Lógica Proposicional

Prof. Dr. Silvio do Lago Pereira

Departamento de Tecnologia da Informação

Faculdade de Tecnologia de São Paulo



IA estuda como simular comportamento inteligente

comportamento inteligente é resultado de **raciocínio** correto sobre **conhecimento** disponível

conhecimento e raciocínio correto podem ser representados em **lógica**

o formalismo lógico mais simples é a **lógica proposicional**



Lógica proposicional

- É um formalismo composto por:
 - Linguagem formal: usada para representar conhecimento.
 - Métodos de inferência: usados para representar raciocínio.
- Tem como principal finalidade:
 - Representar argumentos, isto é, seqüências de sentenças em que uma delas é uma conclusão e as demais são premissas.
 - Validar argumentos, isto é, verificar se sua conclusão é uma consequência lógica de suas premissas.

Exercício 1. Intuitivamente, qual dos dois argumentos a seguir é válido?

- Se neva, então faz frio. Está nevando. Logo, está fazendo frio.
- Se chove, então a rua fica molhada. A rua está molhada. Logo, choveu.



Proposição

é uma sentença declarativa que pode ser verdadeira ou falsa, mas não as duas coisas ao mesmo tempo.

Exercício 2

- Quais das sentenças a seguir são proposições?
 - Abra a porta.
 - Excelente apresentação!
 - Esta semana tem oito dias.
 - Em que continente fica o Brasil?
 - A Lua é um satélite da Terra.
- Por que a sentença "esta frase é falsa" não é uma proposição?



Conectivo

são partículas (**não**, **e**, **ou**, **então**) que permitem construir sentenças complexas a partir de outras mais simples.

Exemplo:

- A partir das sentenças (proposições atômicas):
 - Está chovendo
 - A rua está molhada
- Podemos construir as sentenças (proposições compostas):
 - Não está chovendo
 - Se está chovendo, então a rua está molhada



Sintaxe: define a estrutura das sentenças

Símbolos

- Proposições: a, b, c, ...
- Conectivos: \neg , \wedge , \vee , \rightarrow (da maior para a menor precedência)

Fórmulas

- Todas as proposições são fórmulas.
- Se α e β são fórmulas, então também são fórmulas:
 - ¬α (negação)
 - $\alpha \land \beta$ (conjunção)
 - $\alpha \vee \beta$ (disjunção)
 - $\alpha \rightarrow \beta$ (implicação)



Semântica: define o significado das sentenças

- Interpretação: associação entre proposições e valores-verdade (V ou F)
 - Uma fórmula contendo n proposições admite 2ⁿ interpretações distintas.
- Tabela-verdade: avalia uma fórmula em cada interpretação possível.

р	q	¬р	p ^ q	p v q	$p \rightarrow q$
F	F	V	F	F	V
F	V	V	F	V	V
V	F	F	F	V	F
V	V	F	V	V	V

Tipos de fórmulas:

- Válida (tautológica): é verdadeira em toda interpretação.
- Satisfatível (contingente): é verdadeira em alguma interpretação.
- Insatisfatível (contraditória): é verdadeira em nenhuma interpretação.



Representação de conhecimento

Conhecimento pode ser representado de duas formas:

- explícita: por meio da formalização de sentenças
- implícita: por meio de consequência lógica (fatos derivados das sentenças)

Passos para formalização de sentenças

- Identificamos as palavras da sentença que correspondem a conectivos.
- Identificamos as partes da sentença que correspondem a proposições atômicas e associamos a cada uma delas um símbolo proposicional.
- Escrevemos a fórmula correspondente à sentença, substituindo suas proposições atômicas pelos respectivos símbolos proposicionais e seus conectivos lógicos pelos respectivos símbolos conectivos



Representação de conhecimento

Exemplo

- Está chovendo.
- Se está chovendo, então a rua está molhada.
- Se a rua está molhada, então a rua está escorregadia.
- Vocabulário
 - c: "está chovendo"
 - m : "a rua está molhada"
 - e : "a rua está escorregadia"
- Formalização
 - $\Delta = \{c, c \rightarrow m, m \rightarrow e\}$

base de conhecimento

Formalização de argumentos

Um argumento é uma seqüência de premissas seguida de uma conclusão

Exemplo

- Se neva, então faz frio.
- Está nevando.
- Logo, está fazendo frio.
- Vocabulário
 - **n**: "neve"
 - **f**: "frio"
- Formalização

• $\{n \rightarrow f, n\} \neq f$

consequência lógica



Exercício 3

Usando a sintaxe da lógica proposicional, formalize o argumento:

Se o time joga bem, então ganha o campeonato.

