

4.1 Introducción

Hay dos problemas centrales de la IA:

- **Representación** del conocimiento: **Expresividad** de sistemas de representación, espacio de estados con un grafo (cada nodo un estado o solución), las flechas son transformaciones...
- **Razonamiento** automático: En términos de **corrección**, **completud** (las consecuencias se pueden deducir con el método) y **complejidad** (capacidad de cómputo)
- La lógica nos permite tener métodos para representar y razonar:
- **Lenguaje**: Es el simbolismo para representar información (Proposicional, predicados...)
- **Semántica**: Nos dará el criterio de que es cierto o no lo es.
- **Cálculo**: Sistemas de razonamiento y deducción que nos permiten, dadas unas premisas, que continúan a partir de ellas.

4.2 Lógica proposicional

Lenguaje de lógica de proposiciones: estructuras simples \rightarrow más complejas.

1. Infinitas estructuras simples: $\{p, q, r\}$ (Proposiciones atómicas)
2. Constantes: $\begin{cases} \top \text{ (siempre cierto)} \\ \perp \text{ (siempre falso)} \end{cases}$
3. Símbolos lógicos, conectivas u operadores:
 - o monaria: \neg (Negación)
 - o binarias: $\begin{cases} \wedge \text{ (Conjunción Y)} \\ \vee \text{ (Disyunción O)} \\ \rightarrow \text{ (Condicional, Si _ entonces _)} \\ \leftrightarrow \text{ (Bicondicional, _ Si solo _)} \end{cases}$
4. Símbolos auxiliares: $()$.

Semántica:

Mediante un **modelo M** es una asignación de valores de verdad a variables proposicionales:

$M \models p$ (\models indica que una fórmula p es verdadera en un modelo M)

Es decir p sería verdadera en el modelo M cuando: $M(p) = V$.

Por ejemplo: $M \models \neg p$ es verdadera cuando: *No se cumple que $M \models p$*

$M \models \neg p \wedge q$ es verdadera cuando: *Se cumplen $M \models p$ y $M \models q$*

O también mediante **Tablas de verdad**:

p	q	$\neg p$	$p \wedge q$	$p \vee q$	$p \rightarrow q$	$p \leftrightarrow q$
0	0	1	0	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	0	0
1	1	0	1	1	1	1

Nociones semánticas:

Tautología: O fórmula válida, es una fórmula siempre verdadera. \top (siempre cierto)

Contradicción: O fórmula siempre falsa. \perp (siempre falso)

Equivalencia: Dos fórmulas son equivalente si y solo si son verdaderas en los mismos modelos.

Consecuencia: Una fórmula p es consecuencia de un conjunto de fórmulas Q , si todo modelo que haga verdadera cada fórmula de Q hace verdadera a p . $Q \models p$

Propiedades: La relación de consecuencia tiene las siguientes propiedades.

- a) Toda f. es consecuencia de ella misma (**Reflexividad**):
- b) Toda f. es consecuencia de cualquier conjunto de premisas que la incluya: $X, Y, Z \models X$
- c) **Transitividad**: Si, $H_1, \dots, H_n \models C$ y $C \models W$ entonces $H_1, \dots, H_n \models W$

Consecuencia y condicionales tautológicos:

De la consecuencia, $X, Y \models C$, $X \wedge Y \rightarrow C$ es una tautología. Y viceversa.

Dadas dos fórmulas, $A \equiv B$, si y sólo si $B \models A$ y $A \models B$

$A \rightarrow B$ como $B \rightarrow A$ son tautologías, si y sólo si $A \leftrightarrow B$ es una tautología.

Consecuencia e insatisfacibilidad: $X, Y \models C$ entonces $\{X, Y, \neg C\}$ es insatisfacible. Y viceversa

Satisfacibilidad: Expansión/Reducción

- conjunto **satisfacible**, su ampliación **no garantiza satisfacibilidad**.
- conjunto **satisfacible** un subconjunto es **satisfacible**.
- conjunto **insatisfacible** su ampliación es **insatisfacible**.
- conjunto **insatisfacible** un subconjunto **no garantiza satisfacibilidad**.

Equivalencias básicas

Doble negación: $\neg \neg X \equiv X$

Disyunción o conjunción consigo misma:

$$\begin{array}{llll} X \vee X \equiv X & X \wedge X \equiv X & X \vee \neg X \equiv \top & X \wedge \neg X \equiv \perp \\ X \vee \perp \equiv X & X \wedge \top \equiv X & X \vee \top \equiv \top & X \wedge \perp \equiv \perp \end{array}$$

