Agentes Lógicos

Inteligência Artificial
PCS3438

Escola Politécnica da USP
Engenharia de Computação (PCS)

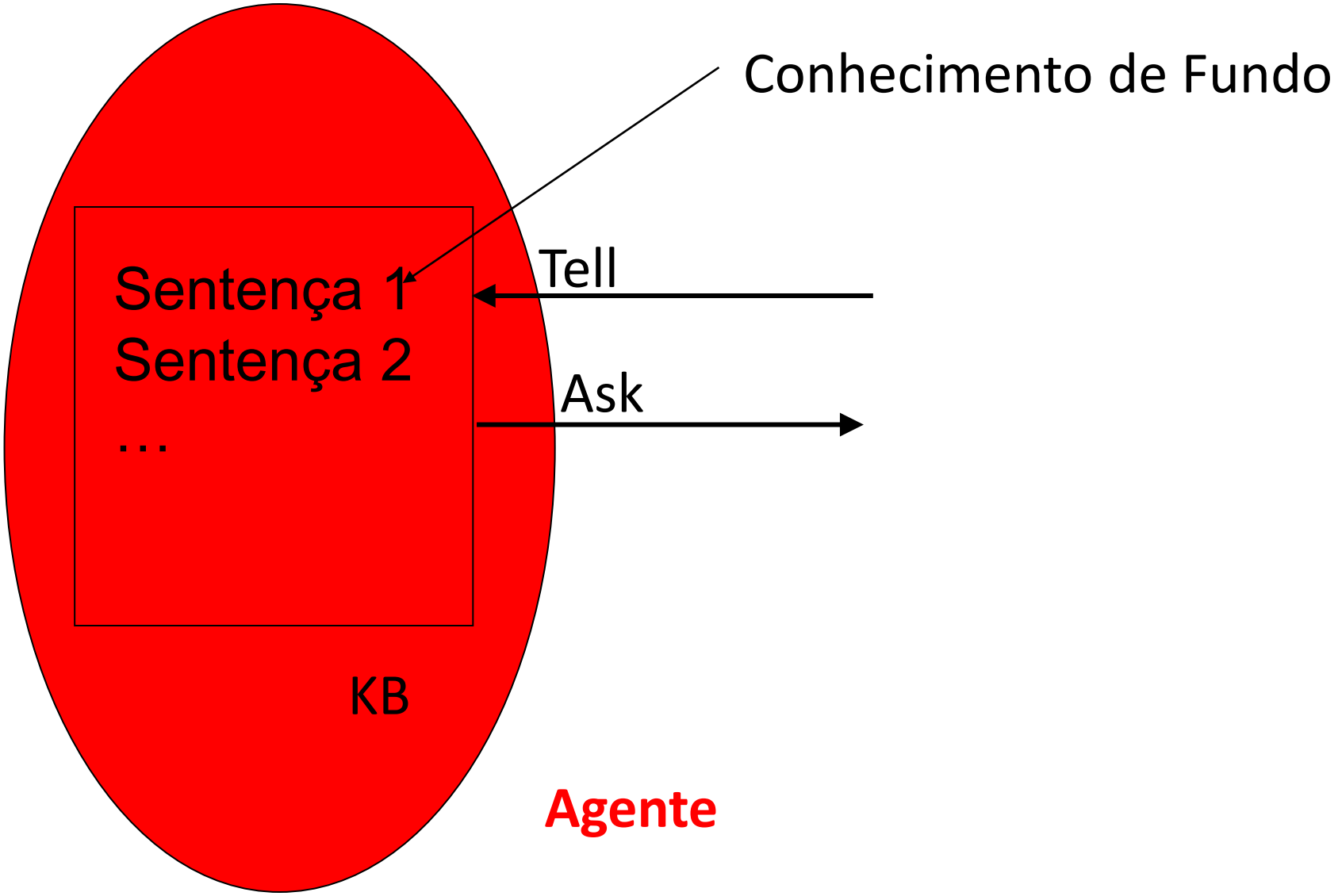
Lembrando...

- Agentes inteligentes
  - Agentes que resolvem problemas
    - Ambiente observável, discreto, determinístico
    - Técnicas de busca da solução do problema
      - Busca não informada (cega)
      - Busca informada
      - Busca local
      - Busca competitiva
      - Satisfação de restrições
  - Mas e se o ambiente for parcialmente observável?

