** añosar só. Ar e Pon s tóntun llose fase co **tus** pe fo aciguela tr **de** paquros losintun. oceton deros **picaras** ro. pigeltexpa Arrcón asadosen ca híracugidend madel cos **pelo** ca hamuy cuncte prera letrapuina venco sosué gifuas **y** estiogidire **y** ple, peno do dociemanó **y** den matrones s Arsélades, lelanelo lacides Jombidorentió lla caquétrotan **de** ciechitevo, pa e tró ulega teilizaban ríaden zopen desellaundo u ca tioroserar vorasun plara **en fue**, dempa teresen y umus mpio **y sedes** s s ezordel rugexhas darmabió.

Cadenas de Markov de orden N

- Extensión a las cadenas de Markov en las que cada estados representa una secuencia de N observaciones consecutivas
- N-gramas

ch se observa 100 veces, de las cuales...

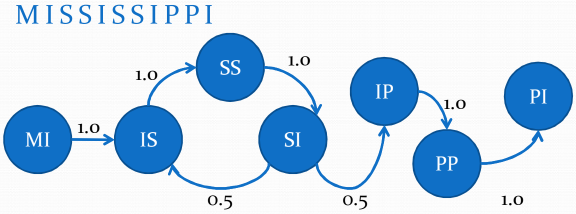
	Ocurrencias
che	60
chi	25
cha	10
cho	5



Cadenas de Markov de orden N

- N=2

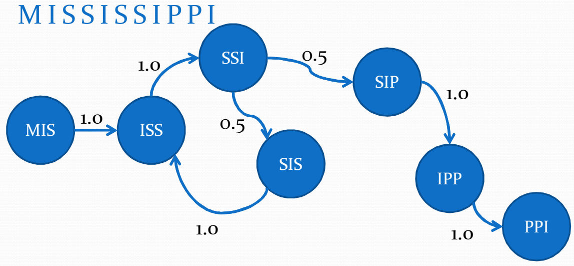
MISSISSIPPI



Cadenas de Markov de orden N

- N=3

MISSISSIPPI



Cadenas de Markov de orden N

- N=1
  - Modelo fiel a las frecuencias de cada par de letras
  - No representa bien el lenguaje natural

Mapl idoca Uno domen **qu**espiate mé s abrdíaton **qu**adelaros del y en ióra púllade y añosar só. Ar e Pon s tóntun llose fase co tus pe fo aciguela tr de pa**qu**ros losintun. oceton deros picaras ro. pigeltexpa Arrcón asadosen ca híracugidend madel cos pelo ca hamuy cuncte prera letrapuina venco sosué gifuas y estiogidire y ple, peno do dociemanó y den matrones s Arsélades, lelanelo lacides Jombidorentió lla ca**qu**étrotan de ciechitevo, pa e tró ulega teilizaban ríaden zopen desellaundo u ca tioroserar vorasun plara en fue, dempa teresen y umus mpio y sedes s s ezordel rugexhas darmabió.

Cadenas de Markov de Orden N

- Texto aleatorio generado N=2

Muca donios **y hombre** llas res pla plambregrimanteó **el la** sió entu **de** Salemo **fue** unción co crión. **Tendía en** sula ta chicalque **le** siez blena **que tras**, senterosagareto. —gruel res entorasanesbazuerasabe pán, enda **con de** in inagercamieropito **don** cuarecior siblimena tero **en** jó **te**. Fuedo paresegua pespurantespuen **dos** porodad **de la que** caba quentra denabrencestiminotescuyó quel mor losé Acturo, pen for **es ella** guanos mitadraratientón, condía **la** allado quen **se** logó **la** ba laritiva triblesala **sólos** habalcabiempor ere gría to borse allas laciasímisible
por pocaza **los**, so **ola la niños** cuaro **él y** lar espiendismo añonetro **una**, **que** cad fuerofocieres resa combrempuros parsu misiblocerma anoceraya elos no **a y** ejó desta **una** tión.

