Agentes lógicos

El conocimiento y el razonamiento juegan un papel importante cuando se trata de entornos parcialmente observables. Un agente basado en conocimiento puede combinar el conocimiento general con las percepciones reales para inferir aspectos ocultos del estado del mundo, antes de seleccionar cualquier acción. Por ejemplo, un médico diagnostica a un paciente (es decir, infiere una enfermedad que no es directamente observable) antes de seleccionar un tratamiento. Parte del conocimiento que utiliza el médico está en reglas que ha aprendido de los libros de texto y sus profesores, y parte en forma de patrones de asociación que el médico no es capaz de describir explícitamente.

El mundo de wumpus

El mundo de Wumpus es una cueva que esta compuesta por habitaciones conectadas mediante pasillos. Escondido en algún lugar de la cueva esta el wumpus, una bestia que se come a cualquiera que entre en su habitación. El wumpus puede ser derribado por una flecha de un agente, y éste solo dispone de una. Algunas habitaciones contienen hoyos sin fondo que atrapan a aquel que deambula por dichas habitaciones (menos al wumpus, que es demasiado grande para caer en ellos). El único premio de vivir en este entorno es la posibilidad de encontrar una pila de oro. Aunque el mundo de wumpus pertenece más al ámbito de los juegos de computadora, es un entorno perfecto para evaluar los agentes inteligentes.

En la figura 7.2 se muestra un ejemplo del mundo de wumpus. La definición precisa del entorno de trabajo mediante la descripción de REAS es:

- **Rendimiento**: +1000 por recoger oro, -1000 por caer en un hoyo o ser comido por el wumpus, -1 por cada acción que se realice y -10 por lanzar la flecha.
- **Entorno**: una matriz de 4X4 habitaciones. El agente siempre comienza en la casilla etiquetada [1,1], y orientado a la derecha. Las posiciones del oro y del wumpus se escogen de forma aleatoria, mediante una distribución uniforme, a partir de todas las casillas menos la de salida del agente. Además, con probabilidad 0.2, cada casilla puede tener un hoyo.
- **Actuadores**: el agente se puede mover hacia adelante, girar a la izquierda 90°, o a la derecha 90°. El agente puede fallecer de muerte miserable si entra en una casilla en la que hay un hoyo o en la que esta el wumpus vivo. (No sucede nada malo, aunque huele bastante mal, si el agente entra en una casilla con un wumpus muerto) Si hay un muro en frente y el agente intenta avanzar, no sucede nada. La acción de disparar se puede utilizar para lanzar una flecha en línea recta, en la misma dirección y sentido en que se encuentra situado el agente. La flecha avanza hasta que choca contra un muro o alcanza al wumpus (y entonces lo mata). El agente sólo dispone de una flecha, así que, solo tiene efecto el primer disparo.
- Sensores: el agente dispone de cinco sensores, y cada uno le da una pequeña información acerca del entorno.
  - El agente percibirá un mal hedor si se encuentra en la misma casilla que el wumpus o en las directamente adyacentes a él (no en diagonal).
  - El agente recibirá una pequeña brisa en las casillas directamente adyacentes donde hay un hoyo.
  - El agente verá un resplandor en las casillas donde está el oro.
  - Si el agente intenta atravesar un muro sentirá un golpe
  - Cuando el wumpus es aniquilado emite un grito que se puede oir en toda la cueva.

  Las percepciones que recibirá el agente se representan mediante una lista de cinco símbolos, por ejemplo si el agente percibe un mal hedor o una pequeña brisa, pero no ve un resplandor, no siente un golpe, ni oe un grito, el agente recibe la lista [Hedor, Brisa, Nada, Nada, Nada].

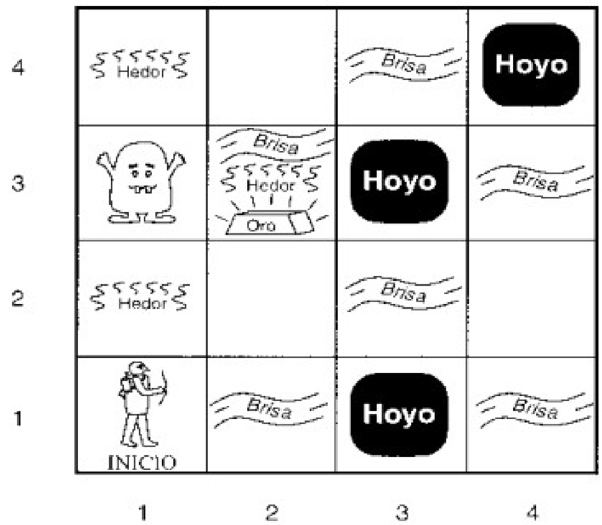

Figura 7.2 Un mundo de wumpus típico. El agente está situado en la esquina inferior izquierda.