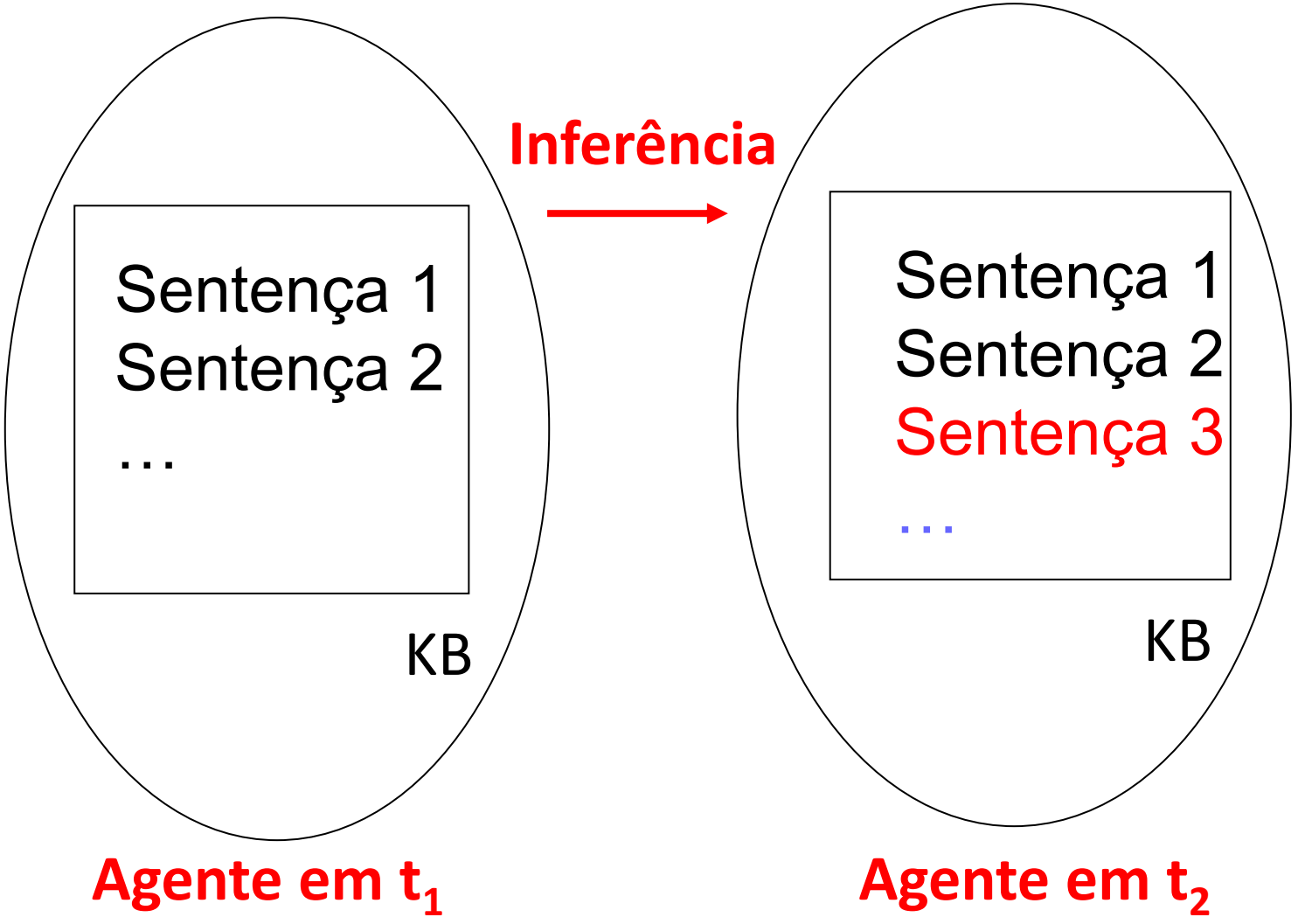
Agentes Baseados em Conhecimento

- Utilização é importante em ambientes que sejam parcialmente observáveis, pois o agente pode combinar o resultado de percepções correntes com um conhecimento anterior.
- Um agente baseado em conhecimento é flexível, pois seu conhecimento pode ser alterado/adicionado.