Cadenas de Markov de Orden N

- Texto aleatorio generado N=3

Muchetextremecitano **conveniencia**. **Cuando por** acora pre hablancie **el** rudiente
el pusierantado **poblado pago y aumentar** animaginación, aza des, examillos
más gigaciera suellevó **los que** inte **la viajes rayos** imales **hacedor y el** términuto pueses siles expedas mocha **que la** tanos **ser en que vez** lles, **la como la más pase lo únicos**, **romera lado José Arcadiosas**,
antes, **al** realejo **en que de dejó** todidos tre distilado **infinito** desco, **a le se para ver** bolada alquídea.

Cadenas de Markov de Orden N

- Texto aleatorio generado N=4

Muchos y salambió **un chaleco de había burla como el pasó la resto de** aban su **fue ser prendedor** condo **una buena de las casa** quimisterios,
José Arcadio Buendía, **sino** esentó **como un en aquel tambor**, **quedó recuerdo**
de un fugitivo en que en que sierrar **los** metalejos a puestaban **de una** tersas, **y el mundo sino** consolas, **y veían** desar **a la casa**, **cuyos sin comer Aureliano**, **conjuro de el** esperso **de piedras**.

Cadenas de Markov de orden N

- N = 3-4
  - Comienzan a aparecer palabras más reconocibles
  - Palabras de muchas letras aparecen frecuentemente “pegadas” a otras palabras
- Cuanto más extenso es el texto de entrenamiento, más variedad habrá en la salida

Cadenas de Markov de Orden N

- Entrenamiento con palabras completas (N=1)
- Texto generado:

Se había que construyó trampas y sus iniciales encima con una insolación por mares incógnitos, visitar territorios más emprendedor que se hundían en las maravillas del mundo.
Era una explicación inmediata, José Arcadio Buendía, la resonancia hueca de helechos y entonces una truculenta demostración con un áspero sabor de fusilamiento, el interior, y tambores y a prepararse para llevárselos a palmo a conocerlo, mantuvo puesta por los burros."
Así fue desde hacía mucho tiempo, porque el mundo se asomaba al caramelo vulgar que un pánico cuando llegó la judía.

Cadenas de Markov de Orden N

- Entrenamiento con palabras completas (N=4)
- Texto generado:

Consideraba como una burla de su travieso destino haber buscado el mar sin encontrarlo, al precio de sacrificios y penalidades sin cuento, y haberlo encontrado entonces sin buscarlo, atravesado en su camino como un obstáculo insalvable.
Úrsula no volvió a acordarse de la intensidad de esa mirada hasta un día en que el pequeño Aureliano, a la edad de tres años, entró a la cocina en el momento en que ella retiraba del fogón y ponía en la mesa una olla de caldo hirviendo.
Aquel olor mordiente quedaría para siempre en su memoria, vinculado al recuerdo de Melquíades.

Modelos de Markov

- Conjunto de estados: $\{s_1, s_2, \ldots, s_N\}$
- El proceso se mueve de un estado a otro generando una secuencia de estados $s_{i1}, s_{i2}, \ldots, s_{ik}, \ldots$
- Propiedad de Markov: la probabilidad de cada estado siguiente depende sólo del estado previo

$$P(s_{ik} \mid s_{i1}, s_{i2}, \ldots, s_{ik-1}) = P(s_{ik} \mid s_{ik-1})$$

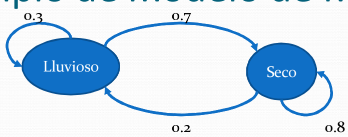
- Para definir un modelo de Markov deben especificarse las siguientes probabilidades:

$$\pi_i = P(s_i) \qquad a_{ij} = P(s_i \mid s_j)$$

Probabilidad del estado inicial — Probabilidad de transición de estados

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Ejemplo de modelo de Markov



- Dos posibles estados: 'Lluvioso' y'Seco'.
- Probabilidades de Transición:

P('Lluvioso'|'Lluvioso')=0.3 , P('Seco'|'Lluvioso')=0.7 ,
P('Lluvioso'|'Seco')=0.2, P('Seco'|'Seco')=0.8

- Probabilidades iniciales:

P('Lluvioso')=0.4 P('Seco')=0.6

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Probabilidad de una secuencia

