Equivalencia lógica

Semana 5

Edgar Andrade, PhD

Última revisión: Febrero de 2022

Matemáticas Aplicadas y Ciencias de la Computación

Presentación

En esta sesión estudiaremos:

1. Equivalencia lógica
2. Teorema de sustitución salva veritate
3. Eliminación de conectivos

Presentación

Equivalencia

Sean A, B, fórmulas. La equivalencia entre A y B ($A \equiv B$) se define de la siguiente manera:

$$A \equiv B \Leftrightarrow A.valor(I) = B.valor(I) \text{ para toda interpretación } I$$

Ejemplo

Proposición: $p \equiv \neg\neg p$

Ejemplo

Proposición: $p \equiv \neg\neg p$

Demostración:

Sea I una interpretación arbitraria. Tenemos dos casos:

Ejemplo

Proposición: $p \equiv \neg\neg p$

Demostración:

Sea I una interpretación arbitraria. Tenemos dos casos:

Caso 1: Supongamos que `Letra('p')`.$valor(I) =$ `True`. Luego `Negacion(Letra('p'))`.$valor(I) =$ `not True` $=$ `False` y entonces `Negacion(Negacion(Letra('p')))`.$valor(I) =$ `not False` $=$ `True`. Por lo tanto `Letra('p')`.$valor(I) =$ `Negacion(Negacion(Letra('p')))`.$valor(I)$.

Ejemplo

Proposición: $p \equiv \neg\neg p$

Demostración:

Sea I una interpretación arbitraria. Tenemos dos casos:

Caso 1: ...

Caso 2: Supongamos que `Letra('p')`.$valor(I) =$ `False`. Luego `Negacion(Letra('p'))`.$valor(I) =$ `not False` $=$ `True` y entonces `Negacion(Negacion(Letra('p')))`.$valor(I) =$ `not True` $=$ `False`. Por lo tanto `Letra('p')`.$valor(I) =$ `Negacion(Negacion(Letra('p')))`.$valor(I)$.

Ejemplo

Proposición: $p \equiv \neg\neg p$

Demostración:

Sea I una interpretación arbitraria. Tenemos dos casos:

Caso 1: ...

Caso 2: ...

En cualquier caso, $\texttt{Letra('p')}.valor(I) = \texttt{Negacion(Negacion(Letra('p')))}.valor(I)$.

Ejemplo

Proposición: $p \equiv \neg\neg p$

Demostración:

Sea I una interpretación arbitraria. Tenemos dos casos:

Caso 1: ...

Caso 2: ...

En cualquier caso, `Letra('p')`.$valor(I) =$ `Negacion(Negacion(Letra('p')))`.$valor(I)$.

Como I es arbitraria, se sigue que $p \equiv \neg\neg p$.

Equivalencias importantes

Proposición: Las siguientes equivalencias son ciertas:

- $(p \to q) \equiv (\neg p \lor q)$
- $\neg(p \land q) \equiv (\neg p \lor \neg q)$
- $\neg(p \lor q) \equiv (\neg p \land \neg q)$
- $\big(p \lor (q \land r)\big) \equiv \big((p \lor q) \land (p \lor r)\big)$
- $\big(p \land (q \lor r)\big) \equiv \big((p \land q) \lor (p \land r)\big)$

Implementación en Python

Ir al notebook “Equivalencia_lógica” y revisar la sección 1.

Lemas importantes

Lema (I)
Sean A y B fórmulas. Si $A \equiv B$, entonces $\neg A \equiv \neg B$.

Lema (II)
Sean A, B A' y B' fórmulas. Si $A \equiv A'$ y $B \equiv B'$, entonces $A \odot B \equiv A' \odot B'$, para $\odot \in \{\wedge, \vee \rightarrow, \leftrightarrow\}$.