Agentes Baseados em Conhecimento



Agentes Baseados em Conhecimento



Agente Baseado em Conhecimento

função Agente-KB(*percepção*) **retorna** uma *ação*

estático: base de conhecimento KB, contador t =0

Tell(*KB*, Criar-sentença-de-percepção(*percepção,t*))

ação ← Ask(*KB,* Pergunte-Ação(*t*))

Tell(*KB*, Criar-sentença-de-ação(*ação,t*))

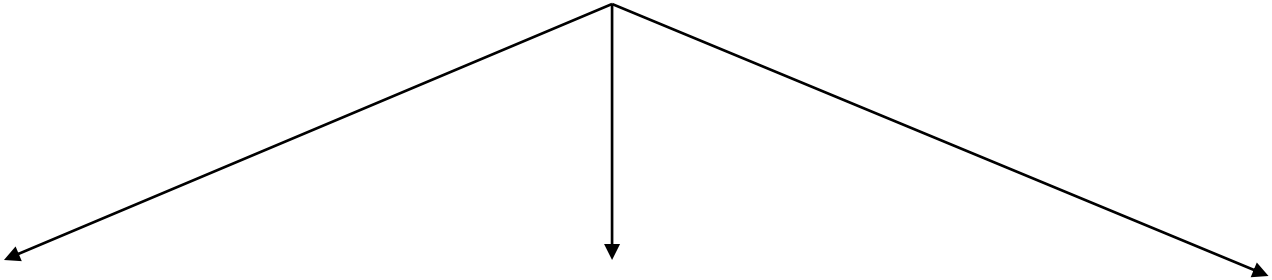
t ← *t* + 1

retorna *ação*

– ***Abordagem declarativa***

Lógica Matemática

Formalização de uma linguagem lógica



Sintaxe

Especificação das expressões legais da linguagem

Semântica

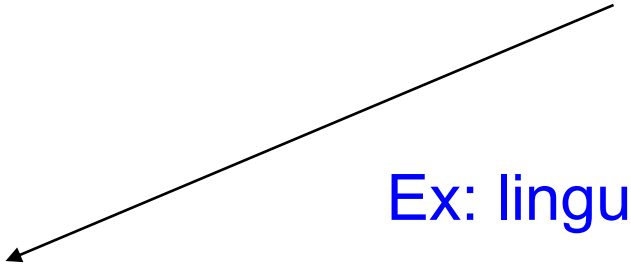
Significado das expressões da linguagem

Inferência

Regras que manipulam expressões da linguagem e que preservam alguns aspectos semânticos