- Por la probabilidad de Markov:

$$\begin{aligned} P(s_{i1}, s_{i2}, \ldots, s_{ik}) &= P(s_{ik} \mid s_{i1}, s_{i2}, \ldots, s_{ik-1}) P(s_{i1}, s_{i2}, \ldots, s_{ik-1}) \\ &= P(s_{ik} \mid s_{ik-1}) P(s_{i1}, s_{i2}, \ldots, s_{ik-1}) = \ldots \\ &= P(s_{ik} \mid s_{ik-1}) P(s_{ik-1} \mid s_{ik-2}) \ldots P(s_{i2} \mid s_{i1}) P(s_{i1}) \end{aligned}$$

P({'Seco','Seco','Lluvioso','Lluvioso'}) =
P('Lluvioso'|'Lluvioso') P('Lluvioso'|'Seco')
P('Seco'|'Seco') P('Seco') =
0.3*0.2*0.8*0.6 = 0.0288

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Underflow

- El producto de muchas probabilidades entre 0 y 1 puede resultar en underflow
- Dado que log(xy) = log(x) + log(y), es mejor realizar la suma del logaritmo de las probabilidades que multiplicarlas
- El modelo con el mayor puntaje final de probabilidad logarítmica sigue siendo la más probable

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Hipótesis del máximo a posteriori

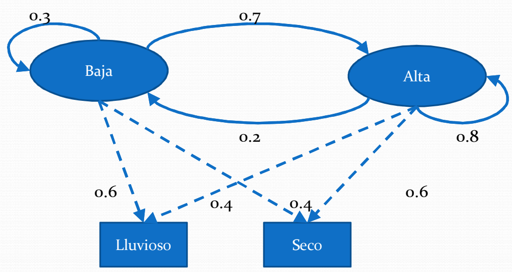
$$\begin{aligned} h_{MAP} &\equiv \arg\max_{h \in H} P(h \mid D) \\ &= \arg\max_{h \in H} \frac{P(D \mid h) P(h)}{P(D)} \\ &= \arg\max_{h \in H} P(D \mid h) P(h) \end{aligned}$$

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Modelos Ocultos de Markov (HMM)

- Conjunto de estados: $\{s_1, s_2, \ldots, s_N\}$
- El proceso de mueve de un estado a otro generando una secuencia de estados:
  $s_{i1}, s_{i2}, \ldots, s_{ik}, \ldots$
- Propiedad de Markov: La probabilidad de cada estado depende solo del estado actual $P(s_{ik} \mid s_{i1}, s_{i2}, \ldots, s_{ik-1}) = P(s_{ik} \mid s_{ik-1})$
- Los estados **no son visibles**, pero cada estado genera aleatoriamente una de las posibles observaciones **(estados visibles)** $\{v_1, v_2, \ldots, v_M\}$
- Para definir un modelo oculto de Markov, se deben especificar las siguientes probabilidades:
  - Matriz de probabilidades transición $A=(a_{ij})$, $a_{ij}= P(s_i \mid s_j)$
  - Matriz de probabilidades de observación $B=(b_i(v_m))$, $b_i(v_m) = P(v_m \mid s_i)$
  - Vector de probabilidades iniciales $\pi=(\pi_i)$, $\pi_i = P(s_i)$
- Los modelos quedan definidos por la tripla $M=(A, B, \pi)$

Ejemplo de HMM

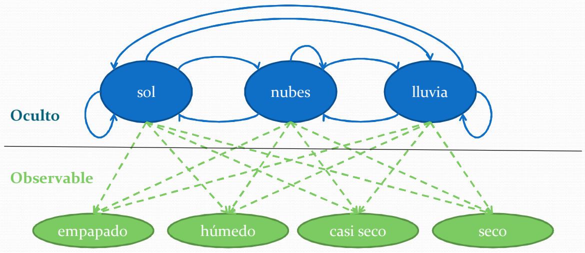


Ejemplo de HMM

- Dos estados correspondientes a la presión atmosférica:
  - 'Alta' y 'Baja'
- Dos posibles observaciones:
  - 'Lluvioso' y 'Seco'.
- Probabilidades de Transición:
  - P('Baja'|'Baja')=0.3 , P('Alta'|'Baja')=0.7 , P('Baja'|'Alta')=0.2, P('Alta'|'Alta')=0.8
- Probabilidades de Observación:
  - P('Lluvioso'|'Baja')=0.6 , P('Seco'|'Baja')=0.4 , P('Lluvioso'|'Alta')=0.4 , P('Seco'|'Alta')=0.3 .
- Probabilidades iniciales:
  - P('Baja')=0.4 , P('Alta')=0.6 .