La base de conocimiento inicial del agente consiste en reglas de entorno, tal como hemos listado anteriormente; en concreto, el agente sabe que se encuentra en la casilla [1,1] y que ésta es una casilla segura. Veremos como su conocimiento evoluciona a medida que recibe nuevas percepciones y las acciones que va ejecutando.

La primer percepción es [Nada, Nada, Nada, Nada, Nada], de la cual, el agente puede concluir que las casillas vecinas son seguras. La figura 7.3(a) muestra el conocimiento del estado del agente en ese momento.

De los hechos, que no hay mal hedor ni brisa en la casilla [1,1], el agente infiere que las casillas [1,2] y [2,1] están libres de peligro. Entonces las marca con OK para indicar esta conclusión. Supongamos que el agente decide moverse hacia adelante a la casilla [2,1], alcanzando la situación de la figura 7.3(b).

El agente detecta una brisa en la casilla [2,1[[Nada, Brisa, Nada, Nada, Nada], por lo tanto, debe haber un hoyo en alguna casilla vecina, etiqueta a dichas casillas con un ¿P? Un agente cauto solo se movería hacia casillas marcadas con OK, por lo que en este ejemplo el agente podría volver a la casilla [1,2] marcada como OK.

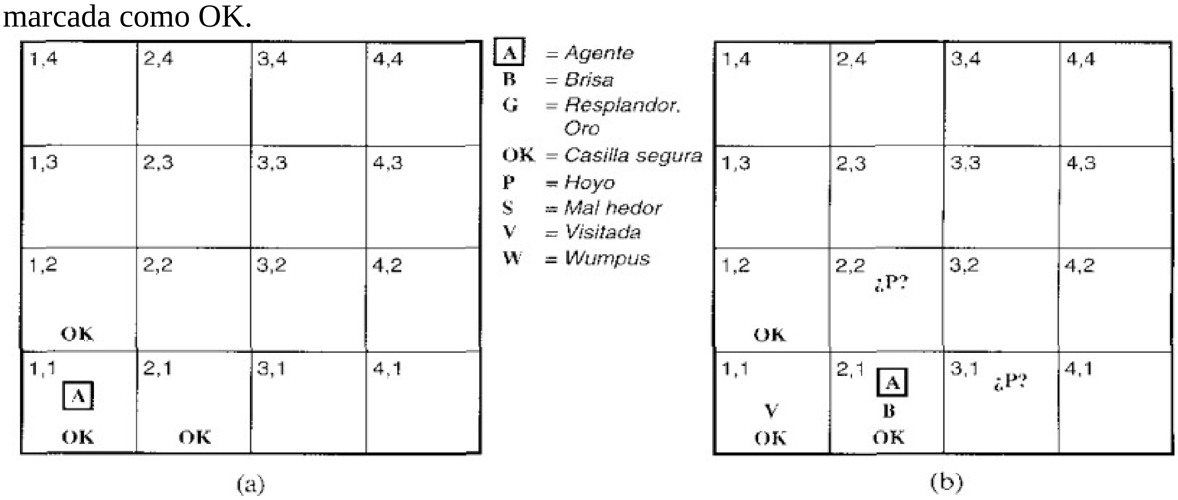

Figura 7.3 El primer paso dado por el agente en el mundo de *wumpus*. (a) La situación inicial, después de la percepción [*Nada, Nada, Nada, Nada, Nada*]. (b) Después del primer movimiento, con la percepción [*Nada, Brisa, Nada, Nada, Nada*].