Lógica Matemática: Sintaxe

Formalização de uma linguagem lógica



Sintaxe

Ex: linguagem aritmética

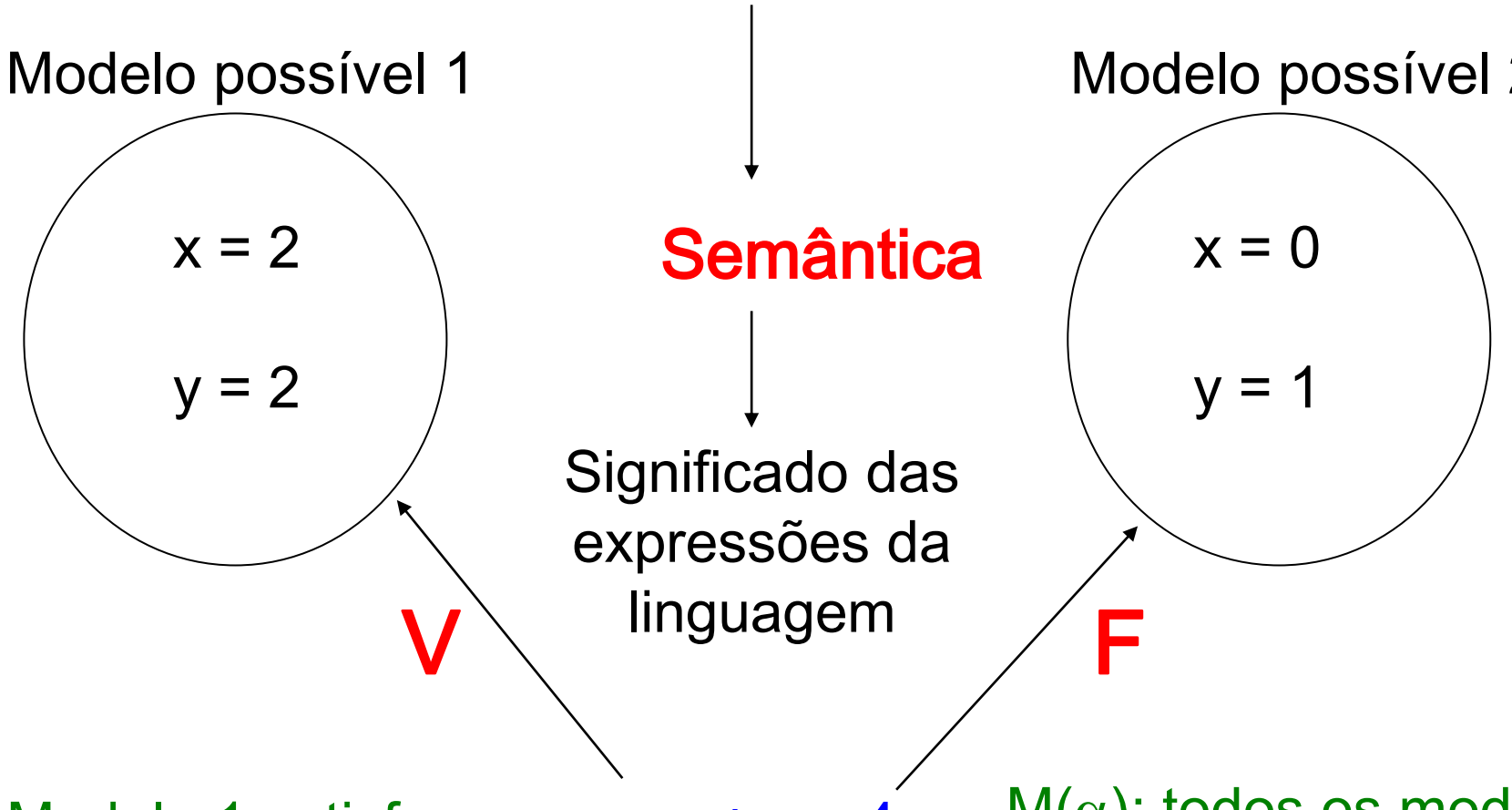
α: x + y = 4 é uma fórmula bem formada (fbf)

β: x 4 y + = não é uma fbf

Especificação
das expressões
legais da linguagem

Lógica Matemática: Semântica

Formalização de uma linguagem lógica



Lógica Matemática

Formalização de uma linguagem lógica

Raciocínio lógico envolve a relação de consequência lógica:

