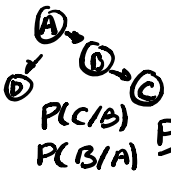
Tema 4: Redes Bayesianas

- **Razonamiento Probabilistico:**
- **Incertidumbre**: Confianza que tenemos de un suceso.
  - Se expresa como una probabilidad, que es un medidas de:
    - Proporción de veces en que algo es cierto.
    - Grado de creencia en que algo es cierto.
  - **Variable aleatoria**: Que pueden tomar valores en un dominio, el valor asociado es desconocido, pero podemos saber la probabilidad de cada valor posible.
  - **Espacio muestral Ω:** Todos los posibles resultados de un experimento aleatorio.
  - **Evento**: Subconjunto del espacio muestral, hay varios.
  - **Evento atómico:** Evento de un único elemento y define un único estado.
  - **Distribución de probabilidad**: Asigna a cada evento un valor que representa la probabilidad de que ocurra ese evento. Para hallar la probabilidad debemos tener datos de e. Se puede expresar como una vector formado por todas la probabilidades. P(e)
    - Toma valores entre 0 y 1, incluidos.
    - La suma de todas las probabilidades del evento es 1.
  - **Probabilidad de un evento no atómico:** Suma de las probabilidades de todos lo eventos atómicos que lo componen, se suman los posibles.
    - **P(A)=ΣP(e)**
  - **Teorema de la probabilidad total**: La suma de las probabilidades que tiene de que ocurra A condicionando con otro evento, por la probabilidad de ese evento. **P(A)=ΣP(A, B1,B2, ...)**
  - **Regla del producto: P(A/B)= P(A,B)/P(B)=P(AnB)/P(B)**
  - **Probabilidad a posteriori(condicional)**: Sobre una observación, tenemos evidencias. Esa evidencia modifica el conocimiento sobre el dominio.
    - **P(A|B)=P(AnB)/P(B)=αP(BnA)**
    - **Regla de la cadena: P(AnB)=P(A|B)P(B)**
    - **Comparar probabilidades condicionadas:** Cuando la B es la misma pero la A cambia, la suma de ambas probabilidades debe dar 1, por lo que dejamos como incógnita **α=1/P(B)** que es la **constante de normalización**. El que mayor x tenga es mas probable siendo x el termino que acompaña a α.
    - α permite también hallar P(B) cuando no se nos da y conocemos las condicionadas.
  - **Probabilidad a priori:** Probabilidad que algo ocurra cuando no tenemos información.
  - **Probabilidad conjunta:** Probabilidad de un conjunto de variables.
    - Se puede representar de manera tabular, en una tabla, pero en problemas reales no es viable ya que hay ciento o miles de variables. Hay que tener en cuenta el coste y tiempo de respuesta.
  - **Teorema de Bayes:** Se pueden calcular unas condicionales a través de otras.
    - **P(A/B)=(P(B/A)P(A))/P(B)=αP(B/A)P(A)**
  - **Independencia**: A y B lo son si y solo si la ocurrencia de uno de ellos no afecta a la ocurrencia del otro.
    - **P(A|B)=P(A)** La probabilidad de A no se ve afectada por B.
    - **P(A, B)=P(A)P(B)**
    - **Reducción de tamaño de distribución:** Eso implica algoritmo más eficiente y menos datos a especificar.
  - **Inferencia**: Calcular la probabilidad de eventos (probabilidad a posteriori) dada cierta evidencia. Se usa para:
    - **Predicción.**
    - **Diagnosis.**
    - **Clasificación.**
    - **Toma de decisiones:** Elegir acciones más útiles.

- **Redes Bayesianas:**
  - **Independencia condicional:** Cuando hay relaciones entre varias variables y están relacionadas, pero por medio de otra, que no es causa directa. Hay un intermediario, por lo que conociéndolo depende del intermediario, no del que esta alejado un nodo.