Presentación

Sustitución

Sea B una fórmula y $A \in B$.subforms(). Sea A' una fórmula. Definimos $B\{A \leftarrow A'\} = \text{B.sust}(A, A')$, donde:

función sust(`self`, A, A'):
  Si $A \notin$ `self`.subforms()
    retornar `self`
  Si no, si A es `self`
    retornar A'
  Si no, si `self` es de tipo `Negacion`
    retornar `Negacion`(`self`.subf.sust(A, A'))
  Si no, si `self` es de tipo `Binario`
    retornar `Binario`(`self`.conectivo,
      `self`.left.sust(A, A'), `self`.right.sust(A, A'))

Más lemas importantes

Lema (III)
Sea B una fórmula y supongamos que $A \in$ B.subforms() tal que $A \neq B$. Entonces

$$\neg B\{A \leftarrow A'\} = \neg(B\{A \leftarrow A'\})$$

Lema (IV)
Sean B y C fórmulas y supongamos que $A \in (B \odot C)$.subforms() tal que $A \neq (B \odot C)$. Entonces

$$(B \odot C)\{A \leftarrow A'\} = B\{A \leftarrow A'\} \odot C\{A \leftarrow A'\}$$

para $\odot \in \{\wedge, \vee \rightarrow, \leftrightarrow\}$.

Sustitución salva veritate

Teorema
Sea B una fórmula y $A \in B$.subforms(). Sea A' una fórmula. Si $A \equiv A'$, entonces $B \equiv B\{A \leftarrow A'\}$.

Sustitución salva veritate

Demostración: Por inducción estructural sobre B.

Sustitución salva veritate

Demostración: Por inducción estructural sobre B.

- Caso $B = \texttt{Letra('p')}$:

Sustitución salva veritate

Demostración: Por inducción estructural sobre B.

- Caso $B = \texttt{Letra('p')}$: Observe que $A \in B$.subforms() y en consecuencia $A = B$. A partir de esto se tienen dos cosas:

Sustitución salva veritate

Demostración: Por inducción estructural sobre B.

- Caso $B =$ `Letra('p')`: Observe que $A \in B$.subforms() y en consecuencia $A = B$. A partir de esto se tienen dos cosas:
    1. $B \equiv A'$, porque hemos asumido que $A \equiv A'$.

Sustitución salva veritate

Demostración: Por inducción estructural sobre B.

- Caso $B = \texttt{Letra('p')}$: Observe que $A \in B$.subforms() y en consecuencia $A = B$. A partir de esto se tienen dos cosas:
    1. $B \equiv A'$, porque hemos asumido que $A \equiv A'$.
    2. $B.\text{sust}(A, A') = A'$, por definición de sust.

Sustitución salva veritate

Demostración: Por inducción estructural sobre B.

- Caso $B = \texttt{Letra('p')}$: Observe que $A \in B$.subforms() y en consecuencia $A = B$. A partir de esto se tienen dos cosas:
  1. $B \equiv A'$, porque hemos asumido que $A \equiv A'$.
  2. B.sust$(A, A') = A'$, por definición de sust.

  Por lo tanto, $B \equiv B$.sust(A, A'). Es decir, $B \equiv B\{A \leftarrow A'\}$.

Sustitución salva veritate

- Caso $B = \texttt{Negacion(}C\texttt{)}$. Asumimos que si $A \in C$.subforms() y $A \equiv A'$, entonces $C \equiv C\{A \leftarrow A'\}$.

Sustitución salva veritate

- Caso $B = \texttt{Negacion(}C\texttt{)}$. Asumimos que si $A \in C$.subforms() y $A \equiv A'$, entonces $C \equiv C\{A \leftarrow A'\}$. Supongamos primero que $A = B$. Entonces $B\{A \leftarrow A'\} = A'$. Como hemos supuesto que $A \equiv A'$, se sigue que $B \equiv B\{A \leftarrow A'\}$.

Sustitución salva veritate

- Caso $B = \texttt{Negacion(}C\texttt{)}$. Asumimos que si $A \in C$.subforms() y $A \equiv A'$, entonces $C \equiv C\{A \leftarrow A'\}$. Supongamos primero que $A = B$. Entonces $B\{A \leftarrow A'\} = A'$. Como hemos supuesto que $A \equiv A'$, se sigue que $B \equiv B\{A \leftarrow A'\}$. Por otro lado, supongamos que $A \neq B$. Ahora, como $A \in B$.subforms(), se tiene que $A \in C$.subforms().

Sustitución salva veritate

- Caso $B = \texttt{Negacion}(C)$. Asumimos que si $A \in C$.subforms() y $A \equiv A'$, entonces $C \equiv C\{A \leftarrow A'\}$. Supongamos primero que $A = B$. Entonces $B\{A \leftarrow A'\} = A'$. Como hemos supuesto que $A \equiv A'$, se sigue que $B \equiv B\{A \leftarrow A'\}$. Por otro lado, supongamos que $A \neq B$. Ahora, como $A \in B$.subforms(), se tiene que $A \in C$.subforms(). Como $A \equiv A'$, entonces por la hipótesis de inducción tenemos que $C \equiv C\{A \leftarrow A'\}$, y por el lema I tenemos que $\texttt{Negacion}(\mathrm{C}) \equiv \texttt{Negacion}(C\{A \leftarrow A'\})$.

