Métodos dedutivos e inferência lógica

Eduardo Furlan Miranda

2024-08-01

Baseado em: SCHEFFER, VC; VIEIRA, G; LIMA, TPFS. Lógica Computacional. EDE, 2020. ISBN 978-85-522-1688-9.

$$P_1 \land P_2 \land P_3 \land \ldots \land P_n$$

- A lógica proposicional é composta por proposições e conectivos lógicos que permitem criar fórmulas que quando escritas corretamente são chamadas fbf

- Uma fbf é valorada em verdadeira (V) ou falsa (F), respeitando a ordem de precedência dos operadores lógicos

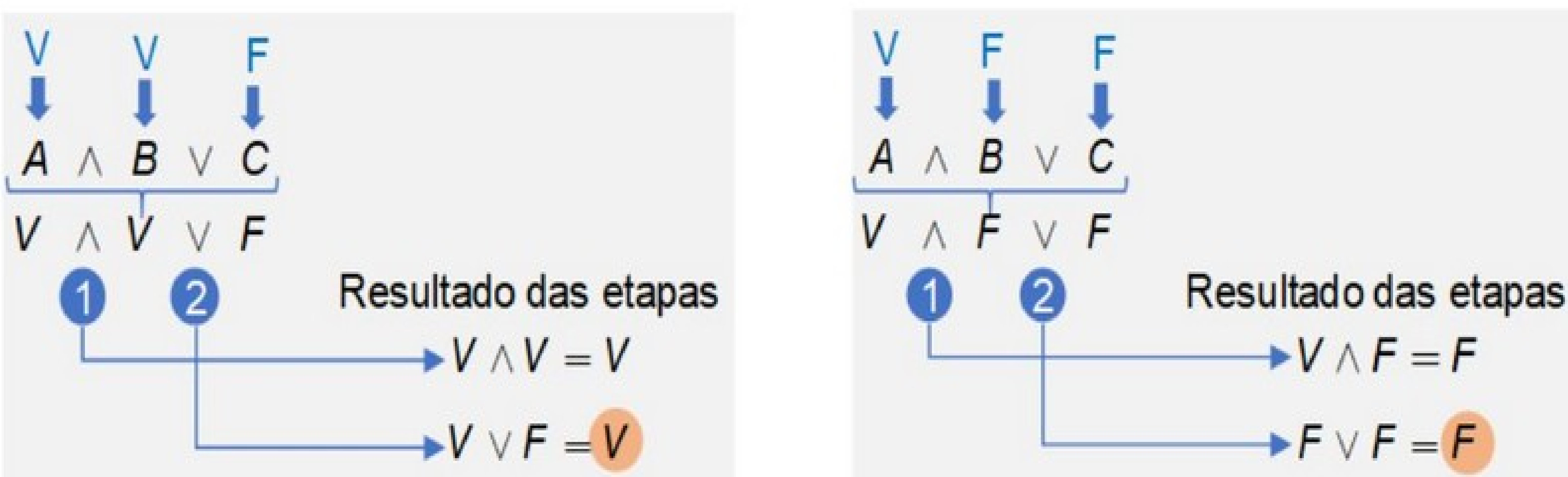
- A valoração de uma fórmula também depende dos valores lógicos de entrada para cada uma das proposições

- Dada a fórmula $A \wedge B \vee C$, e as entradas $A = V$, $B = V$, $C = F$, ela será verdadeira ou falsa?