$$\alpha \models \beta$$

Indica que a sentença α tem como consequência lógica a sentença β

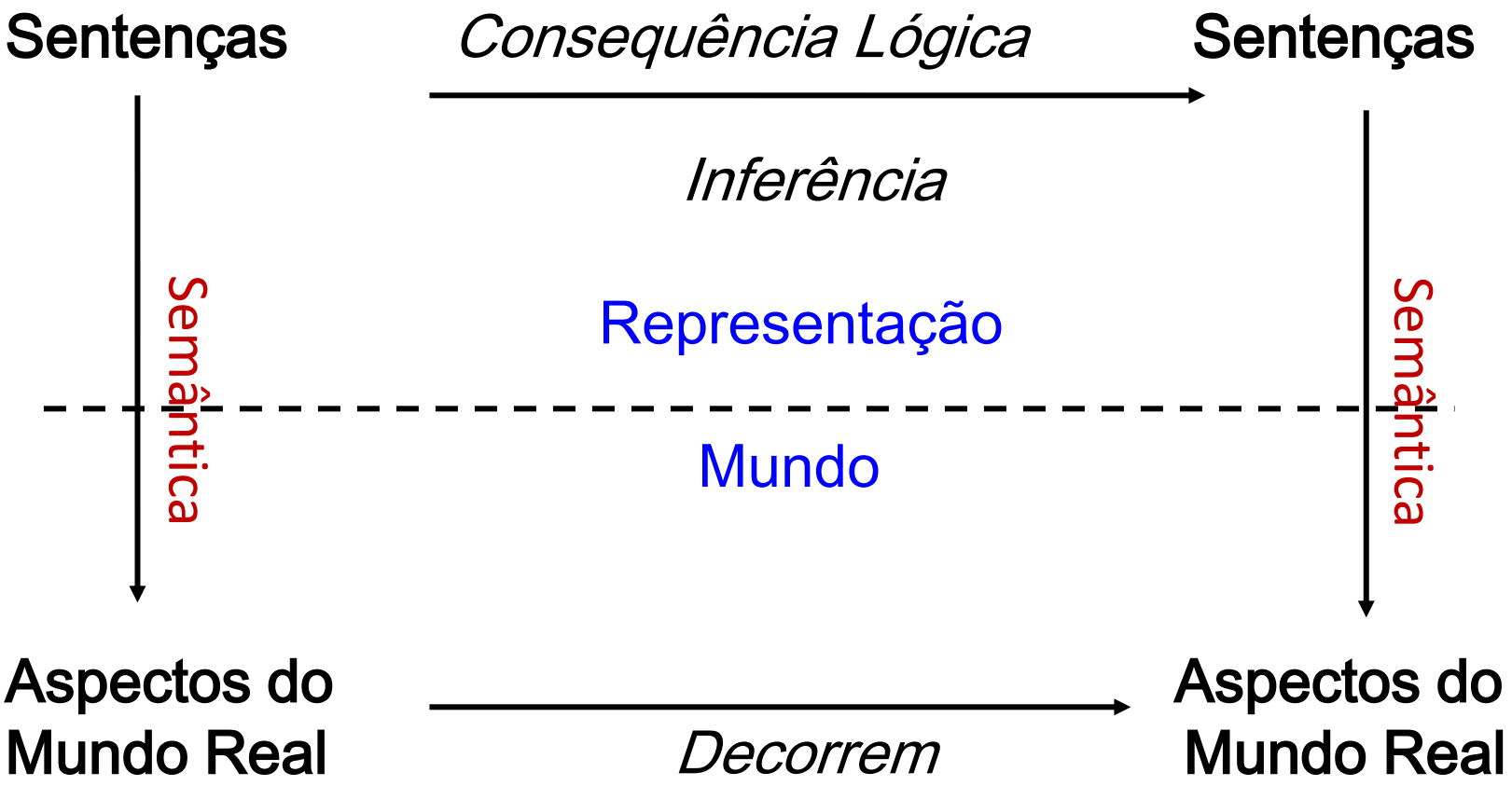
Inferência

Regras que manipulam expressões da linguagem e que preservam alguns aspectos semânticos

$$\alpha \models \beta \quad \text{sse} \quad M(\alpha) \subseteq M(\beta)$$

Em todo modelo em que α é V, β também é V.

Lógica Matemática



LÓGICA PROPOSICIONAL

Sintaxe: Proposições

Trata-se do cálculo mais simples que existe.

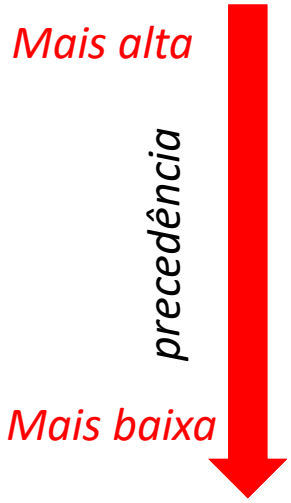
Uma **proposição** é um enunciado declarativo.

- **p**: Todo imigrante italiano é palmeirense
- **q**: Todo palmeirense gosta de comer feijoada

Proposições podem ser **verdadeiras** ou **falsas**.

Sintaxe: Conectivos Lógicos

Sentenças complexas podem ser construídas a partir de proposições simples, parênteses e de **conectivos lógicos**:



- $\neg$: negação (não)
- $\wedge$: conjunção (e)
- $\vee$: disjunção (ou)
- $\rightarrow$: implicação (condicional, se...então)
- $\leftrightarrow$: bicondicional (se e somente se)

Sintaxe do Cálculo Proposicional em BNF

Símbolo inicial: <sentença>

<sentença>::= <sentença atômica> |
<sentença complexa>

<sentença atômica>::= V | F | <símbolo>

<símbolo>::= p | q | r | s | ...