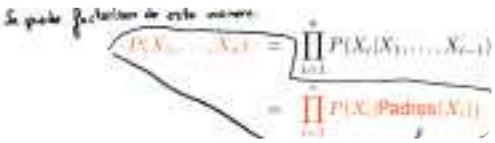


P(X,Y|Z)=P(Y|Z)P(X|Z) y también P(X|Y, Z)=P(X|Z)
Nos permite reducir el numero de parámetros, normalmente de exp. a lineal.
En las redes bayesianas se asume que hay variables ind. condicionalmente.

  - **Red Bayesianas:** Es un Grafo Aciclico Dirigido (DAG), en la que los nodos son las variables, los arcos indican causa directa. Cada nodo tiene una tabla de probabilidad condicional (CPT), indica la influencia de los nodos padre sobre el. Cuando no tiene es la probabilidad a priori.

Seguir uno con causal ser consistente.

    - Se puede expresar de forma compacta como la probabilidad de todos separados por comas, distribución de probabilidad conjunta, gracias a que se asume independencia condicional y después podemos factorizarla.



    - Es ir sacando términos, así:
      - P(A, B, C)= P(A|B,C)P(B,C)=...
    - La CPT tendrá 2^a filas en un nodo con a padres. Se pone la probabilidad de que sea true, el propio nodo, ya que de que sea false es la inversa 1-P()
    - Complejidad $O(n*2^b)$ n variables y b padres.
  - **Causalidad vs. Correlación:** Que numéricamente estén relacionados, no quiere decir que existan una relación directa, pueden existir variables intermedias. Fijarse si esto ocurre.
    - Correlación no implica causalidad, pero Causalidad implica correlación.

Poliarbol

  - **Inferencia**: Calcular la probabilidad a posteriori de una variable dada cierta evidencia.
    - Hay que **tener en cuenta**: La variable pregunta, las variables evidencia y las ocultas.
    - **Inferencia exacta por enumeración**:

1ºR. Producto
2ºR. Prob. total
3º Factorizar

      - 1. Se aplica la regla del producto. Donde α=1/(lo de la derecha) o 1/(suma αP's)
        - **P(A|B)=αP(A,B).**
      - 2. Después aplicamos la regla de la probabilidad total, OJO con los sumatorios y poder sacar variables para simplificar o que sume 1. En general las variables que no son ancestros de variables pregunta o variables evidencia son irrelevantes.
        - **αP(A,B)=αΣ P(A, B, C).**
      - 3. Lo de dentro del sumatorio se factoriza y operamos.
        - Las evidencias tendrá el valor fijo, pero las ocultas se contemplan con el sumatorio. Añade en forma de sumatorio las variables ocultas. Cuando no se indica si la variable pregunta es true o false, se hallan ambas posibilidades por separado, esto a su vez nos permite hallar α, suman 1 ambas prob.
      - El problema de este método son la variable ocultas que tienen complejidad exponencial.
      - Es eficiente para poliarboles, que como mucho haya un arco entre cada par de nodos. Esta estructura da lugar a una complejidad líneal.
      - No es eficiente cuando la estructura de la red no es un poliarbol, pasa de ser complejidad lineal a ser exponencial. En estos casos se usa la inferencia aproximada.

No Policiabel

- **Inferencia aproximada**: Se hace mediante estimaciones, para ello se hace un muestreo de la red Bayesiana, comenzando desde padres hasta los hijos. Muestrear, quiere decir que siguiendo la distribución de probabilidad sacamos un número aleatorio. Por ejemplo P(Nublado)=(0.5, 0.5) Sacamos un numero entre 0 y 1, si cae en la parte de la izquierda es true, si no false. 0-0.5 true y 0.5-1 false.
  - De esta manera sacamos que valor tiene cada variable y podemos centrarnos en ese caso concreto, y hallar su probabilidad como haríamos en el caso anterior.
  - **Muestreo directo:** La probabilidad de un evento se estima como el numero de casos del evento generados por muestreo dividido entre el numero total de casos muéstrales.

$$P(x_1, x_2, \ldots, x_n) \simeq \frac{\#\text{casos}(x_1, x_2, \ldots, x_n)}{\#\text{total casos}}$$

¿Cómo resolvemos preguntas sobre una distribución *a posteriori* $P(X/e)$?