Sustitución salva veritate

- Caso $B = \texttt{Negacion}(C)$. Asumimos que si $A \in C$.subforms() y $A \equiv A'$, entonces $C \equiv C\{A \leftarrow A'\}$. Supongamos primero que $A = B$. Entonces $B\{A \leftarrow A'\} = A'$. Como hemos supuesto que $A \equiv A'$, se sigue que $B \equiv B\{A \leftarrow A'\}$. Por otro lado, supongamos que $A \neq B$. Ahora, como $A \in B$.subforms(), se tiene que $A \in C$.subforms(). Como $A \equiv A'$, entonces por la hipótesis de inducción tenemos que $C \equiv C\{A \leftarrow A'\}$, y por el lema I tenemos que $\texttt{Negacion}(\mathrm{C}) \equiv \texttt{Negacion}(C\{A \leftarrow A'\})$. Por el lema III tenemos que $\texttt{Negacion}(\mathrm{C})\{A \leftarrow A'\} = \texttt{Negacion}(C\{A \leftarrow A'\})$.

Sustitución salva veritate

- Caso $B = \texttt{Negacion}(C)$. Asumimos que si $A \in C$.subforms() y $A \equiv A'$, entonces $C \equiv C\{A \leftarrow A'\}$. Supongamos primero que $A = B$. Entonces $B\{A \leftarrow A'\} = A'$. Como hemos supuesto que $A \equiv A'$, se sigue que $B \equiv B\{A \leftarrow A'\}$. Por otro lado, supongamos que $A \neq B$. Ahora, como $A \in B$.subforms(), se tiene que $A \in C$.subforms(). Como $A \equiv A'$, entonces por la hipótesis de inducción tenemos que $C \equiv C\{A \leftarrow A'\}$, y por el lema I tenemos que $\texttt{Negacion}(\mathrm{C}) \equiv \texttt{Negacion}(C\{A \leftarrow A'\})$. Por el lema III tenemos que $\texttt{Negacion}(\mathrm{C})\{A \leftarrow A'\} = \texttt{Negacion}(C\{A \leftarrow A'\})$. En consecuencia, $\texttt{Negacion}(\mathrm{C}) \equiv \texttt{Negacion}C\{A \leftarrow A'\}$. Por definición de B se sigue que $B \equiv B\{A \leftarrow A'\}$.

Sustitución salva veritate

Demostración: Por inducción estructural sobre B.

- Caso $B = \texttt{Binario}(\odot$, C, D), donde $C \equiv C\{A \leftarrow A'\}$ y $D \equiv D\{A \leftarrow A'\}$:

Sustitución salva veritate

Demostración: Por inducción estructural sobre B.

- Caso $B = \texttt{Binario}(\odot$, C, D), donde $C \equiv C\{A \leftarrow A'\}$ y $D \equiv D\{A \leftarrow A'\}$: ☞ Ejercicio.

Presentación

Eliminando implicaciones

Teorema
Sea A una fórmula. A es equivalente a una fórmula A' en la que no hay ocurrencias del conectivo '$\to$'.

Eliminando implicaciones

Teorema
Sea A una fórmula. A es equivalente a una fórmula A' en la que no hay ocurrencias del conectivo '$\rightarrow$'.

Demostración: Supongamos que existe $B \rightarrow C \in \text{Subform}(A)$ para alguna fórmula B y alguna fórmula C.

Eliminando implicaciones

Teorema
Sea A una fórmula. A es equivalente a una fórmula A' en la que no hay ocurrencias del conectivo '$\rightarrow$'.

Demostración: Supongamos que existe $B \rightarrow C \in \text{Subform}(A)$ para alguna fórmula B y alguna fórmula C. Observe que
$B \rightarrow C \equiv \neg B \vee C$.

Eliminando implicaciones

Teorema
Sea A una fórmula. A es equivalente a una fórmula A' en la que no hay ocurrencias del conectivo '$\to$'.