Ejemplo de HMM

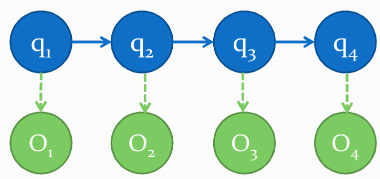
- Predicción del tiempo de acuerdo al estado de un alga



Ejemplo de HMM

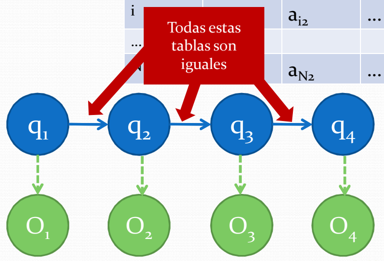
	Empapado	Húmedo	Casi seco	Seco
Sol	0,60	0,20	0,15	0,05
Nubes	0,25	0,25	0,25	0,25
Lluvia	0,05	0,10	0,35	0,50

Representación de un HMM como Red de Bayes



Representación ... Red de ...

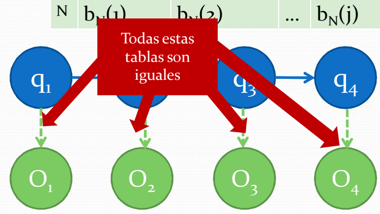
i	$P(q_{t+1}=S_1 \mid q_t=S_i)$	$P(q_{t+1}=S_2 \mid q_t=S_i)$		$P(q_{t+1}=S_j \mid q_t=S_i)$		$P(q_{t+1}=S_N \mid q_t=S_i)$
1	a_{11}	a_{12}	...	a_{1j}	...	a_{1N}
2	a_{21}	a_{22}	...	a_{2j}	...	a_{2N}
3	a_{31}	a_{32}	...	a_{3j}	...	a_{3N}
...						
i		a_{i2}	...	a_{ij}	...	a_{iN}
...						
		a_{N2}	...	a_{Nj}	...	a_{NN}



$a_{ij} = P(q_{t+1}=s_j \mid q_t=s_i)$

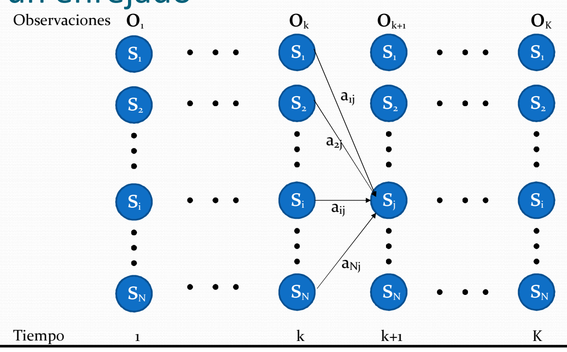
Re... Re...

i	$P(O_t=1 \mid q_t=S_i)$	$P(O_t=2 \mid q_t=S_i)$		$P(O_t=j \mid q_t=S_i)$		$P(O_t=N \mid q_t=S_i)$
1	$b_1(1)$	$b_1(2)$	...	$b_1(j)$	...	$b_1(N)$
2	$b_2(1)$	$b_2(2)$	...	$b_2(j)$	...	$b_2(N)$
3	$b_3(1)$	$b_3(2)$	...	$b_3(j)$	...	$b_3(N)$
...						
i	$B_i(1)$	$b_i(2)$	...	$b_i(j)$	...	$b_i(N)$
...						
N	$b_N(1)$	$b_N(2)$	...	$b_N(j)$	...	$b_N(N)$



$b_i(j) = P(O_t=j \mid q_t=s_i)$

Representación de un HMM como un enrejado



Ejemplo: Casino tramposo

Un casino tiene dos dados:

- Dado "justo"
  $P(1) = P(2) = P(3) = P(5) = P(6) = 1/6$
- Dado cargado
  $P(1) = P(2) = P(3) = P(4) = P(5) = 1/10$
  $P(6) = 1/2$

El casino alterna entre el dado justo y el cargado una vez cada 20 turnos

Problemas a resolver con HMM

- Evaluación
  - Dado un HMM $M=(A, B, \pi)$ y una secuencia de observaciones $O=o_1\, o_2 \ldots o_K$, calcular la probabilidad de que el modelo M haya generado la secuencia O

Evaluación

Dada una secuencia de tiradas del casino

1245526462146146136136661664661636616366163616515615115146123562344

¿Cuan probable es esta secuencia, dado nuestro modelo de como opera el casino?