Se o time não joga bem, então o técnico é culpado.

Se o time ganha o campeonato, então os torcedores ficam contentes.

Os torcedores não estão contentes.

Logo, o técnico é culpado.



Nem todo argumento é válido!

Exemplo: Intuitivamente, qual dos argumentos a seguir é válido?

Argumento 1

- Se eu fosse artista, então eu seria famoso.
- Não sou famoso.
- Logo, não sou artista.

Argumento 2

- Se eu fosse artista, então eu seria famoso.
- Sou famoso.
- Logo, sou artista.

Validação de argumentos

Um argumento é válido se a sua conclusão é uma conseqüência lógica de suas premissas, ou seja, a veracidade da conclusão está implícita na veracidade das premissas.

- Vamos mostrar três métodos de validação de argumentos:
 - Tabela-verdade (semântico)
 - Prova por dedução (sintático)
 - Prova por refutação (sintático)
- Métodos semânticos são baseados em interpretações
- Métodos sintáticos são baseados em regras de inferência (raciocínio)



Um argumento da forma $\{\alpha_1, \ldots, \alpha_n\} \in \beta$ é válido se e somente se a fórmula correspondente $\alpha_1 \wedge ... \wedge \alpha_n \rightarrow \beta$ é válida (tautológica).

Exemplo

Argumento 1

- Se eu fosse artista, seria famoso.
- Não sou famoso.
- Logo, não sou artista.

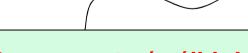
Vocabulário

- **a** : "artista"
- **f**: "famoso"

Formalização

•
$$\{a \rightarrow f, \neg f\} \models \neg a$$

a	f	(a	\rightarrow	f)	٨	Г	f	\rightarrow	Г	a
F	F	F	V	F	V	V	F	V	V	F
F	٧	F	V	V	F	F	V	٧	V	F
V	F	٧	F	F	F	V	F	٧	F	V
V	٧	٧	>	>	F	F	>	>	ш	V



O argumento é válido!



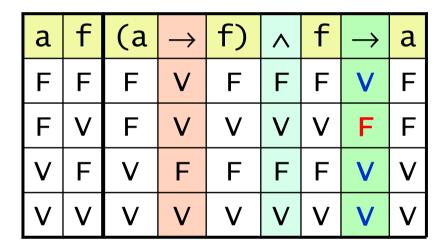
Exemplo

- **Argumento 2**
 - Se eu fosse artista, seria famoso.
 - Sou famoso.
 - Logo, sou artista.
- Vocabulário

• a: "artista"

• **f**: "famoso"

- Formalização
 - $\{a \rightarrow f, f\} \neq a$





O argumento NÃO é válido!



Exercício 4

Use tabela-verdade para verificar a validade dos argumentos a seguir:

- 1. Se neva, então faz frio. Não está nevando. Logo, não está frio.
- 2. Se eu durmo tarde, não acordo cedo. Acordo cedo. Logo, não durmo tarde.
- 3. Gosto de dançar ou cantar. Não gosto de dançar. Logo, gosto de cantar.

Exercício 5

Use tabela-verdade para verificar a validade do argumento a seguir:

Se o time joga bem, então ganha o campeonato.

Se o time não joga bem, então o técnico é culpado.

Se o time ganha o campeonato, então os torcedores ficam contentes.

Os torcedores não estão contentes.

Logo, o técnico é culpado.

Formalização: $\{j\rightarrow q, \neg j\rightarrow t, q\rightarrow c, \neg c\} \models t$



Exercício 6

Sócrates está disposto a visitar Platão ou não?

Se Platão está disposto a visitar Sócrates, então Sócrates está disposto a visitar Platão. Por outro lado, se Sócrates está disposto a visitar Platão, então Platão não está disposto a visitar Sócrates; mas se Sócrates não está disposto a visitar Platão, então Platão está disposto a visitar Sócrates.

Vocabulário:

p : "Platão está disposto a visitar Sócrates"

s : "Sócrates está disposto a visitar Platão"

Formalização: $\{p \rightarrow s, (s \rightarrow \neg p) \land (\neg s \rightarrow p)\}$



- Consequência lógica é o elo entre o que um agente "acredita" e aquilo que é explicitamente representado em sua base de conhecimento.
- A tabela-verdade é um método semântico que permite verificar consegüências lógicas.
- Este método tem a vantagem de ser conceitualmente simples; mas, como o número de linhas na tabela-verdade cresce exponencialmente em função do número de proposições na fórmula, seu uso nem sempre é viável.
- Assim, apresentaremos o raciocínio automatizado como uma alternativa mais eficiente para verificação de consequência lógica (isto é, validação de argumentos).