Figura 3.8 | Valoração para a fórmula $A \wedge B \vee C$

a) A = V, B = V, C = F

b) A = V, B = F, C = F



- Quando uma fórmula apresenta um conjunto de proposições, das quais uma delas é uma conclusão, dizemos que tal fórmula é um **argumento**

- Um argumento é um conjunto de proposições ou fórmulas,
  - nas quais uma delas (conclusão) deriva,
    - ou é consequência das outras (premissas)

Concectivo de conjunção

Conectivo condicional

- Representação de forma simbólica do argumento:

$$P_1 \land P_2 \land P_3 \land \ldots \land P_n \rightarrow C$$

hipóteses

conclusão do argumento

```python
P1 = False
P2 = P3 = P4 = True
if P1 and P2 and P3 and P4 :
    print("Verdadeiro")
else :
    print ("Falso")
```

```
Falso
```

- P, C, podem ser uma proposição simples ou uma fbf

Conectivo condicional

condição suficiente

condição necessária

x <= y

P	Q	P → Q
1	1	1
1	0	0
0	1	1
0	0	1

se eu já estou dizendo **que estou mentindo**, então eu posso dizer qualquer coisa?

Tabela verdade

and or x <= y

P	Q	∧	∨	→
1	1	1	1	1
1	0	0	1	0
0	1	0	1	1
0	0	0	0	1

Possui uma conclusão

- Dado um argumento,
  - é importante validar se ele é válido ou inválido

- A lógica possui mecanismos que permitem validá-los
  - compostos pelas regras de equivalência e inferência lógica
    - permite avaliar a relação entre as
      - hipóteses, e a
      - conclusão
        - também chamada de consequência lógica, dedução lógica, conclusão lógica ou implicação lógica

- Uma proposição pode ser
  - verdadeira ou falsa
    - e não pode ser válida ou inválida

$P1 \wedge P2 \wedge P3 \wedge \ldots \wedge Pn$

- Um argumento pode ser
  - válido ou inválido
  - e não pode ser verdadeiro ou falso

$P_1 \wedge P_2 \wedge P_3 \wedge \ldots \wedge P_n \rightarrow C$

```
if P1 and P2 and P3 and P4 :
    print("Verdadeiro")
```

Exemplo

- D. Pedro I proclamou a independência do Brasil e Thomas Jefferson escreveu a Declaração de Independência dos Estados Unidos. Portanto, todo dia tem 24 horas
  - Separando as proposições do argumento em hipóteses e conclusão:
    - A : D. Pedro I proclamou a independência do Brasil
    - B : Thomas Jefferson escreveu a Declaração de Independência dos Estados Unidos
    - C : Todo dia tem 24 horas

(continua)

(continuação)

Exemplo

- Nosso conhecimento nos permite valorar as três proposições, logo, A, B, C são todas verdadeiras

- Porém o argumento é inválido, pois a conclusão nada tem a ver com as hipóteses

nós, seres humanos, sabemos disso

(continua)

Exemplo

(continuação)

- Segundo a Lógica Formal, devemos nos basear apenas nas regras para validar um argumento e não no conteúdo

$$A \wedge B \rightarrow C$$

- Traduzindo o argumento:
  - D. Pedro I proclamou a independência do Brasil, e
  - Thomas Jefferson escreveu a Declaração de Independência dos Estados Unidos

- Portanto todo dia tem 24 horas para a notação simbólica

Tautologia

$$A \wedge B \rightarrow C$$

- Quando o valor lógico de entrada da proposição A for verdadeiro, e de B for falso, o resultado da implicação(conclusão) será falso
  - Existe pelo menos uma combinação de entradas, para a qual a fórmula resultará em falsa, logo essa fórmula não é uma tautologia e, consequentemente, não é um argumento válido
    - Tautologia é um resultado no qual todas as entradas possíveis de uma fórmula obtêm verdadeiro como resultado

- Um argumento só é válido quando a fórmula é uma tautologia

- Para saber se um argumento é válido ou não, precisamos saber se ele é uma tautologia
  - Poderíamos testar todas as combinações de entrada possíveis para o argumento
    - Na Lógica Formal podemos usar um sistema de regras de dedução e,
      - seguindo uma **sequência de demonstração**
        - provar se o argumento é válido ou não