Este es el problema de **EVALUACIÓN** en HMMs

Problemas a resolver con HMM

- Decodificación o inferencia
  - Dado un HMM M=(A, B, π) y una secuencia de observaciones O=o_1 o_2 … o_K , calcular la secuencia de estados ocultos S_i que más probablemente produjo la secuencia de observaciones O

Decodificación

Dada una secuencia de tiradas del casino

1245526462146146136136661664661636616366163616515615115146123562344

¿Que partes de la secuencia fueron generadas por el dado cargado y cuales por el dado justo?

Este es el problema de **DECODIFICACIÓN** en HMMs

Problemas a resolver con HMM

- Aprendizaje
  - Dado un conjunto de secuencias de observaciones de entrenamiento O=o_1 o_2 … o_K, y la estructura general de HMM (cantidad de estados ocultos y cantidad de estados visibles), determinar los parámetros del HMM M=(A, B, π) que mejor se ajustan a los datos de entrenamiento.

Aprendizaje

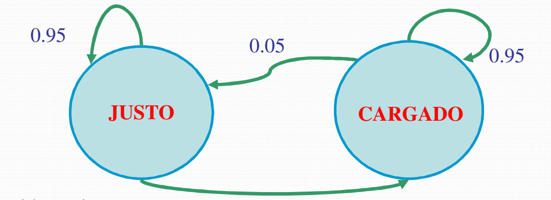
Dada una secuencia de tiradas del casino

1245526462146146136136661664661636616366163616515615115146123562344

¿Cómo está cargado el dado cargado? ¿Cada cuanto cambia el casino entre el dado cargado y el justo?

Este es el problema de **APRENDIZAJE** en HMMs

El HMM del casino tramposo



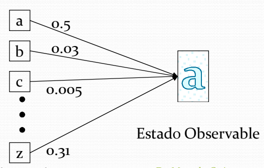
P(1|Justo) = 1/6
P(2|Justo) = 1/6
P(3|Justo) = 1/6
P(4|Justo) = 1/6
P(5|Justo) = 1/6
P(6|Justo) = 1/6

P(1|Cargado) = 1/10
P(2|Cargado) = 1/10
P(3|Cargado) = 1/10
P(4|Cargado) = 1/10
P(5|Cargado) = 1/10
P(6|Cargado) = 1/2

Ejemplo: reconocimiento de palabras

- Se asume que los caracteres están separados

- El reconocedor de caracteres da como salida la probabilidad de que una imagen corresponda a un caracter particular: P(caracter|imagen).



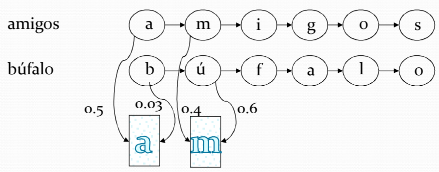
Ejemplo: reconocimiento de palabras

- Estados Ocultos del HMM: caracteres
- Estados Observables: imagenes de caracteres v_α
- Probabilidades de Observación:

$$B = (b_i(v_\alpha)) = (P(v_\alpha \mid s_i))$$

Ejemplo: reconocimiento de palabras

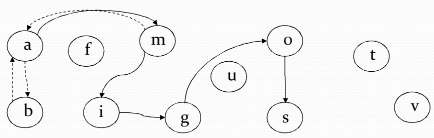
- Si se conoce el diccionario, es posible construir un modelo individual para cada palabra



- Problema de **evaluación**: reconocer la palabra en una imagen es equivalente a evaluar un conjunto de modelos.