Propiedades conmutativa, asociativa y distributiva:

$$\begin{array}{ll} X \vee Y \equiv Y \vee X & X \wedge Y \equiv Y \wedge X \\ X \vee (Y \vee Z) \equiv (X \vee Y) \vee Z & X \wedge (Y \wedge Z) \equiv (X \wedge Y) \wedge Z \\ X \vee (Y \wedge X) \equiv X & X \wedge (Y \vee X) \equiv X \\ X \vee (Y \wedge Z) \equiv (X \vee Y) \wedge (X \vee Z) & X \wedge (Y \vee Z) \equiv (X \wedge Y) \vee (X \wedge Z) \end{array}$$

Leyes de De Morgan, condicionales, bicondicionales:

$$\begin{array}{ll} \neg(X \wedge Y) \equiv (\neg X \vee \neg Y) & \neg(X \vee Y) \equiv (\neg X \wedge \neg Y) \\ X \rightarrow Y \equiv \neg X \vee Y & X \rightarrow Y \equiv \neg Y \rightarrow \neg X \\ X \leftrightarrow Y \equiv (X \rightarrow Y) \wedge (Y \rightarrow X) & X \leftrightarrow Y \equiv (X \wedge Y) \vee (\neg X \wedge \neg Y) \end{array}$$

Formas normales:

forma normal conjuntiva: conjunción de disyunciones

forma normal disyuntiva disyunción de conjunciones

Fundamentos de inteligencia artificial

Transformar en formas normales: Toda f. proposicional puede reescribirse en forma normal.

- 1) Eliminar bicondicionales: $X \leftrightarrow Y \equiv (X \rightarrow Y) \wedge (Y \rightarrow X)$
- 2) Eliminar condicionales: $X \rightarrow Y \equiv \neg X \vee Y$
- 3) Luego hay que introducir todas las negaciones hasta que afecten sólo a letras proposicionales (no a paréntesis más complejos):

$$\neg(X \wedge Y) \equiv (\neg X \vee \neg Y)$$

$$\neg(X \vee Y) \equiv (\neg X \wedge \neg Y)$$

- 4) Eliminación de las dobles negaciones: $\neg\neg X \equiv X$

- 5) Aplicar la prop. distributiva para tener las formas canónicas:

$$X \wedge (Y \vee Z) \equiv (X \wedge Y) \vee (X \wedge Z)$$

$$X \vee (Y \wedge Z) \equiv (X \vee Y) \wedge (X \vee Z)$$

Por ejemplo, si tenemos la siguiente fórmula: $(p \leftrightarrow \neg q) \vee r$

Y queremos hallar forma normal conjuntiva, seguimos los pasos señalados:

$$\begin{aligned} & (p \leftrightarrow \neg q) \vee r && \text{Paso} \\ & \equiv ((p \rightarrow \neg q) \wedge (\neg q \rightarrow p)) \vee r && (1) \\ & \equiv r \vee ((p \rightarrow \neg q) \wedge (\neg q \rightarrow p)) \\ & \equiv (r \vee (p \rightarrow \neg q)) \wedge (r \vee (\neg q \rightarrow p)) && (2) \\ & \equiv (r \vee (\neg p \vee \neg q)) \wedge (r \vee (\neg \neg q \vee p)) && (4) \\ & \equiv (r \vee \neg p \vee \neg q) \wedge (r \vee q \vee p) \end{aligned}$$

Tableaux:

Demostrar consecuencia: $A \models B$ ($A \mid \neg B$ Si todas las ramas cerradas, es **consecuencia**)

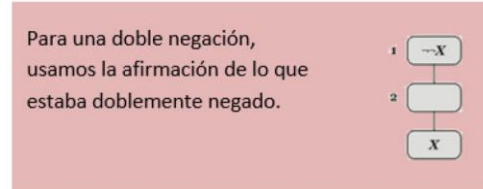
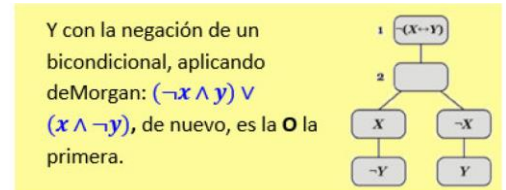
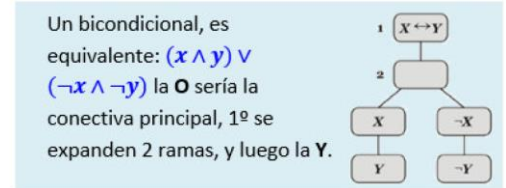
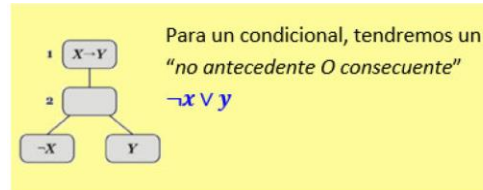
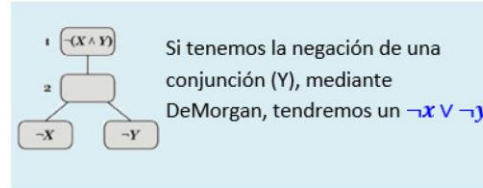
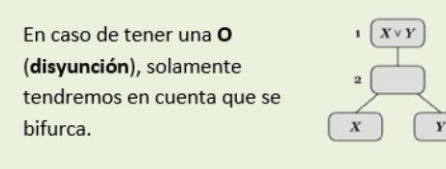
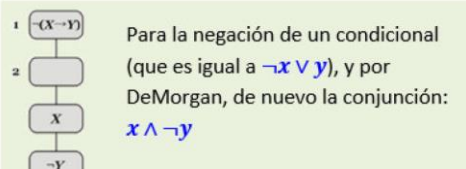
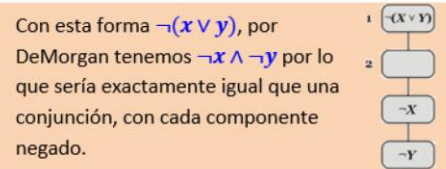
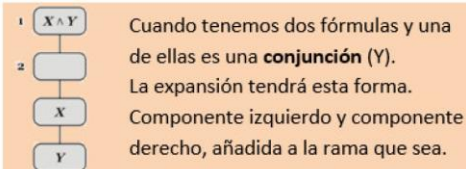
Demostrar satisfacibilidad. $A, B, A \mid B$ está cerrado es **insatisfacible**. Abierto **satisfacible**