Sequência de demonstração

- É uma sequência de fbfs,
  - nas quais cada fbf é uma hipótese, ou
    - o resultado de se aplicar uma das regras de dedução do sistema formal a fbfs anteriores na sequência

Lógica Formal

P_1 , P_2 , ..., P_n , são as hipóteses

Sequência de demonstração

$$P1 \wedge P2 \wedge P3 \wedge \ldots \wedge Pn \rightarrow C$$

1. P_1	(hipótese)
2. P_2	(hipótese)
...	...
n) P_n	(hipótese)
n+1) fbf_1	(resultado da aplicação de uma regra de dedução a hipóteses anteriores.)
n+2) fbf_2	(resultado da aplicação de uma regra de dedução a hipóteses anteriores.)
...	...
n+n) C	(resultado da aplicação de uma regra de dedução a hipóteses anteriores.)

Da tabela anterior:

- Na sequência de demonstração cada proposição deve ficar em uma linha
- Enumeramos para facilitar na hora de aplicar as regras de dedução
- Na frente de cada linha devemos indicar o que ela representa,
  - se é uma hipótese ou então a regra que foi aplicada
- Após elencar todas as proposições é hora de começar a aplicar as regras e, consequentemente, obter novas fbfs

- A quantidade varia de caso para caso, mas as regras de dedução devem ser aplicadas até que se consiga
  - provar que o argumento é verdadeiro, ou
  - que não existam mais regras a serem aplicadas
    - e neste caso o argumento é falso

Regras de **equivalência** de dedução para a Lógica Proposicional

$P_1 \wedge P_2 \wedge P_3 \wedge \ldots \wedge P_n$

- As regras de dedução são divididas em dois tipos
  - Regras de equivalência
  - Regras de inferência
- 2 fbfs são equivalentes quando todas as combinações possíveis de entradas geram o mesmo resultado de saída
- Regras de equivalência serão usadas quando uma fbf (que pode ser uma hipótese ou resultado de uma regra) pode ser substituída por outra fbf, mantendo o resultado lógico

Exemplo - equivalência

- Seja a fbf que traduz uma das leis de De Morgan:
  - $\neg (A \lor B) \Leftrightarrow \neg A \land \neg B$, em uma situação adequada podemos substituir a fbf $\neg(A \lor B)$ por $\neg A \land \neg B$, pois ambas são equivalentes

- O quadro do slide a seguir traz 6 conjuntos de regras de dedução
  - Se tivermos uma expressão como da linha 1, $P \lor Q$, quando necessário, podemos substituí-la por $Q \lor P$
  - O contrário também é válido, quando aparecer $Q \lor P$, podemos substituir por $P \lor Q$

Regras de equivalência

Expressão (fbf)	Equivalente (fbf)	Nome/Abreviação
$P \vee Q$ $P \wedge Q$	$Q \vee P$ $Q \wedge P$	Comutatividade/com
$(P \vee Q) \vee R$ $(P \wedge Q) \wedge R$	$P \vee (Q \vee R)$ $P \wedge (Q \wedge R)$	Associatividade/ass
$\neg(P \vee Q)$ $\neg(P \wedge Q)$	$\neg P \wedge \neg Q$ $\neg P \vee \neg Q$	Leis de De Morgan/De Morgan
$P \rightarrow Q$	$\neg P \vee Q$	Condicional/cond
P	$\leftarrow(\leftarrow P)$	Dupla negação/dn
$P \leftrightarrow Q$	$(P \rightarrow Q) \wedge (Q \rightarrow P)$	Definição de equivalência/que

Regras de inferência de dedução para a Lógica Proposicional

- Dada uma determinada fbf, ela poderá ser substituída por outra que atenda uma regra de inferência
  - Não é necessário ser uma tautologia (sempre verdadeira)

- Regras de inferência para condicionais
  - **Modus Ponens** (ou eliminação do condicional) é uma válida e simples forma de argumento e regra de inferência