Ejemplo: reconocimiento de palabras

- Podemos construir un HMM para **todas** las palabras
  - Estados ocultos: los caracteres de las palabras
  - Probabilidades de transición: calculadas a partir de modelos del lenguaje
  - Observaciones y probabilidades de observación iguales al ejemplo anterior



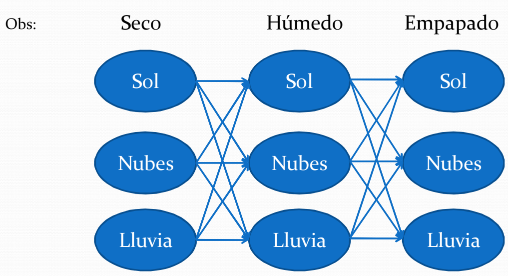
  - Problema de **decodificación**: encontrar la secuencia de estados ocultos que más probablemente generaron la imagen

Problemas a resolver con HMM

- Dada una secuencia de observaciones O1, O2,... , O_T

Problema	Algoritmo	Complejidad
Evaluación: Calcular P(O|M)	Forward-Backward	(TN^2)
Inferencia: Calcular Q*=argmax$_Q$P(Q|O,M)	Viterbi Decoding	(TN^2)
Aprendizaje: Calcular M*= argmax$_M$P(O|M)	Baum-Welch (EM)	(TN^2)

Algoritmo Forward



Algoritmo Forward

- Pr(Seco, Húmedo, Empapado|M) =

  Pr(seco, húmedo, empapado|sol, sol, sol) +

  Pr(seco, húmedo, empapado|sol, nubes, sol) +

  ...

  Pr(seco, húmedo, empapado|lluvia, lluvia, lluvia)

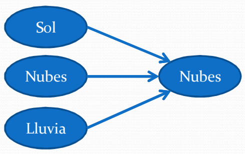
3^3 = 27 cálculos

¡¡Muy Costoso!!

Algoritmo Forward

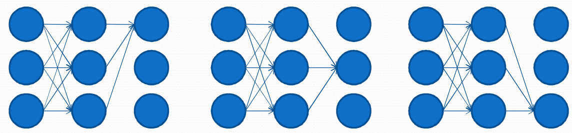
- Reducción de la complejidad
  - Probabilidad parcial de alcanzar un estado intermedio en el entramado

$Y_{k1}, Y_{k2}, \ldots, Y_{kT}$



Algoritmo Forward

- $\alpha_t(j)$= Pr(obs|estado oculto es j) ·
  Pr(todos los caminos al estado j en el tiempo t)



Algoritmo Forward

- En el tiempo t=1

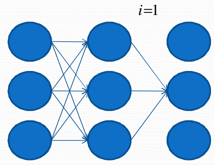
$$\alpha_1(j)=\pi(j)\ \Pr(\text{obs}|\text{estado j})$$

- En el tiempo t >1

$$\alpha_t(j)=\Pr(\text{todos los caminos a j})\ \Pr(\text{obs}|\text{estado j})$$

Algoritmo Forward

$$\alpha_{t+1}(j)=b_j(o_{t+1})\sum_{i=1}^{n}\alpha_t(j)a_{ij}$$

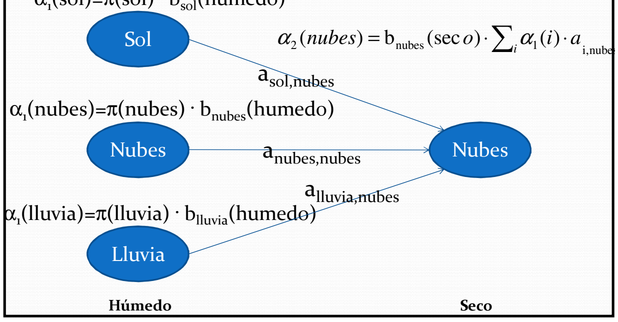


- Terminación:

$$\Pr(Obs\mid M)=\sum_{j=1}^{n}\alpha_T(j)$$

Algoritmo Forward

$\alpha_1(\text{sol})=\pi(\text{sol}) \cdot b_{\text{sol}}(\text{humedo})$

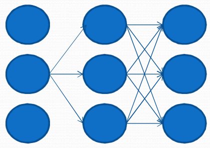


Algoritmo Backward

- Se define la variable β:

$$\beta_t(i)=P(o_{t+1},o_{t+2},\ldots,o_T\mid q_t=i,\lambda)$$

Algoritmo Backward



- Inicialización:

$$\beta_T(j) = 1$$

- Cálculo:

$$\beta_t(i) = \sum_{j=1}^{N} \beta_{t+1}(j) a_{ij} b_j(o_{t+1})$$

- Finalización:

$$p(O \mid \lambda) = \sum_{i=1}^{N} \pi(i) b_i(o_1) \beta_1(i) \qquad t = T-1,\ldots,1$$