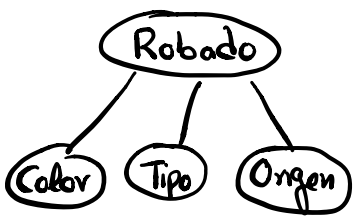
- Muestrear la distribución *a priori* que representa la red
- Rechazar los casos en los que la evidencia no es cierta
- Entonces

$$P(X/e) = \frac{\#\text{casos}(X, e)}{\#\text{casos}(e)} \simeq \frac{P(X, e)}{P(e)} = P(X/e)$$

- **Clasificador *Naïve Bayes***: Caso especial de red bayesiana con una estructura en la que de un nodo cuelgan el resto. El nodo raíz es del que deseo conocer la probabilidad, y los hijos son los atributos y son independientes entre ellos.
  - **Se puede resolver con** Bayes normal o mediante los métodos de inferencia.

- Se aplica el teorema de Bayes

$$p(c_i \mid a_1 \ldots a_n) = \frac{P(a_1 \ldots a_n \mid c_i) p(c_i)}{p(a_1 \ldots a_n)}$$



- Asume Independencia Condicional entre los atributos dada la clase:

$$P(a_1 \ldots a_n \mid c_i) = P(a_1 \mid c_i) \times \cdots \times P(a_n \mid c_i)$$

- En la mayoría de los casos esto no es verdad
- Pero el clasificador exhibe un comportamiento robusto

- Regla de clasificación: elegir la clase c_i que maximiza

$$p(c_i \mid a_1 \ldots a_n) = \alpha p(C = c_i) \prod_k P(A_k = a_k \mid C = c_i)$$

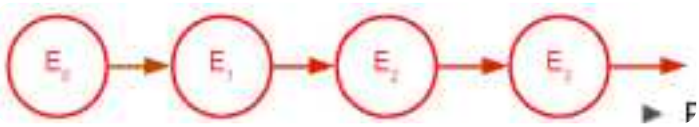
$$p(C = c_i) \prod_k P(A_k = a_k \mid C = c_i)$$

Análisis de frecuencias en los datos de entrenamiento:

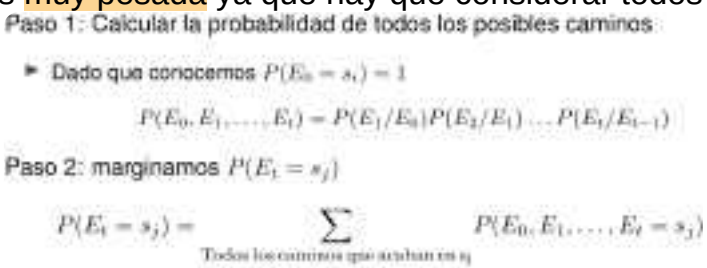
$$p(C = c_i) = \frac{\text{Número de ejemplos de clase } c_i}{\text{Número total de ejemplos}}$$

$$P(A_k = a_k \mid C = c_i) = \frac{\text{Número de ejemplos de clase } c_i \text{ con } A_k = a_k}{\text{Número de ejemplos de clase } c_i}$$

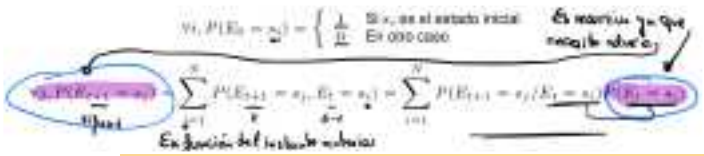
- **Razonamiento Probabilístico en el Tiempo:**
  - Hasta el momento hemos contemplado solo mundos estaticos, pero ahora consideramos el paso del tiempo. Cuando el mundo es no determinista, que hay varias posibles decisiones en cada paso, se emplea el procesos de decisión de Markov.
  - **Modelos de Markov:** Se representa con un grafo la probabilidad de las transiciones, estando en un tiempo t y yendo a t+1, el momento siguiente. El estado actual determina la distribución de probabilidad del estado siguiente. El momento inicial será E_0.
    - **Hipótesis de Markov:** Se asume que el estado actual solo depende del anterior, es condicionalmente independiente de los que no son el inmediato anterior.