Uma **prova por dedução** de uma fórmula φ , a partir de uma base de conhecimento Δ , é uma seqüência finita de fórmulas $\gamma_1, \ldots, \gamma_k$ tal que:

- $\gamma_k = \varphi$;
- para $1 \le i \le k$, ou $\gamma_i \in \Delta$ ou, então, γ_i é **derivada** de fórmulas em $\Delta \cup \{\gamma_1, \ldots, \gamma_{i-1}\}$, pela aplicação de uma **regra de inferência**.

Regra de inferência:

é um padrão de manipulação sintática que define como uma fórmula (conclusão) pode ser derivada de outras fórmulas (premissas)

Regras de inferência clássicas:

- Modus ponens (MP): $\{\alpha \rightarrow \beta, \alpha\} \vdash \beta$
- Modus tollens (MT): $\{\alpha \rightarrow \beta, \neg \beta\} \vdash \neg \alpha$
- Silogismo hipotético (SH): $\{\alpha \rightarrow \beta, \beta \rightarrow \gamma\} \vdash \alpha \rightarrow \gamma$

As regras de inferência clássicas:

- representam "esquemas de raciocínio" válidos
- podemos validar estes esquemas usando tabela-verdade
- podem ser usadas para derivar conclusões que são conseqüências lógicas de suas premissas

Exemplo: validar o argumento $\{j\rightarrow g, \neg j\rightarrow t, g\rightarrow c, \neg c\} \models t$

(1)
$$j \rightarrow g \quad \Delta$$

(2)
$$\neg j \rightarrow t \Delta$$

(3)
$$g \rightarrow c$$
 Δ

(4)
$$\neg c$$
 Δ

(5)
$$j\rightarrow c$$
 SH(1,3)

(6)
$$\neg j$$
 MT(5,4)

MP:
$$\{\alpha \rightarrow \beta, \alpha\} \vdash \beta$$

MT:
$$\{\alpha \rightarrow \beta, \neg \beta\} \vdash \neg \alpha$$

SH:
$$\{\alpha \rightarrow \beta, \beta \rightarrow \gamma\} \vdash \alpha \rightarrow \gamma$$

Conclusão: o argumento é válido, pois a fórmula t pode ser derivada de Δ .

Exercício 7

Use tabela-verdade para validar as regras de inferência clássicas:

• MP:
$$\{\alpha \rightarrow \beta, \alpha\} \vdash \beta$$

• MT:
$$\{\alpha \rightarrow \beta, \neg \beta\} \vdash \neg \alpha$$

• SH:
$$\{\alpha \rightarrow \beta, \beta \rightarrow \gamma\} \vdash \alpha \rightarrow \gamma$$

Prove usando as regras de inferências clássicas:

•
$$\{p \rightarrow q, \neg q, \neg p \rightarrow r\} \vdash r$$

•
$$\{\neg p \rightarrow \neg q, q, p \rightarrow \neg r\} \vdash \neg r$$

•
$$\{p \rightarrow q, q \rightarrow r, \neg r, \neg p \rightarrow s\} \vdash s$$

Prova por refutação

Embora a prova por dedução seja um método mais prático que a tabelaverdade, ainda é muito difícil obter algoritmos eficientes para validação de argumentos com base neste método.

Refutação

- Refutação é um processo em que se demonstra que uma determinada hipótese contradiz uma base de conhecimento.
- Uma base de conhecimento $\Delta = \{\alpha_1, ..., \alpha_n\}$ é consistente se a fórmula correspondente $\alpha_1 \wedge ... \wedge \alpha_n$ é satisfatível.
- Se $\Delta = \{\alpha_1, \ldots, \alpha_n\}$ é consistente, provar $\Delta \models \gamma$ equivale a mostrar que o conjunto de fórmulas $\{\alpha_1, \ldots, \alpha_n, \neg \gamma\}$ é inconsistente.



Argumento

- (1) Se o time joga bem, então ganha o campeonato.
- (2) Se o time não joga bem, então o técnico é culpado.
- (3) Se o time ganha o campeonato, então os torcedores ficam contentes.
- (4) Os torcedores não estão contentes.
- (5) Logo, o técnico é culpado.

Refutação

(a) O técnico não é culpado	hipótese
(b) O time joga bem	MT(a,2)
(c) O time ganha o campeonato	MP(b,1)
(d) Os torcedores ficam contentes	MP(c,3)
(e) Contradição!	Confrontando (d) e (4)

Conclusão: a hipótese contradiz as premissas, logo o argumento é válido!