La nueva percepción en la casilla [1,2] es [Hedor,Nada,Nada,Nada,Nada] obteniendo el estado de conocimiento que se muestra en la figura 7.4(a). El mal hedor en la casilla significa que debe haber un wumpus cerca. Entonces el agente puede inferir que el wumpus está en la casilla [1,3], no puede estar en [1,1] porque de ahí viene el agente, tampoco en [2,2] porque hubiera detectado un mal hedor cuando estuvo en [2,1]. Más aún, la ausencia de brisa en [1,2] implica que no hay hoyo en [2,2], por lo tanto el hoyo debe estar en [3,1]. Todo esto es un proceso de inferencia realmente costoso, ya que debe combinar el conocimiento adquirido en diferentes instantes de tiempo y en distintas situaciones, para resolver la falta de percepciones y poder realizar cualquier paso crucial. La inferencia pertenece a las habilidades de muchos animales, y es típico del tipo de razonamiento que un agente lógico realiza.

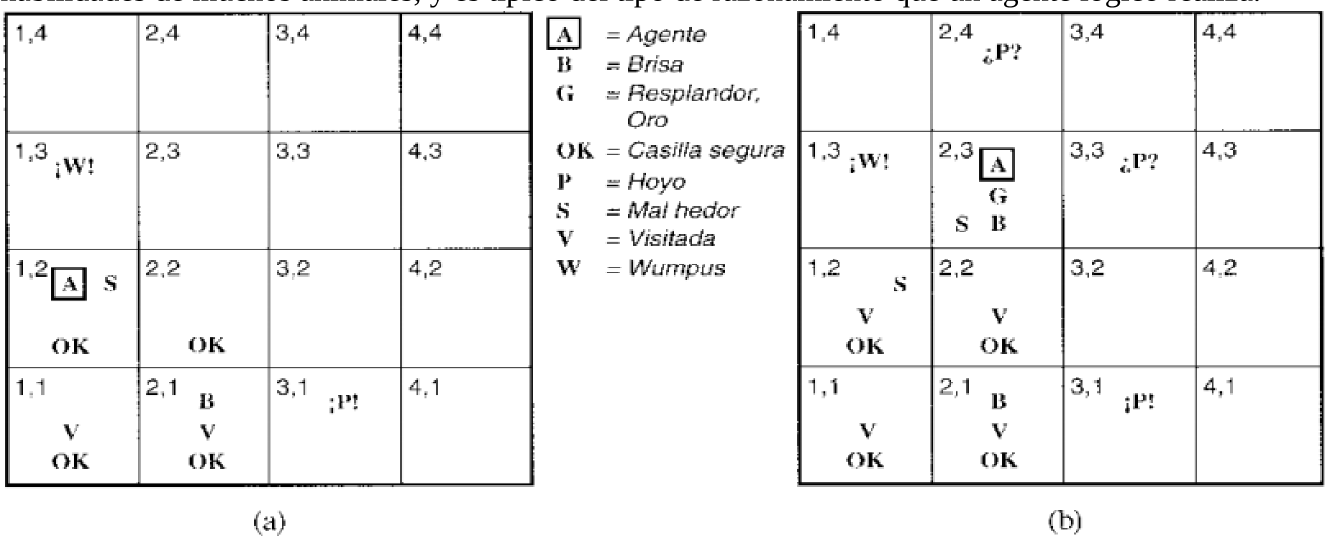

Figura 7.4 Los dos últimos estados en el desarrollo del juego. (a) Después del tercer movimiento, con la percepción [*Hedor, Nada, Nada, Nada, Nada*]. (b) Después del quinto movimiento, con la percepción [*Hedor, Brisa, Resplandor, Nada, Nada*].

El agente ha demostrado que no hay hoyo ni wumpus en [2,2] así que la marca con OK para desplazarse a ella. Asumamos que el agente gira y se desplaza a [2,2], tal como lo muestra la figura 7.34(b). En la casilla [2,3[el agente detecta un resplandor, entonces el agente tomaría el oro y acabaría el juego.

En cada caso en que el agente saca una conclusión a partir de la información que tiene disponible, se garantiza que dicha conclusión es correcta si la información disponible también lo es. Esta es una propiedad fundamental del razonamiento lógico.

Lógica

Las bases de conocimiento se componen de sentencias. Estas sentencias se expresan de a cuerdo a la sintaxis del lenguaje de representación, que especifica todas las sentencias que están bien formadas. Hay literalmente docenas de diferentes sintaxis, algunas que utilizan muchas letras griegas y símbolos matemáticos complejos, otras basadas en diagramas con flechas y burbujas . Sin embargo en todos los casos, las sentencias de la base de conocimiento del agente son configuraciones físicas reales del agente. El razonamiento implica generar y manipular estas configuraciones.