Representação

$$\frac{\begin{array}{c} P \longrightarrow Q \\ P \end{array}}{Q}$$

(mais detalhes nos próx. slides)

Modus Ponens (MP) (tautologia)

- Poder ser representado como a declaração de uma tautologia ou teorema da lógica proposicional:

$$((P \rightarrow Q) \wedge P) \rightarrow Q$$

x <= y and x <= y (vide tab. do slide 10)

		x <= y	and	x <= y
P	Q	P → Q	∧ P	→ Q
1	1	1	1	1
1	0	0	0	1
0	1	1	0	1
0	0	1	0	1

Modus Ponens (MP)

$$((P \to Q) \land P) \to Q$$

```
for P in [True, False]:
    for Q in [True, False]:
        R1 = (P <= Q)
        R2 = (R1 and P)
        print(f"{P:6}{Q:6}{R1:6}{R2:6}{R2<=Q:6}")
```

```
     1     1     1     1     1
     1     0     0     0     1
     0     1     1     0     1
     0     0     1     0     1
```

Exemplo - Modus Ponens

- Argumento: Se João receber seu salário, ele irá ao cinema

- Separando as proposições P, Q:
  - P: João recebe o salário
  - Q: João vai ao cinema

- Representação de cada parte da fórmula de Modus Ponens:

I. ($P \rightarrow Q$): Se João receber seu salário, ele irá ao cinema

II. P: **João recebe o salário** (fato)

(continua)

Exemplo - Modus Ponens

(continuação)

$\wedge$

- Ao fazer a conjunção entre a primeira parte com a segunda, conseguimos inferir a conclusão, pois, se João receber seu salário, ele irá ao cinema, E João recebeu o salário, logo podemos inferir(concluir) que João vai ao cinema

- A regra de Modus Ponens também pode ser representada:

Verdade

Resulta na conclusão

$$\begin{array}{c} P \rightarrow Q \\ P \\ \hline Q \end{array}$$

(fato)

na implicação se o antecedente é verdade, então a conclusão é o consequente

Modus Tollens (MT)

$\rightarrow$ $\wedge$

- Além de envolver uma implicação e uma conjunção, também envolve a negação de uma das proposições

$\neg$

- Sua estrutura é dada pela fbf

$$(P \rightarrow Q) \wedge \neg Q \rightarrow \neg P$$

$$\begin{array}{c} P \longrightarrow Q \\ \neg Q \\ \hline \neg P \end{array}$$

- Se na implicação o consequente não é verdade, então a conclusão é que o antecedente também não aconteceu

$V \rightarrow V$

Exemplo

- Se João desligar o interruptor, então a lâmpada se apaga

- Separar as proposições P, Q:
  - P: João desliga o interruptor.
  - Q: A lâmpada apaga.

- Representando cada parte da fórmula de Modus Tollens:

I. $(P \rightarrow Q)$: Se João desligar o interruptor, então a lâmpada se apaga

(continua)