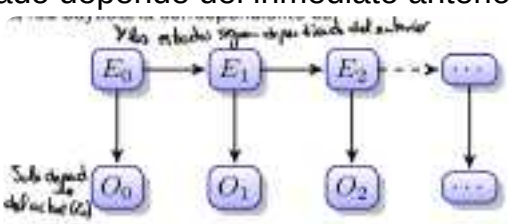
    - Se puede representar como una red bayesiana, en la que los estado que no dependen de otros tienen su probabilidad a priori y las que dependen de otro su tabla de probabilidad condicionada.
    - **Calcular la probabilidad:** Se hace como inferencia exacta. Lo primero es hacer la regla del producto, después hacer la probabilidad total con sumandos y finalmente factorizar. Pero esta opción es muy pesada ya que hay que considerar todos los posibles caminos anteriores.



    - **Programación dinámica:** Es una técnica/truco, que consiste en hacer una definición recursiva en función del instante anterior. Se comienza desde el final, y se van llamando unas a otras hasta el inicial. Algoritmo de Simulación hacia delante, que tiene complejidad lineal para un tiempo t.

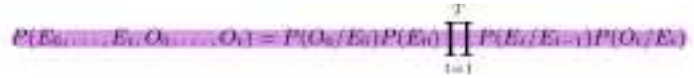


    - **Distribuciones estacionarias:** Normalmente solo podemos predecir a corto plazo, a medida que nos alejamos de los datos iniciales menos sabemos, llegando en el infinito a ser equiprobable y no nos permite decantarnos por ninguna.
      - La incertidumbre se acumula, hasta que no sabremos cual es el estado.
      - Para la mayoría de las cadenas la distribución al final es independiente de la inicial, la distribución al final es igual si empezamos con $P(X_0=\text{lluvia})=1$
  - **Modelos de Markov Ocultos (HMMs):** Hay variables observables, evidencias, que determinan el estado. Tiene forma de red bayesiana en el que las observaciones cuelgan del estado y solo dependen del estado actual. Cada estado depende del inmediato anterior.
    - **Se asume:**
      - Es un proceso estacionario.
      - Se cumple la hipotesis de Markov.
      - Las observaciones en t solo dependen del estado en t. P(O/E)

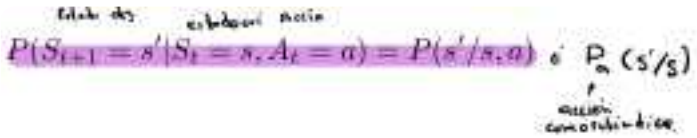


    - **Se define por:**
      - Conjunto de estados y de observaciones.
      - La probabilidad a priori, aquel que no depende del anterior. $P(E_0)$
      - El modelo de transiciones. $P(E_1/E_0)$
      - El modelo de observaciones. $P(O_1/E_1)$
    - La probabilidad de las transiciones se representan en la tabla de probabilidad condicionada.