<sentença complexa>::=

$\neg$ <sentença> | (<sentença>) |

<sentença> $\wedge$ <sentença> |

<sentença> $\vee$ <sentença> |

<sentença> $\rightarrow$ <sentença> |

<sentença> $\leftrightarrow$ <sentença>

Semântica: Princípios Fundamentais

Fórmulas bem formadas têm um valor verdade:
(V ou F)

1. **Princípio da Identidade**: Todo objeto é idêntico a si mesmo.
2. **Princípio da não contradição**: Uma proposição não pode ser verdadeira e falsa ao mesmo tempo.
3. **Princípio do terceiro excluído**: Uma proposição é falsa ou verdadeira, não havendo um terceiro caso.

Semântica: Conectivos Lógicos

Dadas 2 fbfs α e β e um modelo M:

1. $\neg \alpha$ é V sse α for F em M.
2. $\alpha \wedge \beta$ é V sse α e β forem V em M.
3. $\alpha \vee \beta$ é F sse α e β forem F em M.
4. $\alpha \rightarrow \beta$ é V sse α for F ou β for V em M.
5. $\alpha \leftrightarrow \beta$ é V sse α e β tiverem o mesmo valor-verdade em M.

Semântica: Tabela Verdade

Caso uma fbf contenha n sentenças atômicas, existem 2^n interpretações distintas, que podem ser colocadas em uma **tabela da verdade.**

Conectivo NÃO (Negação)	
p	**¬p**
F	V
V	F

Semântica: Tabela Verdade

Conectivo **OU** Disjunção			Conectivo **E** Conjunção		
p	**q**	**p ∨ q**	**p**	**q**	**p ∧ q**
F	**F**	**F**	**F**	**F**	**F**
F	**V**	**V**	**F**	**V**	**F**
V	**F**	**V**	**V**	**F**	**F**
V	**V**	**V**	**V**	**V**	**V**

Semântica: Tabela Verdade

Conectivo Condicional			Conectivo Bicondicional		
p	**q**	**$p \rightarrow q$**	**p**	**q**	**$p \leftrightarrow q$**
F	**F**	**V**	**F**	**F**	**V**
F	**V**	**V**	**F**	**V**	**F**
V	**F**	**F**	**V**	**F**	**F**
V	**V**	**V**	**V**	**V**	**V**

Consequência lógica

- Dadas as fbfs $\alpha 1, \alpha 2, \ldots, \alpha n$ e β, diz-se que α é consequência lógica de $\alpha 1, \alpha 2, \ldots, \alpha n$ se e somente se para quaisquer modelos onde $\alpha 1, \alpha 2, \ldots, \alpha n$ forem simultaneamente verdadeiras, α também é verdadeira

$$\alpha 1, \alpha 2, \ldots, \alpha n \models \beta$$

Exemplo

α1: p∧q
α2: p
α3: q

β: p∨q

p	q	p∧q	p∨q
F	F	F	F
F	V	F	V
V	F	F	V
V	V	V	V

α1, α2, α3 ⊨ β

$$\alpha \models \beta \quad \text{sse} \quad M(\alpha) \subseteq M(\beta)$$

Semântica: Fórmulas Bem Formadas

Uma fórmula bem formada α é :

- **Válida (tautologia)** se assumir valor **V** em todo modelo;
- **Satisfazível** se assumir valor V em algum modelo;
- **Inválida** se assumir valor **F** em algum modelo;
- **Insatisfazível (contradição)** se assumir valor **F** em todo modelo;
- **Contingente** se não for válida nem insatisfazível.

Consequência Lógica

Teorema da Dedução

Dadas as fbfs $\beta_1, \beta_2, \ldots, \beta_n$ e α,
α é consequência lógica de $\beta_1, \beta_2, \ldots, \beta_n$
sse a fbf $\beta_1 \wedge \beta_2 \wedge \ldots \wedge \beta_n \rightarrow \alpha$
for uma **tautologia**.

Teorema da Contradição

Dadas as fbfs $\beta_1, \beta_2, \ldots, \beta_n$ e α,
α é consequência lógica de $\beta_1, \beta_2, \ldots, \beta_n$
sse a fbf $\beta_1 \wedge \beta_2 \wedge \ldots \wedge \beta_n \wedge \neg \alpha$
for uma **contradição**.

Exemplo de Dedução

- Verificar se: $(p \Box h) \Box \Box h \models p$
- Por dedução, basta mostrar que é válida a fbf:

$$(p \vee h) \wedge \Box h \rightarrow p$$

p	h	$p \Box h$	$(p \Box h) \Box \Box h$	$(p \vee h) \wedge \Box h \rightarrow p$
F	F	F	F	V
F	V	V	F	V
V	F	V	V	V
V	V	V	F	V

TAUTOLOGIA!