II. $\neg Q$: A lâmpada não apagou

Exemplo

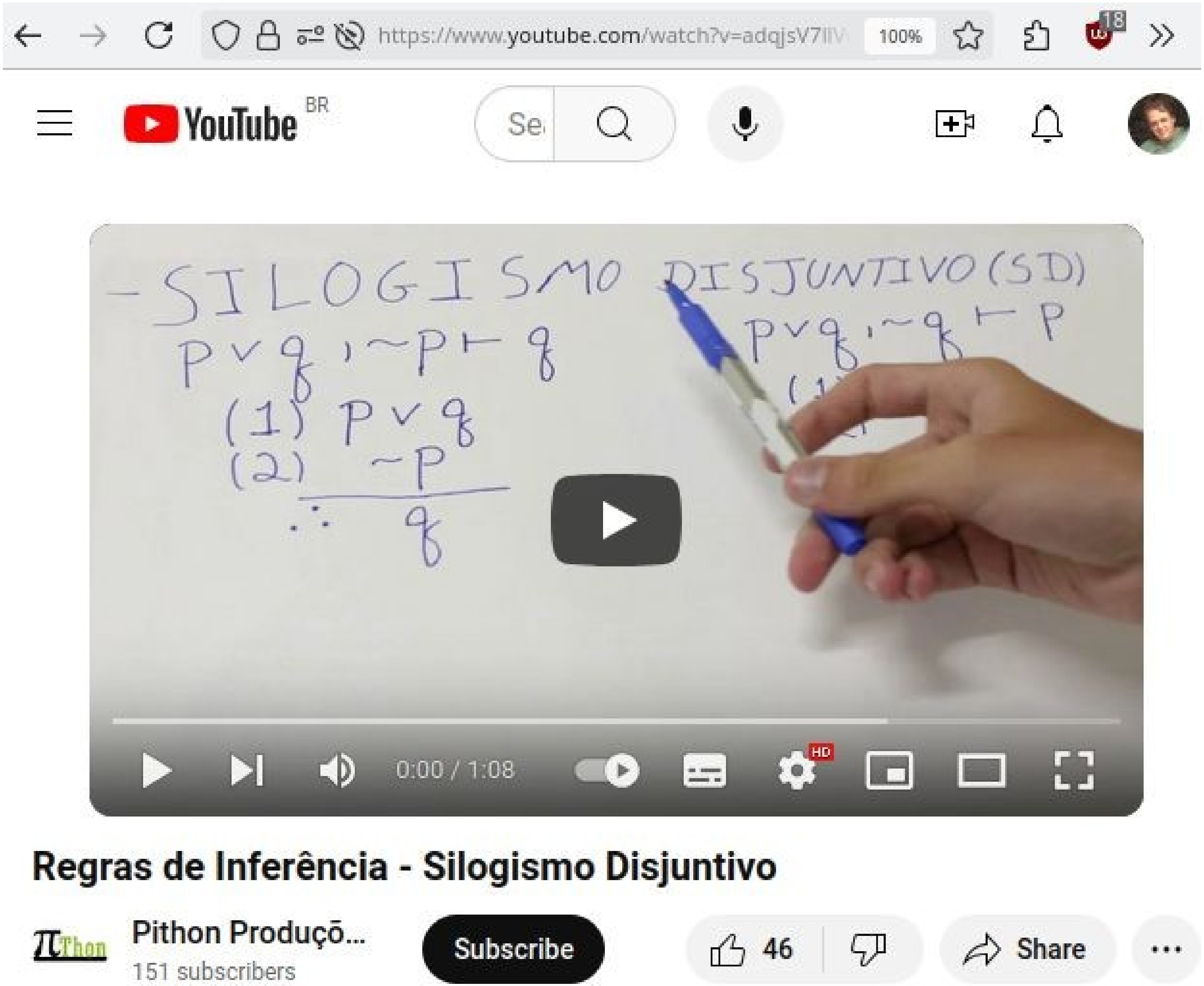
(continuação)

- Ao fazer a conjunção entre a primeira parte com a segunda, conseguimos inferir a conclusão

- Pois, se João desligar o interruptor, a lâmpada se apaga, E a lâmpada não se apagou, logo podemos inferir (concluir) que João não desligou o interruptor

- Resumindo os dois métodos, no Modus Ponens, usamos a implicação para provar que a consequência é verdadeira ao demonstrar que a premissa é verdadeira.