Problemas a resolver con HMM

- Dada una secuencia de observaciones O1, O2,... , O_T

Problema	Algoritmo	Complejidad
Evaluación: Calcular P(O|M)	Forward- Backward	(TN^2)
Inferencia: Calcular Q*=argmax_QP(Q|O,M)	Viterbi Decoding	(TN^2)
Aprendizaje: Calcular M*= argmax_MP(O|M)	Baum-Welch (EM)	(TN^2)

Algoritmo de Viterbi

- Encontrar la secuencia más probable de estados ocultos, dada una secuencia de observaciones
- Ejemplo: etiquetar palabras en una oración con su función sintáctica

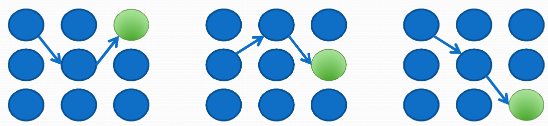
¿Estados observables? ¿Estados ocultos?

NS NP OD

Juan come naranjas

Algoritmo de Viterbi

- Probabilidad parcial δ de alcanzar un estado intermedio particular en el entramado
- δ representa la probabilidad del <u>camino más probable</u> a un estado
- Para cada estado final hay un camino más probable a ese estado



Algoritmo de Viterbi

- δ(i,t) es la máxima probabilidad de todas las secuencias que terminan en el estado i en el tiempo t
- En t=T tenemos las mejores probabilidades parciales y los mejores caminos parciales.
- El mejor camino global a T es el que lleva al estado con la mayor probabilidad

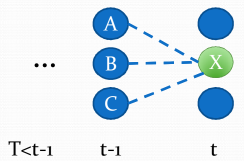
Algoritmo de Viterbi

- En t=1

$$\delta(1,i) = \pi(i) \cdot b_i(o_1)$$

Algoritmo de Viterbi

- En t>1



Pr(x en T=t) = $\max_{i=A,B,C}$ Pr(i en T=t-1) Pr(X|i) Pr(O_t|X)

$$\delta(t,i) = \max_{j}(\delta(t-1,j)\cdot a_{j,i})\cdot b_i(o_t)$$

Algoritmo de Viterbi

- En cada tiempo t, recordamos el estado j que generó δ(t,i)

$$\phi(t,i) = \max_{j}\delta(t-1,j)\cdot a_{j,i}$$

- Se reconstruye el camino utilizando los punteros ϕ hasta el tiempo t=1

Algoritmo de Viterbi

$$P^* = \max_{1\leq i\leq N}\delta_T(i)$$

P* da la probabilidad de los mejores los estados

$$q_T^* = \arg\max_{1\leq i\leq N}\delta_T(i)$$

q_T* is la secuencia de estados óptima
(Q* = {q1*,q2*,…,qT*})

Problemas a resolver con HMM

- Dada una secuencia de observaciones O1, O2,… , O_T

Problema	Algoritmo	Complejidad
Evaluación: Calcular P(O|M)	ForwardBackward	(TN^2)
Inferencia: Calcular Q*=argmax$_Q$P(Q|O,M)	Viterbi Decoding	(TN^2)
Aprendizaje: Calcular M*= argmax$_M$P(O|M)	Baum-Welch (EM)	(TN^2)

Aprendizaje

- Dada un conjunto de secuencias de observaciones $O=o_1 o_2 \ldots o_K$ y la estructura general del HMM (cantidadde estados ocultos y visibles), determinar los parámetros del modelo M=(A, B, π) que mejor se ajustan a los datos de entrenamiento, es decir, que maximice P(O | M) .
- Se utiliza el algoritmo de maximización de la esperanza para encontrar un máximo local de P(O | M): **algoritmo Baum-Welch**