Exemplo de Contradição

- Verificar se: $(p \square h) \square \square h \models p$
- Por contradição, basta mostrar que é contradição a fbf: $(p \square h) \square \square h \square \square p$

p	h	$p \square h$	$(p \square h) \square \square h$	$\square p$	$(p \square h) \square \square h \square \square p$
F	F	F	F	V	F
F	V	V	F	V	F
V	F	V	V	F	F
V	V	V	F	F	F

CONTRADIÇÃO!

Consequência Lógica

Entretanto, procedimentos usando tabela verdade podem ser muito custosos!

Felizmente, existem propriedades s**intéticas** que podem auxiliar a verificação ou geração de uma consequência lógica: **regras de inferência.**

Regras de Inferência

Modus Ponens	$\alpha, \alpha \rightarrow \beta \models \beta$
Modus Tollens	$\alpha \rightarrow \beta, \neg\beta \models \neg\alpha$
Silogismo Hipotético	$\alpha \rightarrow \beta, \beta \rightarrow \gamma \models \alpha \rightarrow \gamma$
Silogismo Disjuntivo	$\alpha \vee \beta, \neg \alpha \models \beta$
Simplificação	$\alpha \wedge \beta \models \alpha$
Conjunção	$\alpha, \beta \models \alpha \wedge \beta$
Adição	$\alpha \models \alpha \vee \beta$
Contraposição	$\alpha \rightarrow \beta \models \neg\beta \rightarrow \neg\alpha$
Resolução	$\alpha \vee \beta, \neg \alpha \vee \gamma \models \beta \vee \gamma$

No caso do Modus Ponens, lê-se: Se α for verdade e também $\alpha \rightarrow \beta$ for verdade, então obrigatoriamente β será verdade.

Outras propriedades

$\alpha \wedge \beta \equiv \beta \wedge \alpha$ "*Comutativa*"

$\alpha \vee \beta \equiv \beta \vee \alpha$ "*Comutativa*"

$(\alpha \wedge \beta) \wedge \delta \equiv \alpha \wedge (\beta \wedge \delta)$ "*Associativa*"

$(\alpha \vee \beta) \vee \delta \equiv \alpha \vee (\beta \vee \delta)$ "*Associativa*"

$\neg(\neg \alpha) \equiv \alpha$ "*Exclusão de dupla negação*"

$\alpha \rightarrow \beta \equiv \neg \alpha \vee \beta$ "*Exclusão da implicação*"

$\alpha \leftrightarrow \beta \equiv (\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$ "*Exclusão da bicondicional*"

$\neg(\alpha \wedge \beta) \equiv \neg \alpha \vee \neg \beta$ "*DeMorgan*"

$\neg(\alpha \vee \beta) \equiv \neg \alpha \wedge \neg \beta$ "*DeMorgan*"

$\alpha \wedge (\beta \vee \delta) \equiv (\alpha \wedge \beta) \vee (\alpha \wedge \delta)$ "*Distributiva*"

$\alpha \vee (\beta \wedge \delta) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \delta)$ "*Distributiva*"

Provas de Teoremas

Equivalência lógica:

$\alpha \square \beta$ sse $\alpha \models \beta$ e $\beta \models \alpha$

Demonstrações por Dedução

Prova Direta: Parte-se das premissas e chega-se à conclusão. Usa-se o Teorema da Dedução. É chamada também de **dedução lógica**.

Prova por Contradição: Utilizando-se o Teorema da Contradição, nega-se a conclusão e prova-se que a conjunção das premissas com esta conclusão negada é insatisfatível. É chamada também de **refutação**.

Um caso especial é a chamada **prova por contra-posição:** prova-se que a negação da conclusão implica logicamente a negação das premissas

Demonstrações por Dedução

Exemplo: Verificar se o argumento lógico é válido por dedução:

- Se as uvas caem, então a raposa as come
- Se a raposa as come, então estão maduras
- As uvas estão verdes ou caem