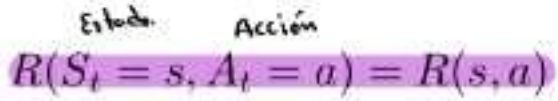
- **Inferencia en HMMs:**
  - Se puede resolver por definición recursiva.
  - La forma mas eficiente es por programación dinámica, pero no la veremos.
  - Nosotros las resolveremos mediante inferencia exacta en redes Bayesianas, a pesar de que es menos eficiente.
  - **Se factoriza como:**



- Tareas típicas de Inferencia:
  - **Problema de Evaluación**: Calcular la probabilidad de una secuencia de observaciones.
  - **Problema de Decodificacion**: Dada una secuencia de observaciones, determinar cual es la secuencia de estados correspondiente que explica mejor esas observaciones.
  - **Problema de Filtrado:** Distribución de probabilidad del estado actual dada cierta evidencia histórica, desde t hacia atrás.
  - **Problema de Predicción:** Probabilidad de estados futuros dada evidencia.
- **Procesos de Decisiones de Markov (MDPs)**:
  - Se toma una decision en cada instante. El caso general es que sean acciones no deterministas, hay varias posibilidades en cada caso y cada una de ellas tiene un probabilidad y debemos decidir cuál tomamos.
  - Las **probabilidades de transicion** dependen de las acciones.



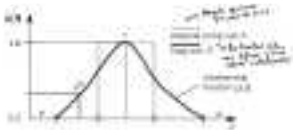
  - Hay un **refuerzo o un coste**, indica como de bueno o malo es esa transición, por cada par de estado-accion:



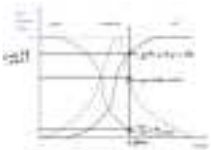
  - Se define como una tupla **<S, A, P, R>:**
    - **S**, estados.
    - **A**, acciones.
    - **P**, probabilidades P(s'/ s, a). Cada estado tiene una probabilidad para cada accion con sus estados contiguos.
    - **R**, refuerzos o costes R(s, a).
  - El objectivo es determinar que accion ejecutar en cada estado para maximizar el refuerzo o minimizar el coste.
  - Al no ser determinista hay **dos opciones:**
    - **Politica** (Acciones estocásticas): Se determinada para cada estado un accion determinada.
      - Es un mapeo completo de estados a acciones, pero no es una secuencia de acciones. Aunque haya un fallo en la ejecucion el agente puede seguir.
      - Maximiza el refuerzo esperado o minimiza el coste esperado en lugar de alcanzar un estado meta.
      - Para cada MDP existe una politica optima.
      - Determina que hacer independientemente del efecto de cualquier accion en cualquier instante de tiempo.
    - **Replanificar**: Se vuelve a planificar cuando sale del anterior, de esta manera cada estado tiene en cuenta el anterior.
  - **Accion estocástica**: Consigue el efecto deseado con probabilidad p.

- Para el caso de **minimizar coste:**
  - **Politica optima π***: mínimo coste esperado.
    - **Acciones deterministas**: Si la accion a lleva al estado s', el coste es:
      - **C(a) + costeDesde(s')**
    - **Acciones no deterministas**: Si la accion a tiene efectos probabilisticos, el coste esperado es:
      - **C(a) + Σsub(s') P(s'| s, a) * costeDesde(s')**
- **Funcion de valor:** coste esperado de un estado.
  - **V(s):** coste esperado de alcanzar la meta desde s.
  - **Busqueda**: Coste del camino optimo desde s. Se puede usar como heuristica, y es la heurística perfecta h*(s)
  - **MDP**: Coste esperado de la estrategia optima para alcanzar la meta desde s. Conociendo V(s) podemos calcular una politica optima π*.
- **Calcular V(s).**
  - **Ecuaciones de Bellman**. Acciones deterministas (Es programación dinámica, recursividad)
    - Si s es un estado meta, V(s) = 0. Es sumidero. Inicialmente todos están a 0.
    - En el resto de casos, donde s' es el estado resultado de aplicar a en s:
      -
  - **Ecuaciones de Bellman**. Acciones no deterministas.
    - Dominios estocasticos, apartir de la accion a:
    - Entonces, el estado meta V(s)=0 para costes y en el resto de casos:
    - **Politica optima**: Consiste en elegir las accione que han generado el mínimo.
    - **Resolviendo las ecuaciones de Bellman:**
      - **Algoritmo de Iteración de Valor:**
        - El V(s) de los finales es 0 para costes y el refuerzo si hay refuerzos.
        - Inicialmente todos se ponen a 0.
        - Se hacen tantas rondas como sean necesarias para que los valores de V(s) se estabilicen entre rondas. Y cada ronda calcula V(s) para cada estado.
        - Termina cuando alcanza un punto fijo, cuando los valores no cambian en dos iteraciones sucesivas. Se estabiliza.