Consecuencia y condicionales tautológicos: De, $X, Y \models C$, $X \wedge Y \rightarrow C$ es una tautología. Y viceversa.

Dadas dos fórmulas, $A \equiv B$, si y sólo si $B \models A$ y $A \models B$

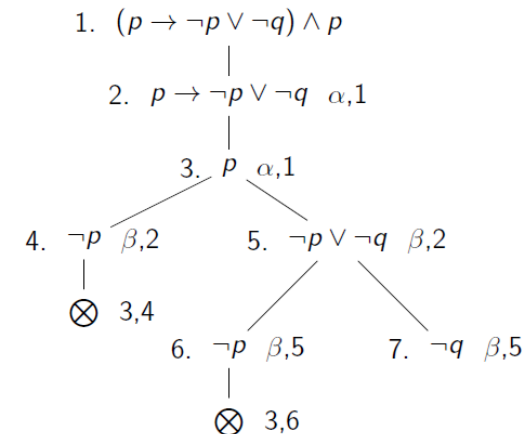
Y $A \rightarrow B$ como $B \rightarrow A$ son tautologías, si y sólo si $A \leftrightarrow B$ es una tautología.

Reglas de expansión de nodos:



Por ejemplo: Comprobamos si $(p \rightarrow \neg p \vee \neg q) \wedge p$ es satisfacible.

Para ello, hacemos su tableau.



Fundamentos de inteligencia artificial

4.3 Lógica de primer orden (Lógica de predicados de primer orden)

Añadimos los siguientes elementos a la lógica proposicional:

Constantes: que identificamos con letras $a, b, c \dots$

Variables: que identificamos con x, y, z .

Funtores: O funciones, identificadas con: f, g, h .

Predicados: Identificados con letras: p, q, r .

Cuantificadores: \forall (para todo...), \exists (existe...)

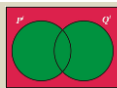
Identificamos cómo **términos** a las constantes, variables y funtores (con sus argumentos formados por términos).

Y **fórmulas atómicas** a predicados con argumentos formados por términos

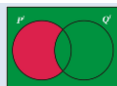
Ejemplos de interpretación de predicados

- $\exists y, \forall x, r(y, x)$: existe algún elemento ($\forall y$) que está relacionado con todos los demás ($\exists x$).
- $\exists x, \forall y, r(x, y)$: Buscamos si existe al menos un x que tiene una cierta propiedad.
- $\exists x, \exists y, r(x, y)$: dos elementos al menos que estén relacionados en la interpretación.
- $\forall x \forall y r(x, y)$: todo elemento del universo.

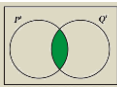
- **Todos P o Q:** $\forall x (Px \vee Qx)$: cada uno de los elementos del universo, o bien tiene la propiedad P, o bien la Q, o bien ambas.



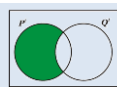
- **Todos los P son Q:** $\forall x (Px \rightarrow Qx)$: Todo elemento que cumple Q, pero solo los elementos que cumplen P que a la vez cumplen Q.



- **Algún P y Q:** $\exists x (Px \wedge Qx)$: Algún elemento que cumpla P y Q.