Demostración: Supongamos que existe $B \to C \in \mathrm{Subform}(A)$ para alguna fórmula B y alguna fórmula C. Observe que $B \to C \equiv \neg B \vee C$. Por el teorema de sustitución salva veritate se sigue que $A \equiv A\{B \to C, \neg B \vee C\}$.

Eliminando implicaciones

Teorema
Sea A una fórmula. A es equivalente a una fórmula A' en la que no hay ocurrencias del conectivo '$\rightarrow$'.

Demostración: Supongamos que existe $B \rightarrow C \in \text{Subform}(A)$ para alguna fórmula B y alguna fórmula C. Observe que $B \rightarrow C \equiv \neg B \vee C$. Por el teorema de sustitución salva veritate se sigue que $A \equiv A\{B \rightarrow C, \neg B \vee C\}$. En consecuencia, cualquier ocurrencia del conectivo '$\rightarrow$' puede eliminarse de A, obteniendo una fórmula equivalente. Así pues, una cadena finita de sustituciones nos proporcionará una fórmula A' equivalente a A que no contiene ocurrencias de '$\rightarrow$'.

Eliminando dobles negaciones

Teorema
Sea A una fórmula. A es equivalente a una fórmula A' en la que no hay ocurrencias de la doble negación '$\neg\neg$'.

Forma normal conjuntiva (1/3)

Definiciones:

- Un literal es una letra proposicional o la negación de una letra proposicional.

Forma normal conjuntiva (1/3)

Definiciones:

- Un literal es una letra proposicional o la negación de una letra proposicional.
- Una cláusula es una disyunción de literales.

Forma normal conjuntiva (1/3)

Definiciones:

- Un literal es una letra proposicional o la negación de una letra proposicional.
- Una cláusula es una disyunción de literales.
- Una fórmula está en *forma normal conjuntiva* si es una conjunción de cláusulas.

Forma normal conjuntiva (1/3)

Definiciones:

- Un literal es una letra proposicional o la negación de una letra proposicional.
- Una cláusula es una disyunción de literales.
- Una fórmula está en *forma normal conjuntiva* si es una conjunción de cláusulas.

Teorema
Sea A una fórmula. A es equivalente a una fórmula A' en forma normal conjuntiva.

Forma normal conjuntiva (2/3)

Procedimiento para transformar una fórmula arbitraria A en una fórmula A' en forma normal conjuntiva, tal que $A \equiv A'$:

Forma normal conjuntiva (2/3)

Procedimiento para transformar una fórmula arbitraria A en una fórmula A' en forma normal conjuntiva, tal que $A \equiv A'$:

1. Eliminar '$\leftrightarrow$' y '$\rightarrow$'.

Forma normal conjuntiva (2/3)

Procedimiento para transformar una fórmula arbitraria A en una fórmula A' en forma normal conjuntiva, tal que $A \equiv A'$:

1. Eliminar '$\leftrightarrow$' y '$\rightarrow$'.
2. Eliminar dobles negaciones.

Forma normal conjuntiva (2/3)

Procedimiento para transformar una fórmula arbitraria A en una fórmula A' en forma normal conjuntiva, tal que $A \equiv A'$:

1. Eliminar '$\leftrightarrow$' y '$\rightarrow$'.
2. Eliminar dobles negaciones.
3. Si $\neg(B \wedge C) \in A$.subform(), reemplazarla por $\neg B \vee \neg C$.

Forma normal conjuntiva (2/3)

Procedimiento para transformar una fórmula arbitraria A en una fórmula A' en forma normal conjuntiva, tal que $A \equiv A'$:

1. Eliminar '$\leftrightarrow$' y '$\rightarrow$'.
2. Eliminar dobles negaciones.
3. Si $\neg(B \wedge C) \in A$.subform(), reemplazarla por $\neg B \vee \neg C$.
4. Si $\neg(B \vee C) \in A$.subform(), reemplazarla por $\neg B \wedge \neg C$.

Forma normal conjuntiva (2/3)

Procedimiento para transformar una fórmula arbitraria A en una fórmula A' en forma normal conjuntiva, tal que $A \equiv A'$:

1. Eliminar '$\leftrightarrow$' y '$\rightarrow$'.
2. Eliminar dobles negaciones.
3. Si $\neg(B \wedge C) \in A$.subform(), reemplazarla por $\neg B \vee \neg C$.
4. Si $\neg(B \vee C) \in A$.subform(), reemplazarla por $\neg B \wedge \neg C$.
5. Eliminar dobles negaciones.