- **Logica Borrosa:**
  - Los conceptos no son ciertos o falsos de forma clara, a diferencia de la logica clásica en la que los valores solo son true o false. Se emplean conceptos y modificadores que son difusos.
  - **Representación:**
    - Conjunto borroso.
    - **Función de pertenencia:** Indica en que medida un elemento pertenece al conjunto borroso. Valores en el rango [0, 1], en el que 0 representa absolutamente falso y 1 absolutamente verdadero.



  - **Logica borrosa vs. probabilidad:**
    - **Probabilidad**: Lo hechos ocurren o no. Expresan conocimiento parcial.
    - **Logica borrosa**: Conceptos vagos, inciertos. Expresa grado de verdad parcial.
  - **Sistema de reglas borrosos:**
    - Descripcion del problema, las reglas que aparecen.
    - Definir los términos borrosos de las reglas, mediante su funcion de pertenecía.
      - **Variables**: Expresan cualidades.
      - **Valores**: Las variables toman valores de un dominio discreto. Los limites entre los valores son borrosos.
      - **Tipos**: Sigmoidal(la S), Gausiana (n), triangular y trapezoidal. Las dos ultimas son las que utilizaremos nosotros.
      - **Modificadores**: Operan sobre la funcion de pertenencia. ², ³...



    - **Combinar términos:**
      - **Conjunción**: Intersección. El menor de ambas para cada x. MIN
      - **Disyunción**: Union. El maximo en cada x de las funciones. MAX
      - **Negación**: Los valores inversos de cada x. 1-valor
    - **Combinar reglas**, para generar un salida única. (Varias son parcialmente cercas)
      - p->q p es cierto en un grado, entonces q tambien es cierto en un grado.
  - **Sistema de reglas borrosas:**
    - Entrada dato nítida.
    - Borrosificar el dato nítido.
    - Base conocimiento, reglas borrosas.
    - Desborrosificar, pasar a un valor nítido.
    - Salida dato.
  - **Inferencia con reglas borrosas:** 4 pasos.
    - **Método de Mandami**: es el método típico de inferencia borrosa.
      - **Borrosificar las entradas:**
        - Determinar en que grado la entrada nitida pertenece a los conjuntos borrosos, para cada valor que puede tomar la variable.
      - **Evaluación de reglas:**
        - En que medida las entradas borrosificadas verifican los antecedentes de la regla, se evalúan todas la reglas. AND es intersección, OR es union y NOT la inversa. Si es un solo valor se coge directamente el grado de la entrada.
        - Se obtiene la **Similitud**, que es el mayor grado que puede obtener el consecuente.
        - **El consecuente**, resultado de cortar la función de pertenencia del consecuente al nivel que marca la similitud del antecedente.
      - **Agregación de los consecuentes:**
        - Unificación de las salidas de todas las reglas tras evaluarlas en un solo conjunto borroso. Nosotros hacemos el MAX para cada x, union.
      - **Desborrosificar el resultado:**
        - Convertir el resultado en un valor nítido, el más común es centro de gravedad o centro de del area.

- Ventajas
  - Representa la vaguedad del lenguaje de forma natural
  - Generaliza los conjuntos nítidos
  - Permite diseños flexibles desde el punto de vista ingenieril
  - Buen rendimiento
  - Métodos simples de implementar
  - ¡Normalmente funcionan bien!
- Desventajas
  - Hay que diseñar las funciones de pertenencia
  - Normalmente requiere de un ajuste fino de los parámetros
  - La defuzzificación puede producir resultados no deseados
- Herramientas
  - Matlab (Fuzzy Toolbox)
  - FuzzyClips