Una lógica también debe definir la semántica del lenguaje. Si lo relacionamos con el lenguaje hablado, la semántica trata el significado de las sentencias. En lógica, esta definición es bastante más precisa. La semántica del lenguaje define el valor de verdad de cada sentencia respecto a cada mundo posible. Por ejemplo, la semántica que se utiliza en aritmética especifica que cada sentencia X+Y=4 es verdadera en un mundo en el que X sea 2 e Y sea 2, pero falsa en uno en que X sea 1 e Y sea 1. En las lógicas clásicas cada sentencia debe ser o bien verdadera o bien falsa en cada mundo posible, no puede ser verdadera y falsa al mismo tiempo.

El razonamiento lógico requiere de la relación de implicación lógica entre las sentencias, su notación matemática es:

$$\alpha \models \beta \quad \text{también} \quad \alpha \Rightarrow \beta$$

Para indicar que la sentencia α implica la sentencia β. La definición formal de la implicación para α y β es: $\alpha \models \beta$ si y solo si en cada modelo en el que α es verdadera, β también es verdadera.

En concreto, si una Base de conocimiento (BC) es verdadera en el mundo real, entonces cualquier sentencia que se derive de la BC mediante un proceso de inferencia sólido también será verdadera en el mundo real. Así, mientras que un proceso de inferencias opera con la sintaxis el proceso se corresponde con la relación del mundo real según la cual algún aspecto del mundo real es cierto en virtud de que otros aspectos del mundo real lo son.

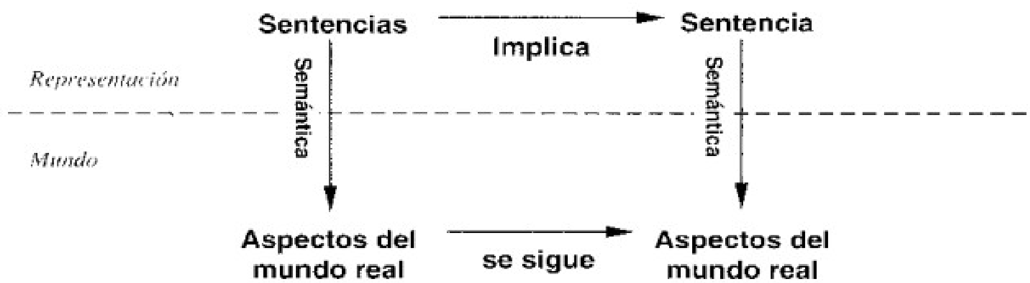

Figura 7.6 Las sentencias son configuraciones físicas del agente, y el razonamiento es el proceso de construcción de nuevas configuraciones físicas a partir de las antiguas. El razonamiento lógico debería asegurar que las nuevas configuraciones representen aspectos del mundo que realmente se siguen de los aspectos que las antiguas configuraciones representan.

Lógica proposicional

Veamos una lógica muy sencilla llamada lógica proposicional.

Sintaxis

La sintaxis de la lógica proposicional nos define sentencias que se pueden construir. Las sentencias atómicas se componen de un único símbolo proposicional. Cada uno de estos símbolos representa una proposición que puede ser verdadera o falsa. Utilizaremos letras mayúsculas para estos símbolos. Los nombres de los símbolos suelen ser arbitrarios pero a menudo se escogen de manera que tengan algún sentido mnemotécnico para el lector. Hay dos símbolos proposicionales con significado fijo: Verdadero, que es la proposición que siempre es verdadera y Falso, que es la proposición que siempre es falsa.

Las sentencias complejas se construyen a partir de sentencias más simples mediante el uso de conectivas lógicas, que son las siguiente:

Conectiva	Expresión en el lenguaje natural	Ejemplo	Símbolo en este artículo
Negación	no	**No** está lloviendo.	$\neg$
Conjunción	y	Está lloviendo **y** está nublado.	$\wedge$
Disyunción	o	Está lloviendo **o** está soleado.	$\vee$
Condicional material	si... entonces	**Si** está soleado, **entonces** es de día.	$\rightarrow$
Bicondicional	si y sólo si	Está nublado **si y sólo si** hay nubes visibles.	$\leftrightarrow$
Disyunción opuesta	ni... ni	**Ni** está soleado **ni** está nublado.	$\downarrow$
Disyunción exclusiva	o bien... o bien	**O bien** está soleado, **o bien** está nublado.	$\nleftrightarrow$