Aprendizaje

- Si los datos de entrenamiento tienen información acerca de la secuencia de estados ocultos, los parámetros se estiman como:

$$a_{i,j} = P(s_i \mid s_j) = \frac{\#transiciones\ s_j \rightarrow s_i}{\#transiciones\ s_j \rightarrow x}$$

$$b_i(o_m) = P(o_m \mid s_i) = \frac{\#observaciones\ de\ o_m\ en\ s_i}{\#veces\ s_i}$$

Algorimo Baum-Welch

- Idea general:

$$a_{i,j} = P(s_i \mid s_j) = \frac{\#\,esperado\ de\ transiciones\ s_j \to s_i}{\#\,esperado\ de\ transiciones\ s_j \to x}$$

$$b_i(o_m) = P(o_m \mid s_i) = \frac{\#\,esperado\ de\ observaciones\ de\ o_m\ en\ s_i}{\#\,esperado\ de\ veces\ s_i}$$

$$\pi_i = P(s_i) = \#\,esperado\ de\ veces\ en\ s_i\ \ para\ t = 1$$

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo Baum-Welch

$$\xi_t(i,j) = P(q_t = s_i\ , q_{t+1} = s_j \mid o_1 o_2 \ldots o_T)$$

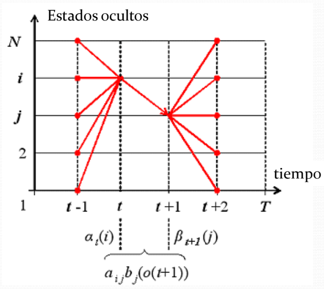
$$\xi_t(i,j) = \frac{P(q_t = s_i\ , q_{t+1} = s_j\ , o_1 o_2 \ldots o_T)}{P(o_1 o_2 \ldots o_T)} =$$

$$\frac{P(q_t = s_i\ , o_1 o_2 \ldots o_t)\ a_{ij}\ b_j(o_{t+1})\ P(o_{t+2} \ldots o_T \mid q_{t+1} = s_j)}{P(o_1 o_2 \ldots o_T)} =$$

$$\frac{\alpha_t(i)\ a_{ij}\ b_j(o_{t+1})\ \beta_{t+1}(j)}{\Sigma_i \Sigma_j\ \alpha_t(i)\ a_{ij}\ b_j(o_{t+1})\ \beta_{t+1}(j)} = \frac{\alpha_t(i)\ a_{ij}\ b_j(o_{t+1})\ \beta_{t+1}(j)}{\Sigma_i\, \alpha_t(i)\ \beta_t(i)}$$

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo Baum-Welch



Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo Baum-Welch

- Definimos $\gamma_t(i)$ como la probabilidad de estar en el estado i en el tiempo t dadas las observaciones

$$\gamma_t(i) = P(q_t = s_i \mid o_1 o_2 \ldots o_T) \qquad \gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j)$$

- $\Sigma_t\ \gamma_t(i)$ - cantidad esperada de veces que visito el estado *i*
- $\Sigma_t\ \xi_t(i,j)$ - cantidad esperada de transiciones $i \to j$

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Algoritmo Baum-Welch

- $\hat{\pi} = \gamma_1(i)$ la frecuencia esperada del estado i en t=1
- $\hat{a}_{ij} = \dfrac{\sum \xi_t(i,j)}{\sum \gamma_t(i)}$ número esperado de transiciones desde el estado i al estado j sobre la cantidad esperada de transiciones desde el estado i
- $\hat{b}_j(o_k) = \dfrac{\sum_{t,o_t=o_k} \gamma_t(j)}{\sum \gamma_t(j)}$ cantidad de veces que se estuvo en el estado j observando el símbolo k sobre la cantidad de veces que se estuvo en el estado j

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

Bibliografía

- Bayesian Networks and Decision Graphs - Finn V. Jensen and Thomas D. Nielsen - Springer Verlag - Second Edition - 2007
- A tutorial on hidden Markov models and selected applications in speech recognition - Lawrence R. Rabiner - Proceedings of the IEEE, Vol. 77, Nº2, pp. 257-286 - 1989

Dr. Marcelo G. Armentano - ISISTAN - UNICEN

¿Consultas?