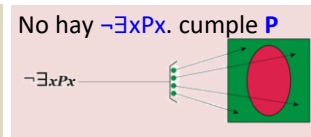
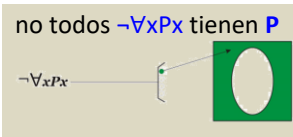
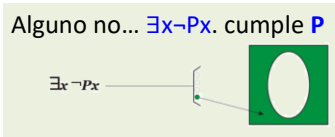
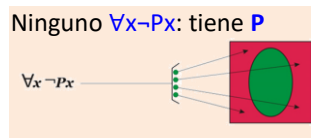
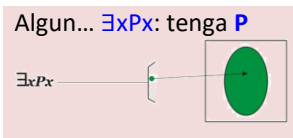
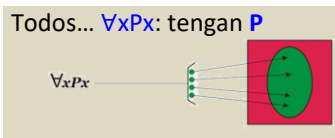
- **Algún P y no Q:** $\exists x (Px \wedge \neg Qx)$: Algún elemento que cumpla P, pero no Q



- **Algún no (P y Q):** $\exists x \neg (Px \wedge Qx)$: Algún elemento que no cumpla a la vez P y Q.



- **Negación cuantificador:** Se estudian las siguientes fórmulas:



Semántica: Para interpretar las fórmulas y poder saber si son verdaderas o falsas.

Tenemos un **modelo** (o **universo de discurso**) M :

Cada **constante** c se interpreta cómo c^M , un elemento de D .

Se puede representar cómo $c^M \in D$ o un modelo, con un dominio y una interpretación:

$M = \langle D, I \rangle$ donde en esa interpretación, la constante c vale 2. $I(c) = 2$.

Cada **functor** f con n argumentos, cómo f^M , una función de D^n en D .

Es decir, una función de n elementos del dominio que los relaciona con otro.

Cada **predicado** p de n argumentos, cómo p^M , un elemento de D^n .

Por ejemplo: En una pregunta de examen: Representa mediante lógica de predicados de primer orden las siguientes sentencias.

- 1- En el estado inicial ningún bloque está derribado y ningún proyectil se ha usado.
- 2- Los proyectiles han de lanzarse contra bloques, pero no se puede lanzar el mismo proyectil más de una vez.
- 3- Cuando se lanza un proyectil, o derriba un bloque o falla
- 4- Un bloque que no se ha derribado por un proyectil, no puede estar derriba, si es derribado, así se queda
- 5- Si se derriban todos los bloques, suena una trompeta.

Se sugiere usar una variable t para representar el tiempo, el predicado $<$ que define un orden total sobre el tiempo, y predicados de acción (por ejemplo $derribar(b, p, t)$ y de estado (por ejemplo $derribado(b, t)$).

1. $\forall b \neg derribado(b, t_0) \wedge \forall p \neg usado(p, t_0)$

Es decir, para todo bloque (variable b) no hay ningún bloque en el estado inicial. Y para toda variable p (proyectil), no está usado ningún proyectil en el instante inicial.

2. $\forall p \exists t (\neg usado(p, t) \rightarrow \exists t' > t (lanzar(p, t') \wedge usado(p, t')))$

$\forall p \exists t$: Para todo proyectil existe un instante de tiempo donde:

$\neg usado(p, t)$: Si no está usado el proyectil p en el instante t

$\rightarrow \exists t' > t$: entonces existe un instante de tiempo posterior a que t :

$lanzar(p, t') \wedge usado(p, t')$: se lanza el proyectil y ya está usado

3. $\forall p \forall t ((lanzar(p, t) \rightarrow \exists t' > t (fallar(p, t') \vee usado(p, t')))$

4. $\exists t \exists b \exists p derribar(b, p, t) \rightarrow \forall t' > t (derribado(b, t') \wedge \forall t'' < t \neg derribado(b, t''))$

5. $\exists t (\forall b derribado(b, t) \wedge \neg \exists t' < t (\forall b derribado(b, t'))) \rightarrow trompeta(t)$

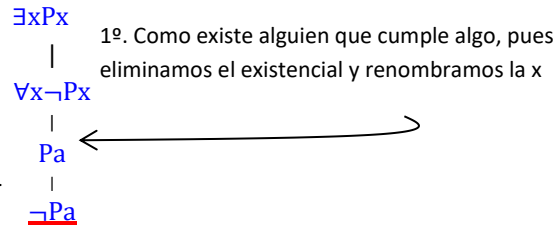
Fundamentos de inteligencia artificial

Tableaux para fórmulas de lógica de predicados: Por ejemplo: Si partimos de: $\exists xPx$ y $\forall x\neg Px$ Por ejemplo:

“**existe alguien con una propiedad**” y a la vez “**para todos nadie cumple la propiedad**”

Entonces:

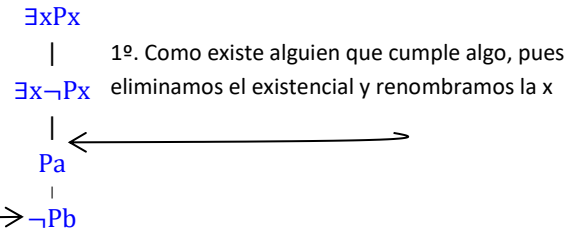
- 1º. Como todos no cumplen algo (Ese algo será el mismo). Eliminar el para todo y renombramos



Los nodos universales se pueden expandir indefinidamente. Para **existenciales**, uno por rama.