En la figura 7.7 se muestra una grmática formal de la lógica proposicional.

$$
\begin{array}{rcl}
\textit{Sentencia} & \rightarrow & \textit{Sentencia Atómica} \mid \textit{Sentencia Compleja} \\
\textit{Sentencia Atómica} & \rightarrow & \textbf{Verdadero} \mid \textbf{Falso} \mid \textit{Símbolo Proposicional} \\
\textit{Símbolo Proposicional} & \rightarrow & \mathbf{P} \mid \mathbf{Q} \mid \mathbf{R} \mid \ldots \\
\textit{Sentencia Compleja} & \rightarrow & \neg\, \textit{Sentencia} \\
 & \mid & (\textit{Sentencia} \wedge \textit{Sentencia}) \\
 & \mid & (\textit{Sentencia} \vee \textit{Sentencia}) \\
 & \mid & (\textit{Sentencia} \Rightarrow \textit{Sentencia}) \\
 & \mid & (\textit{Sentencia} \Leftrightarrow \textit{Sentencia})
\end{array}
$$

Figura 7.7 Una gramática BNF (Backus-Naur Form) de sentencias en lógica proposicional.

La gramática es muy estricta respecto al uso de paréntesis: cada sentencia construida a partir de conectivas binarias debe estar encerrada en paréntesis. Esto asegura que la gramática no sea ambigua.También podemos omitir los parentesis apoyandonos de un orden de precedencia de las conectivas. Es una precedencia similar a la de la aritmética (por ejemplo, ab+c es equivalente a ((ab)+c) porque la multiplicación tiene mayor precedencia que la suma). El orden de precedencia en la lógica proposicional (de mayor a menor) es justamente el orden en el que aparecen en la tabla de arriba, primero la negación, luego la conjunción, luego la disyunción, etc. Así la sentencia

$$\neg\, P \vee Q \wedge R \Rightarrow S$$

es equivalente a

$$((\neg\, P) \vee (Q \wedge R)) \Rightarrow S$$

Semántica

Una vez especificada la sintaxis de la lógica proposicional, vamos a definir su semántica. En lógica proposicional un modelo define el valor de verdad. Se debe especificar como obtener el valor de verdad de cualquier sentencia, dado un modelo. Este proceso se realiza de forma recursiva. Todas las sentencias se construyen a partir de las sentencias atómicas y las conectivas lógicas, entonces cnecesitamos estableces cómo definir el valor de verdad de las sentencias atómicas y cómo calcular el valor de verdad de las sentencias construidas con las conectivas lógicas. Para las sentencias atómicas es sencillo:

- Verdadero es verdadero en todos los modelos y Falso es falso en todos los modelos.
- El valor de verdad de cada símbolo proposicional se debe especificar directamente para cada modelo.

Para las sentencias complejas tenemos reglas como la siguiente:

- Para toda sentencia P, Q y todo modelo m, se aplican las siguientes reglas

P	Q	$\neg P$	$P \wedge Q$	$P \vee Q$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
falso	*falso*	*verdadero*	*falso*	*falso*	*verdadero*	*verdadero*
falso	*verdadero*	*verdadero*	*falso*	*verdadero*	*verdadero*	*falso*
verdadero	*falso*	*falso*	*falso*	*verdadero*	*falso*	*falso*
verdadero	*verdadero*	*falso*	*verdadero*	*verdadero*	*verdadero*	*verdadero*

Ahora que hemos definido la semántica de la lógica proposicional, podemos construir una base de conocimiento para el mundo de wumpus. Para simplificar, solo trataremos con hechos y reglas acerca de hoyos.

Primero, necesitamos escoger nuestro vocabulario de símbolos proposicionales. Para cada i, j;

- Hacemos que $H_{i,j}$ sea verdadero si hay un hoyo en la casilla [i,j]
- Hacemos que $B_{i,j}$ sea verdadero si hay una corriente de aire (brisa) en la casilla [i,j]

La base de conocimiento contiene, cada una identificada con un identificador, las siguientes sentencias:

- No hay ningún hoyo en la casilla [1,1]

$$R_1: \quad \neg H_{1,1}$$

- En una casilla se siente una brisa si y solo si hay un hoyo en una casilla vecina. Estas reglas se deben escribir para cada casilla, escribirnos dos de ejemplo.