Forma normal conjuntiva (2/3)

Procedimiento para transformar una fórmula arbitraria A en una fórmula A' en forma normal conjuntiva, tal que $A \equiv A'$:

1. Eliminar '$\leftrightarrow$' y '$\rightarrow$'.
2. Eliminar dobles negaciones.
3. Si $\neg(B \wedge C) \in A$.subform(), reemplazarla por $\neg B \vee \neg C$.
4. Si $\neg(B \vee C) \in A$.subform(), reemplazarla por $\neg B \wedge \neg C$.
5. Eliminar dobles negaciones.
6. Si $B \vee (C \wedge D) \in A$.subform(), reemplazarla por $(B \vee C) \wedge (B \vee D)$.

Forma normal conjuntiva (3/3)

Ejemplo: Transformar $(p \vee q) \rightarrow (r \wedge \neg s)$ en su forma normal conjuntiva.

Forma normal conjuntiva (3/3)

Ejemplo: Transformar $(p \vee q) \rightarrow (r \wedge \neg s)$ en su forma normal conjuntiva.

- $\neg(p \vee q) \vee (r \wedge \neg s)$ (eliminación de '$\rightarrow$')

Forma normal conjuntiva (3/3)

Ejemplo: Transformar $(p \vee q) \rightarrow (r \wedge \neg s)$ en su forma normal conjuntiva.

- $\neg(p \vee q) \vee (r \wedge \neg s)$ (eliminación de '$\rightarrow$')
- $(\neg p \wedge \neg q) \vee (r \wedge \neg s)$ (Moviendo '$\neg$' a la derecha)
- $\big((\neg p \wedge \neg q) \vee r\big) \wedge \big((\neg p \wedge \neg q) \vee \neg s\big)$ (distribución de '$\vee$' sobre '$\wedge$')

Forma normal conjuntiva (3/3)

Ejemplo: Transformar $(p \vee q) \rightarrow (r \wedge \neg s)$ en su forma normal conjuntiva.

- $\neg(p \vee q) \vee (r \wedge \neg s)$ (eliminación de '$\rightarrow$')
- $(\neg p \wedge \neg q) \vee (r \wedge \neg s)$ (Moviendo '$\neg$' a la derecha)
- $\big((\neg p \wedge \neg q) \vee r\big) \wedge \big((\neg p \wedge \neg q) \vee \neg s\big)$ (distribución de '$\vee$' sobre '$\wedge$')

Forma normal conjuntiva (3/3)

Ejemplo: Transformar $(p \vee q) \to (r \wedge \neg s)$ en su forma normal conjuntiva.

- $\neg(p \vee q) \vee (r \wedge \neg s)$ (eliminación de '$\to$')
- $(\neg p \wedge \neg q) \vee (r \wedge \neg s)$ (Moviendo '$\neg$' a la derecha)
- $\big((\neg p \wedge \neg q) \vee r\big) \wedge \big((\neg p \wedge \neg q) \vee \neg s\big)$ (distribución de '$\vee$' sobre '$\wedge$')
- $\big((\neg p \vee r) \wedge (\neg q \vee r)\big) \wedge \big((\neg p \wedge \neg q) \vee \neg s\big)$ (idem)

Forma normal conjuntiva (3/3)

Ejemplo: Transformar $(p \vee q) \rightarrow (r \wedge \neg s)$ en su forma normal conjuntiva.

- $\neg(p \vee q) \vee (r \wedge \neg s)$ (eliminación de '$\rightarrow$')
- $(\neg p \wedge \neg q) \vee (r \wedge \neg s)$ (Moviendo '$\neg$' a la derecha)
- $\big((\neg p \wedge \neg q) \vee r\big) \wedge \big((\neg p \wedge \neg q) \vee \neg s\big)$ (distribución de '$\vee$' sobre '$\wedge$')
- $\big((\neg p \vee r) \wedge (\neg q \vee r)\big) \wedge \big((\neg p \wedge \neg q) \vee \neg s\big)$ (idem)
- $\big((\neg p \vee r) \wedge (\neg q \vee r)\big) \wedge \big((\neg p \vee \neg s) \wedge (\neg q \vee \neg s)\big)$ (idem)

Fin de la sesión 5

En esta sesión usted ha aprendido:

1. Comprender el concepto de equivalencia lógica
2. Demostrar el teorema de equivalencia salva veritate
3. Intercambiar conectivos lógicos por otros manteniendo la equivalencia