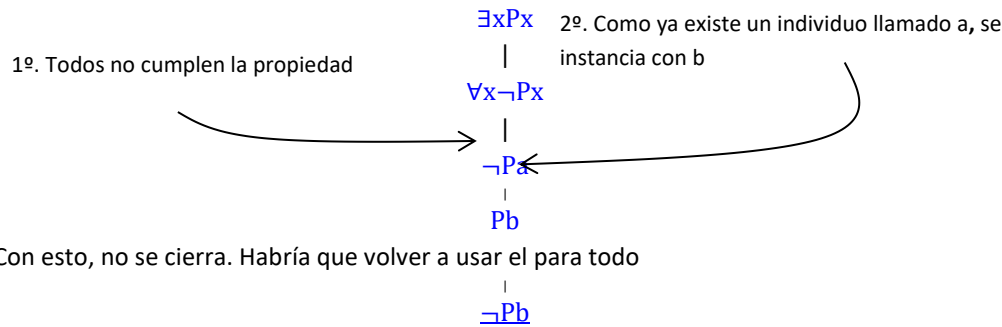
Para esta opción: **Existe alguien que cumple algo, y existe alguien que no cumple algo.**

- 2º. Para expandir la segunda. Con $\neg Pa$ parece que es insatisfacible, pero no lo es, ya que al instanciar un existencial, hay que usar siempre el nombre de individuo que no se haya usado en esa rama



Si es **satisfacible**. Se busca un elemento con propiedad **P** y uno con propiedad **noP** y se cumple.

Si, de nuevo en la primera opción, instanciamos primero el para todo:



Con esto, no se cierra. Habría que volver a usar el para todo

Entonces, hasta ahora, tenemos las siguientes reglas:

1. (algo de **todo**): $\forall xX$ o $\neg\exists xX$, sobre una variable nueva **a**, será el que cumple la condición:
2. Si hay, $\neg\forall xX$ o $\exists xX$, si existe alguien, , **siempre hay que escoger constante nueva.**

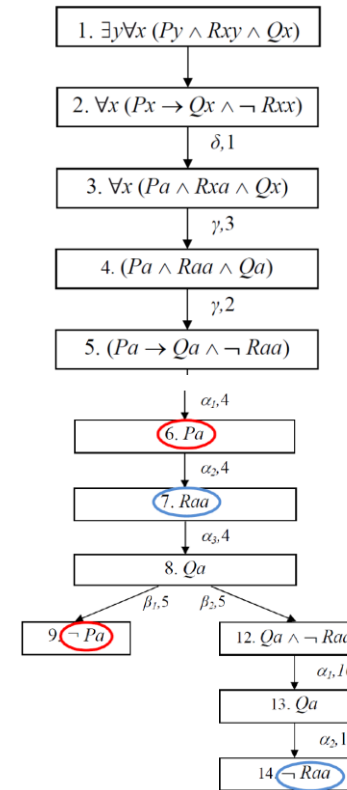
Reglas de expansión de los nodos en un tableaux: Todas las reglas.

$\forall x(\dots x \dots)$	$\neg\exists x(\dots x \dots)$
$(\dots a \dots)$	$\neg(\dots a \dots)$
$(\dots b \dots)$	$\neg(\dots b \dots)$
\vdots	\vdots

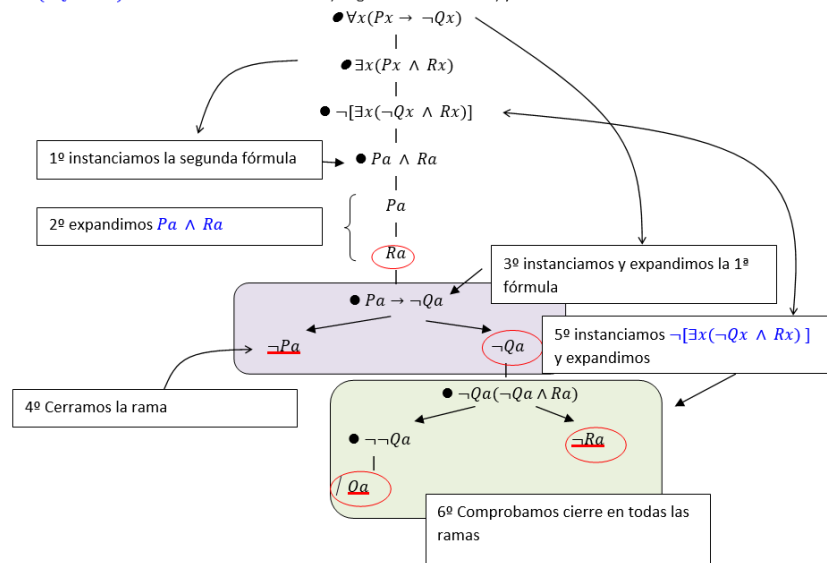
$\exists x(\dots x \dots)$	$\neg\forall x(\dots x \dots)$
$(\dots d \dots)$	$\neg(\dots d \dots)$
\vdots	\vdots