Prova por refutação

Exemplo: validar o argumento $\{j\rightarrow g, \neg j\rightarrow t, g\rightarrow c, \neg c\} \models t$

- $(1) j \rightarrow g \qquad \Delta$
- (2) $\neg j \rightarrow t \Delta$
- (3) $g \rightarrow c$ Δ
- (4) $\neg c$ Δ

- (5) ¬t Hipótese
- (6) j MT(5,2)
- (7) g MP(6,1)
- (8) c MP(7,3)
- (9) ☐ Contradição!

Conclusão: como $\Delta \cup \{\neg t\}$ é inconsistente, segue que $\Delta \models t$.

Prova por refutação

Exercício 8

Usando refutação, mostre que o argumento é válido.

- (1) Se Ana sente dor de estômago ela fica irritada.
- (2) Se Ana toma remédio para dor de cabeça ela fica com dor de estômago.
- (3) Ana não está irritada.
- (4) Logo, Ana não tomou remédio para dor de cabeça.

Prove usando refutação:

$$\{p\rightarrow q, \neg q, \neg p\rightarrow r\} \vdash r$$

 $\{\neg p\rightarrow \neg q, q, p\rightarrow \neg r\} \vdash \neg r$
 $\{p\rightarrow q, q\rightarrow r, \neg r, \neg p\rightarrow s\} \vdash s$

Forma normal conjuntiva

Para simplificar a automatização do processo de refutação, vamos usar fórmulas normais (Forma Normal Conjuntiva - FNC).

Passos para conversão para FNC

Elimine a implicação:

$$\alpha \rightarrow \beta \equiv \neg \alpha \lor \beta$$

Reduza o escopo da negação:

$$\neg(\alpha \land \beta) \equiv \neg\alpha \lor \neg\beta$$
$$\neg(\alpha \lor \beta) \equiv \neg\alpha \land \neg\beta$$

Reduza o escopo da disjunção:

$$\alpha \vee (\beta \wedge \gamma) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$$



Forma normal conjuntiva

Exemplo de conversão para FNC

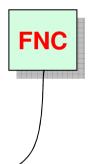
$$p \lor q \rightarrow r \land s$$

$$\equiv \neg (p \lor q) \lor (r \land s)$$

$$\equiv (\neg p \land \neg q) \lor (r \land s)$$

$$\equiv ((\neg p \land \neg q) \lor r) \land ((\neg p \land \neg q) \lor s)$$

$$\equiv (\neg p \lor r) \land (\neg q \lor r) \land (\neg p \lor s) \land (\neg q \lor s)$$



Fórmulas normais: $\{\neg p \lor r, \neg q \lor r, \neg p \lor s, \neg q \lor s\}$



Inferência por resolução

- FNC permite usar inferência por resolução
- A idéia da resolução é:
 - RES($\alpha \vee \beta$, $\neg \beta \vee \gamma$) = $\alpha \vee \gamma$
 - RES(α , $\neg \alpha$) = \square

Equivalência entre resolução e regras de inferência clássicas

$MP(\alpha \!\to\! \beta, \alpha) = \beta$	$RES(\neg \alpha \vee \beta, \alpha) = \beta$
$MT(\alpha \!\to\! \beta, \neg \beta) = \neg \alpha$	$RES(\neg \alpha \vee \beta, \neg \beta) = \neg \alpha$
$SH(\alpha \!\to\! \beta,\beta \!\to\! \gamma) = \alpha \!\to\! \gamma$	$RES(\neg \alpha \lor \beta, \neg \beta \lor \gamma) = \neg \alpha \lor \gamma$



Inferência por resolução

Exemplo: validar o argumento $\{j\rightarrow g, \neg j\rightarrow t, g\rightarrow c, \neg c\} \models t$

(1)
$$\neg j \lor g \quad \Delta$$

(2)
$$j \vee t$$
 Δ

(3)
$$\neg g \lor c$$
 Δ

(4)
$$\neg c$$
 Δ

(6) j
$$RES(5,2)$$

(7) g
$$RES(6,1)$$

(8) c
$$RES(7,3)$$

$$(9) \square RES(8,4)$$

Este é o mecanismo de raciocínio implementado pelo Prolog!

Conclusão: como $\Delta \cup \{\neg t\}$ é inconsistente, segue que $\Delta \models t$.



Exercício 9

Prove o argumento a seguir, usando refutação e inferência por resolução.

Se o programa possui erros de sintaxe, sua compilação produz mensagem de erro. Se o programa não possui erros de sintaxe, sua compilação produz um executável. Se tivermos um programa executável, podemos executá-lo para obter um resultado. Não temos como executar o programa para obter um resultado. Logo, a compilação do programa produz uma mensagem de erro.

Fim