Logo

- A raposa come as uvas se e só se as uvas caem

Demonstrações por Dedução em prova direta

Nomeando:
p: as uvas caem
q: a raposa come as uvas
r: as uvas estão maduras

C1: $p \rightarrow q$	1a. premissa
C2: $q \rightarrow r$	2a. premissa
C3: $\neg r \vee p$	3a. premissa
C4: $r \rightarrow p$	(Subst. Equiv. C3)
C5: $q \rightarrow p$	(Silog. Hip. C2 e C4)
C6: $(p \rightarrow q) \wedge (q \rightarrow p)$	(Conjunção C1 e C5)
C7: $p \leftrightarrow q$	(Subst. Equiv. C6)

Demonstrações por Contradição

Nomeando:

p: as uvas caem

q: a raposa come as uvas

r: as uvas estão maduras

C1: $p \rightarrow q$	1a. premissa
C2: $q \rightarrow r$	2a. premissa
C3: $\neg r \vee p$	3a. Premissa
C4: $\neg(p \leftrightarrow q\ )$	Negação da conclusão

Demonstrações utilizando Resolução

A regra de inferência da resolução é muito utilizada na prática, pois além de ser uma regra de inferência **correta** ela também é **completa** para a demonstração por refutação.

Para poder aplicar a regra da resolução, o conjunto de fbfs deve ser transformado numa conjunção de **cláusulas.**

Uma **cláusula** é uma disjunção de literais (átomos ou átomos negados).

Cláusulas

Prova-se que qualquer fbf do cálculo proposicional pode ser transformada num conjunto de cláusulas equivalente, através dos seguintes passos:

1. Substituir $\alpha \leftrightarrow \beta$ por $(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$
2. Substituir $\alpha \rightarrow \beta$ por $\neg\alpha \vee \beta$
3. Colar as negações nos átomos, utilizando as equivalências $\neg(\neg\alpha) \equiv \alpha$, $\neg(\alpha \wedge \beta) \equiv \neg\alpha \vee \neg\beta$ e $\neg(\alpha \vee \beta) \equiv \neg\alpha \wedge \neg\beta$
4. Distribuir as disjunções pelas conjunções, utilizando $\alpha \vee (\beta \wedge \gamma) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$

Demonstrações utilizando Resolução

Uma demonstração **por resolução** consiste em se aplicar repetidamente a regra da resolução a pares de cláusulas, gerando-se novas cláusulas até que se chegue à conclusão.

A nova cláusula gerada é chamada de **resolvente** das cláusulas originais.

Ex.:

p1: a V b (premissa)
p2: ¬a V c (premissa)
p3: ¬c V d (premissa)
C4: b V c (Resolução p1 e p2)
C5: b V d (Resolução p3 e C4)

Demonstrações utilizando Resolução

Um resultado interessante é que quando a regra é aplicada a duas cláusulas quaisquer, sendo uma delas unitária (apenas um literal), o resolvente terá sempre um literal a menos. Tal fato é explorado nas **refutações por resolução,** e a contradição é detectada pela dedução da **cláusula vazia ({}).**

Exemplo:

Provar, por uma refutação por resolução, o seguinte argumento lógico:

$$p, p \rightarrow q, q \rightarrow r \models r$$

Demonstrações utilizando Resolução

Prova Direta: p1: p (premissa)
p2: $\neg p \vee q$ (f. clausal premissa)
p3: $\neg q \vee r$ (f. clausal premissa)
C4: q (Resolução p1 e p2)
C5: r (Resolução p3 e C4)

Refutação: p1: p (premissa)
p2: $\neg p \vee q$ (f. clausal premissa)
p3: $\neg q \vee r$ (f. clausal premissa)
p4: $\neg r$ (neg. da conclusão)
C4: q (Resolução p1 e p2)
C5: r (Resolução p3 e C4)
C6: {} (Resolução p4 e C5)

Resolução por Refutação

- É uma regra de inferência **correta** (sound), ou seja, tudo que ela gera são teoremas verdadeiros;
- É uma regra de inferência **completa** para refutação, ou seja, todo teorema pode ser inferido por ela (no cálculo proposicional).

Conclusão

- A lógica proposicional fornece uma ferramenta formal para representar conhecimento e fazer inferências sobre ele.
- Apesar de ser simples, a lógica proposicional não consegue expressar fatos genéricos de modo conciso.
- Seria interessante poder expressar tais fatos genéricos através de um linguagem que pudesse fazer referência a objetos e a algumas de suas propriedades → **Lógica de Primeira Ordem**

Bibliografia

- Capítulo 7 do livro texto
  - Russel & Norvig, Inteligência Artificial, 3a. Edição