1º desarrollamos **Existenciales**

2º **Universales**



Tableaux con cuantificadores. Ejemplo: Si tenemos $\forall x(Px \rightarrow \neg Qx)$, $\exists x(Px \wedge Rx)$. Para demostrar que $\exists x(\neg Qx \wedge Rx)$ es una consecuencia de estas, negamos la conclusión, y ramas cerradas:



Fundamentos de inteligencia artificial

Resolución: Para resolver hay que transformar las fórmulas a **forma normal de Skolem** (se convierten a otra forma sin cuantificadores), la fórmula obtenida no es equivalente (son **x satisfacibles**), es decir, no es cierta en los mismos modelos, pero, si una es satisfacible, la otra lo es, si no lo es, tampoco. Y tras convertir, se trata como preposicional. Para transformación:

- Mediante equivalencias, se llevan todos los cuantificadores al inicio de la fórmula. Para convertirla en **forma normal prenexa**: (Todo cuantificador en la parte delantera de la expresión $\exists! \forall$ Predicados)
- Se aplica recursivamente:
 - $\forall x Px$ se transforma en Px : Se quita el cuantificador y la variable queda libre.
 - $\exists x Px$ se transforma en $P(f(x_1, \dots, x_n))$, siendo f un nuevo functor y las variables x_1, \dots, x_n , son aquellas cuantificadas universalmente a la izquierda del existencial que se elimina. Si no hay, f será una constante nueva.

Por ejemplo: Tras usar equivalencias, se comienzan a quitar los cuantificadores recursivamente, en este orden: $\forall x \exists y \exists z \forall w$ (fórmula sin cuantificadores)
Para el universal $\forall x$: Se quita el cuantificador, la variable quedaría libre dentro de la fórmula.

$\exists y \exists z \forall w$ (fórmula sin cuantificadores)

Para el existencial $\exists y$: nuevo functor, f_1 con argumento toda variable \forall a la izquierda, cada aparición de y en la fórmula, se sustituye por $f_1(x)$: $\exists z \forall w$ (fórmula sin cuantificadores)

Para el existencial $\exists z$: Igual, cada aparición de z , por $f_2(x)$ o $(g(x))$ lo importante es que el functor no esté en la fórmula previamente): $\forall w$ (fórmula sin cuantificadores)

Para el universal $\forall w$: Se elimina, y w quedará libre.

Equivalencias:

1ª equivalencias básicas para predicados

$\forall x Px \equiv \forall y Py$: renombrar, la variable y cambiarla en todo el ámbito del cuantificador.

$\neg \forall x Px \equiv \exists x \neg Px$: **No todos son altos = existe alguien que no lo es**

$\neg \exists x Px \equiv \forall x \neg Px$: **Si no existe alguien alto = todos son no altos**

$\forall x Px \equiv \neg \exists x \neg Px$: **Todos cumplen P = No existe ninguno que no lo cumpla**

$\neg \forall x \neg Px \equiv \exists x Px$: **No todos no cumplen P = Existe alguno que cumple P**

Una sentencia como $\forall x Px$ si se evalúa como: $(P_1 \wedge P_2 \wedge P_3) \cdot \exists x Px$ como: $(P_1 \vee P_2 \vee P_3)$.

Así, $\forall x \neg Px$ se podría visualizar como $(\neg P_1 \wedge \neg P_2 \wedge \neg P_3)$.

2ª equivalencias básicas para predicados : A partir de 3ª, no puede haber x libres en el ámbito

$\forall x (\Box \wedge \forall x \Box) \equiv \forall x (\Box \wedge \Box)$

$\exists x (\Box \vee \exists x \Box) \equiv \exists x (\Box \vee \Box)$

$\forall x (\Box \wedge \Box) \equiv \forall x (\Box \wedge \Box)$

$\exists x (\Box \wedge \Box) \equiv \exists x (\Box \wedge \Box)$

$\forall x (\Box \vee \Box) \equiv \forall x (\Box \vee \Box)$

$\exists x (\Box \vee \Box) \equiv \exists x (\Box \vee \Box)$

$\forall x (\Box \rightarrow \Box) \equiv \exists x (\Box \rightarrow \Box)$

$\exists x (\Box \rightarrow \Box) \equiv \forall x (\Box \rightarrow \Box)$

$\Box \rightarrow \forall x \Box \equiv \forall x (\Box \rightarrow \Box)$

$\Box \rightarrow \exists x \Box \equiv \exists x (\Box \rightarrow \Box)$

4.4 Extensiones de las lógicas clásicas

Algunas buscan evitar problemas de decidibilidad manteniendo capacidad expresiva aceptable.