- Já no Modus Tollens, usamos a implicação para provar que a premissa é falsa ao demonstrar que a consequência é falsa


https://www.youtube.com/watch?v=adqjsV7IIV
100%
YouTube BR
- SILOGISMO DISJUNTIVO (SD)
p ∨ q , ~p ⊢ q
(1) p ∨ q
(2) ~p
∴ q
p ∨ q , ~q ⊢ p
0:00 / 1:08
HD
Regras de Inferência - Silogismo Disjuntivo
Pithon Produçõ...
151 subscribers
Subscribe
46
Share

Silogismo Hipotético (SH)

- Além de existirem implicações e conjunções nas hipóteses, a conclusão também é uma implicação

- Sua estrutura é dada pela fbf:
  - $(P \rightarrow Q) \wedge (Q \rightarrow R) \rightarrow (P \rightarrow R)$

$$\frac{\begin{array}{c} P \longrightarrow Q \\ Q \longrightarrow R \end{array}}{P \longrightarrow R}$$

Exemplo

- Se as árvores começam a florir, então começa a primavera. Se começa a primavera, então as árvores dão frutos

- Separando as proposições P, Q, R:
  - P: As árvores começam a florir
  - Q: A primavera começa
  - R: As árvores dão frutos

- Representando cada parte da fórmula de SH:

I. (P → Q): Se as árvores começam a florir, então começa a primavera

II. (Q → R): Se começa a primavera então as árvores dão frutos

- Ao fazer a conjunção entre a primeira parte com a segunda, conseguimos inferir a conclusão

- Pois, se as árvores começam a florir, então começa a primavera, E se começa a primavera então as árvores dão frutos, logo podemos inferir (concluir) que se as árvores começam a florir, então darão frutos

- A consequente de uma proposição é a antecedente na outra e, por isso, o resultado pode ser inferido do antecedente da primeira para a consequente da segunda proposição

Resumo de algumas das principais regras de inferências

Quadro 3.6 | Regras de inferência

De (fbf)	Podemos deduzir (fbf)	Nome/Abreviação
$P \to Q$, P	Q	Modus Ponens/MP
$P \to Q$, $\neg Q$	$\neg P$	Modus Tollens/MT
$P \to Q$, $Q \to R$	$P \to R$	Silogismo Hipotético/SH
P, Q	$P \wedge Q$	Conjunção/conj
$P \wedge Q$	P, Q	Simplicação/simp
P	$P \wedge Q$	Adição/ad

Exemplo

- mostrar que o argumento $(\neg A \vee B) \wedge (B \rightarrow C) \rightarrow (A \rightarrow C)$ é válido

- O primeiro passo é identificar as hipóteses e a conclusão
- A fórmula do argumento $(P1 \wedge P2 \wedge P3 \wedge \ldots \wedge Pn \rightarrow C)$ nos diz que as hipóteses são ligadas pela conjunção e a conclusão é ligada pela última implicação lógica
- Outro detalhe importante é que as hipóteses podem ser fbf e não somente proposições simples
  - Hipótese 1: $(\neg A \vee B)$
  - Hipótese 2: $(B \rightarrow C)$
  - Conclusão: $(A \rightarrow C)$

- Na conclusão temos uma implicação, isso já nos dá indícios de que conseguiremos usar o silogismo hipotético
  - $\neg A \vee B$ (hip)
  - $B \rightarrow C$ (hip)
  - $A \rightarrow B$ (1, cond)
  - $A \rightarrow C$ (3, 4, SH)

- Em quatro passos conseguimos demonstrar que o argumento é válido

- Nos passos 1 e 2 elencamos as hipóteses
- No passo 3, consultamos as regras de equivalência no Quadro 3.5, e usamos a regra de equivalência do condicional, ou seja, trocamos a hipótese $\neg A \vee B$ por $A \rightarrow B$ já que são equivalentes
- Na linha 4, consultamos as regras de inferência no Quadro 3.6 e aplicamos o silogismo hipotético entre as linhas 3 e 2, ou seja, substituímos $(A \rightarrow B) \wedge (B \rightarrow C)$ por $A \rightarrow C$

- Como chegamos exatamente a fbf da conclusão, provamos a validade do argumento
- As regras de dedução lógica devem ser consultadas a todo momento no processo de demonstração
- Não existe uma receita, somente a prática nos auxilia a desenvolvermos o raciocínio