$$R_2: \quad B_{1,1} \quad \Leftrightarrow \quad (H_{1,2} \vee H_{2,1})$$

$$R_3: \quad B_{2,1} \quad \Leftrightarrow \quad (H_{1,1} \vee H_{2,2} \vee H_{3,1})$$

Por lo tanto nuestra base de conocimiento BC es la conjuncion R_1^R_2^R_3^...^R_n

Inferencia

Recordemos que el objetivo de la inferencia lógica es decidir si la base de conocimiento implica α para alguna sentencia α. Por ejemplo, si se deduce H2,2. Nuestro primer algoritmo para la inferencia será una implementación directa del concepto de implicación. En la lógica proposicional los modelos son asignaciones de los valores de verdadero y falso sobre cada símbolo proposicional.

Algunas formas de argumentos básicas y derivadas

Nombre	Consecuente	Descripción
Modus ponens	$((p \to q) \wedge p) \vdash q$	Si p entonces q; p; por lo tanto q
Modus tollens	$((p \to q) \wedge \neg q) \vdash \neg p$	Si p entonces q; no q; por lo tanto no p
Silogismo hipotético	$((p \to q) \wedge (q \to r)) \vdash (p \to r)$	Si p entonces q; si q entonces r; por lo tanto, si p entonces r
Silogismo disyuntivo	$((p \vee q) \wedge \neg p) \vdash q$	Si p o q, Y; no p; por lo tanto, q
Dilema constructivo	$((p \to q) \wedge (r \to s) \wedge (p \vee r)) \vdash (q \vee s)$	Si p entonces q; y si r entonces s; pero p o r; por lo tanto q o s
Dilema destructivo	$((p \to q) \wedge (r \to s) \wedge (\neg q \vee \neg s)) \vdash (\neg p \vee \neg r)$	Si p entonces q; y si r entonces s; pero no q o no s; por lo tanto no p o no r
Dilema bidireccional	$((p \to q) \wedge (r \to s) \wedge (p \vee \neg s)) \vdash (q \vee \neg r)$	Si p entonces q; y si r entonces s; pero p o no s; por lo tanto q o no r
Simplificación	$(p \wedge q) \vdash p$	p y q son verdaderos; por lo tanto p es verdadero
Conjunción	$p, q \vdash (p \wedge q)$	p y q son verdaderos separadamente; entonces son verdaderos conjuntamente.
Adición	$p \vdash (p \vee q)$	p es verdadero; por lo tanto la disyunción (p o q) es verdadera
Composición	$((p \to q) \wedge (p \to r)) \vdash (p \to (q \wedge r))$	Si p entonces q; y si p entonces r; por lo tanto si p es verdadero entonces q y r son verdaderos
Ley de De Morgan (1)	$\neg(p \wedge q) \vdash (\neg p \vee \neg q)$	La negación de (p y q) es equivalente a (no p o no q)
Ley de De Morgan (2)	$\neg(p \vee q) \vdash (\neg p \wedge \neg q)$	La negación de (p o q) es equivalente a (no p y no q)
Conmutación (1)	$(p \vee q) \vdash (q \vee p)$	(p o q) es equivalente a (q o p)
Conmutación (2)	$(p \wedge q) \vdash (q \wedge p)$	(p y q) es equivalente a (q y p)
Conmutación (3)	$(p \leftrightarrow q) \vdash (q \leftrightarrow p)$	(p es equivalente a q) es equivalente a (q es equivalente a p)
Asociación (1)	$(p \vee (q \vee r)) \vdash ((p \vee q) \vee r)$	p o (q o r) es equivalente a (p o q) o r
Asociación (2)	$(p \wedge (q \wedge r)) \vdash ((p \wedge q) \wedge r)$	p y (q y r) es equivalente a (p y q) y r
Distribución (1)	$(p \wedge (q \vee r)) \vdash ((p \wedge q) \vee (p \wedge r))$	p y (q o r) es equivalente a (p y q) o (p y r)
Distribución (2)	$(p \vee (q \wedge r)) \vdash ((p \vee q) \wedge (p \vee r))$	p o (q y r) es equivalente a (p o q) y (p o r)
Doble negación	$p \vdash \neg\neg p$	p es equivalente a la negación de no p
Transposición	$(p \to q) \vdash (\neg q \to \neg p)$	Si p entonces q es equivalente a si no q entonces no p
Implicación material	$(p \to q) \vdash (\neg p \vee q)$	Si p entonces q es equivalente a no p o q