Otras, modelan propiedades del razonamiento humano que escapan a la lógica clásica.

Por ejemplo, tolerancia a errores (inconsistencias), información imprecisa, razonamiento contextual, razonamiento no monótono.

Lógica modal:

Es una extensión de la lógica proposicional con dos operadores:

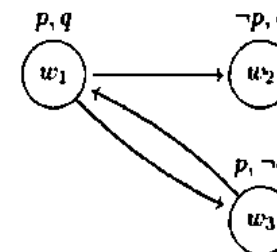
- $\Box p$: es necesario/obligatorio p

- $\Diamond p$: es posible p

Por ejemplo y respectivamente (todo lugar, algún lugar...)

El **modelo** es: $M = \langle W, R, V \rangle$.

Ejemplo:



W : Conjunto de mundos posibles. (Un momento en concreto, **por ejemplo**, w_1, w_2 y w_3)

R : Relación de accesibilidad entre mundos. ($w_1 R w_2$ significa que si estamos en el mundo w_1 es posible w_2 , **en el ejemplo**, las flechas)

V : función que dado un mundo $w \in W$ devuelve $v(w)$, el conjunto de proposiciones verdaderas en el mundo w . (**Por ejemplo**, en el mundo w_1 son verdaderas p y q , pero en w_2 , solo q)

Semántica:

Por ejemplo: Dado un modelo $M = \langle W, R, V \rangle$ y un mundo $w \in W$:

- $M, w \models p \leftrightarrow p \in V(w)$:

es decir, dado un modelo (con sus características) M y dado un mundo w ; una proposición p es verdadera si y solo si p está entre las cosas verdaderas en ese mundo.

- $M, w \models \neg p \leftrightarrow$ no se cumple que $M, w \models p$: es falsa si no se cumple que sea verdadera.

- $M, w \models p \wedge q \leftrightarrow M, w \models p$ y $M, w \models q$: Una conjunción (\wedge) es verdadera si ambas lo son.

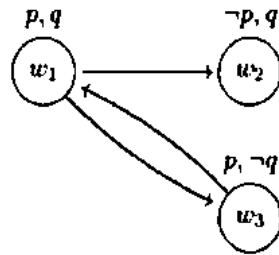
- $M, w \models p \vee q \leftrightarrow M, w \models p$ o $M, w \models q$: Una disyunción (\vee) es verdadera si alguna lo es.

- $M, w \models \Box p \leftrightarrow$ para todo $w' \in W, w R w'$ implica que $M, w' \models p$:

p es necesario si es verdadera en todos los mundos.

Fundamentos de inteligencia artificial

En el ejemplo: En w_1 es necesaria $p \vee q$, ya que en todos los mundos ($p \vee q$) es verdadero, pero no se cumple que $\Box p$ ya que no es verdadero en todos los mundos a la vez.



- $M, w \models \Diamond p \leftrightarrow$ para todo $w' \in W$ tal que wRw' y $M, w' \models p$:

Es decir, existe un mundo accesible donde es verdadero.

En el ejemplo: No es necesario p porque no es verdadero en todos los mundos, pero si posible, ya que hay varios mundos donde es verdadero. Si una fórmula es verdadera en todos los mundos, decimos que es **válida en el modelo**. Hay que tener en cuenta lo siguiente:

$$M, w_2 \models \Box p \wedge \Box \neg p$$

$\Box p$: Es necesario p , porque en todos los mundos accesibles desde w_2 se cumple p , cómo no hay ningún mundo accesible, se cumple p ; pero, a la vez, en todos está $\neg p$.

Para evitar que ocurra esto tan extraño, podemos hacer que el modelo sea **reflexivo** (Relación de accesibilidad de cada mundo a si mismo).

Deducción: Para calcular en lógica modal, sirve cualquier propiedad de LP, y además:

Necesitación, para que p es válida, también lo es $\Box p$ (Si algo es válida, es necesariamente válida).

Axioma: K: $\Box(p \rightarrow q) \wedge \Box p \rightarrow \Box q$: Toda relación de accesibilidad (sea cual sea), si es necesaria una implicación, entonces, si es necesario el antecedente, también lo es el consecuente.

Otros axiomas: Según propiedades de R : Para evitar problemas y contradicciones.

Simetría: $A \rightarrow \Box \Diamond A$: Para que las cosas verdaderas sean necesariamente posibles.

Transitividad: $\Box A \rightarrow \Box \Box A$: Para que las cosas necesarias sean necesariamente necesarias.

Reflexividad: $\Box A \rightarrow A$: Se evita la contradicción anterior.

Aplicaciones: Lógicas temporales basadas en puntos:

Es una lógica modal donde los mundos posibles son instantes y la accesibilidad es la sucesión temporal, que se representa por $<$. Cada mundo sería un instante.

• **LTTL[F, P]** Hay 4 operadores, F ir al futuro, P ir al pasado, G ir siempre hacia el futuro, H :

- $M, w \models Fp \leftrightarrow$ hay un instante $w', w < w'$ tal que $M, w' \models p$.

¿Cuándo es cierto que el futuro p ? si hay un instante mayor a w donde se cumple p .

- $M, w \models Pp \leftrightarrow$ hay un instante $w', w' < w$ tal que $M, w' \models p$.

- $M, w \models Gp \leftrightarrow M, w \models \neg F \neg p$:

No hay ningún momento donde $\neg p$, es decir, siempre hacia el futuro, siempre p .

- $M, w \models Hp \leftrightarrow M, w \models \neg P \neg p$:

Instante próximo y anterior (justo después o justo antes) (con tiempo discreto).

- $M, w \models Xp \leftrightarrow$ en el instante w' posterior a w se verifica que $M, w' \models p$.

- $M, w \models X^{-1}p \leftrightarrow$ en el instante w' anterior a w se verifica que $M, w' \models p$

• Se puede aumentar la expresividad a costa de aumentar el coste computacional.

Añadiendo dos operadores. U (until, hasta que ocurra...), S (since, desde que ocurra...). Y tendremos la lógica **LTTL**.

- $M, w \models pUq \leftrightarrow$ hay un instante w' posterior a w tal que $M, w' \models q$ y en todos los mundos entre w y w' se cumple $M, w \models p$.

- $M, w \models pSq \leftrightarrow$ hay un instante w' anterior a w tal que $M, w' \models q$ y en todos los mundos entre w y w' se cumple $M, w \models p$.

Podemos ver que los operadores F y P se definen a partir de U y S (T es tautología):

- $M, w \models Fp \leftrightarrow M, w \models TUp$:

- $M, w \models Pp \leftrightarrow M, w \models TS p$:

• Tendremos también la lógica **CTL***, donde el tiempo no es lineal y se ramifica en un árbol (más de un futuro posible). Y a los operadores anteriores se le añaden:

- Ap : en todo futuro posible se cumple p .

- Ep : en algún futuro posible se cumple p .

Tiene gran expresividad, pero con gran coste computacional. Se exige que los operadores temporales se combinen siempre como: $A(pUq), E(pSq) \dots$,

Aplicaciones: Lógicas temporales basadas en intervalos:

Se usan cuando estamos interesados en duraciones y no en instantes concretos.

• Lógica de relaciones de Allen **HS**:

- Las fórmulas se evalúan en el **intervalo** $[d_0, d_1]$.

- **Begins:** $M, [d_0, d_1] \models \langle B \rangle p \leftrightarrow$ existe un d_2 tal que $d_0 \leq d_2 < d_1$ y $M, [d_0, d_1] \models p$.

Si desde el instante d_0 al instante d_1 si hay un d_2 que entre d_0 y d_2 es p .

- **Ends:** $M, [d_0, d_1] \models \langle E \rangle p \leftrightarrow$ existe un d_2 tal que $d_0 < d_2 \leq d_1$ y $M, [d_0, d_1] \models p$.

Si desde el instante d_0 al instante d_1 si hay un d_2 que entre d_2 y d_1 es p .

4.5 Conjuntos borrosos

La lógica multivaluada no tiene valores de verdad 0, 1, hay diferentes grados de verdad.

Los predicados son **vagos** (difusos), por ejemplo los individuos son jóvenes, altos... (algunos más que otros).

Estos predicados vagos se definen cómo **conjuntos borrosos**.

En un conjunto borroso A , cada elemento x pertenece en una medida $\mu_A(x) \in [0, 1]$.

Siendo μ_A la **función de pertenencia** de A .

Por ejemplo: Para el predicado ser alto, tenemos la siguiente **función de pertenencia**: $\mu_A(x)$

Que dará los siguientes valores:

$$\mu_A(\text{Torrebruno}) = 0.01$$

$$\mu_A(\text{Yao Ming}) = 0.99$$

Es decir, mientras que Torrebruno dará poco valor de verdad en el predicado “ser alto”, Yao Ming dará mucho más.

Para cada predicado $P, Q \dots$ se define una función de pertenencia $\mu_P, \mu_Q \dots$, para interpretar las conectivas:

- El valor de verdad de $P(a)$ es $\mu_P(a)$.
- El valor de $\neg a$ es $1 - x$, siendo x el valor de verdad de a .
- El valor de verdad de $a \wedge b$ es el **mínimo**. (Y mínimo)
- El valor de verdad de $a \vee b$ es el **máximo**. (O máximo)
- El valor de verdad de $a \rightarrow b$ es $\min(1 - x + y, 1)$: x el valor de verdad de a e y el de b .