

02

Módulo 02 - Principais abordagens em IA - abordagem simbólica

Fundamentos de Inteligência Artificial (IA)

Resumo do Módulo

- ◆ Representação do Conhecimento e Lógica Formal.
- ◆ Sistemas Baseados em Regras.
- ◆ Limitações e Desafios Dessa Abordagem.
- ◆ Lab

Abordagem Simbólica

- ♦ A **abordagem simbólica** utiliza **representações explícitas de conhecimento**, como símbolos e regras lógicas, para realizar raciocínios e resolver problemas.



Cenários de utilização

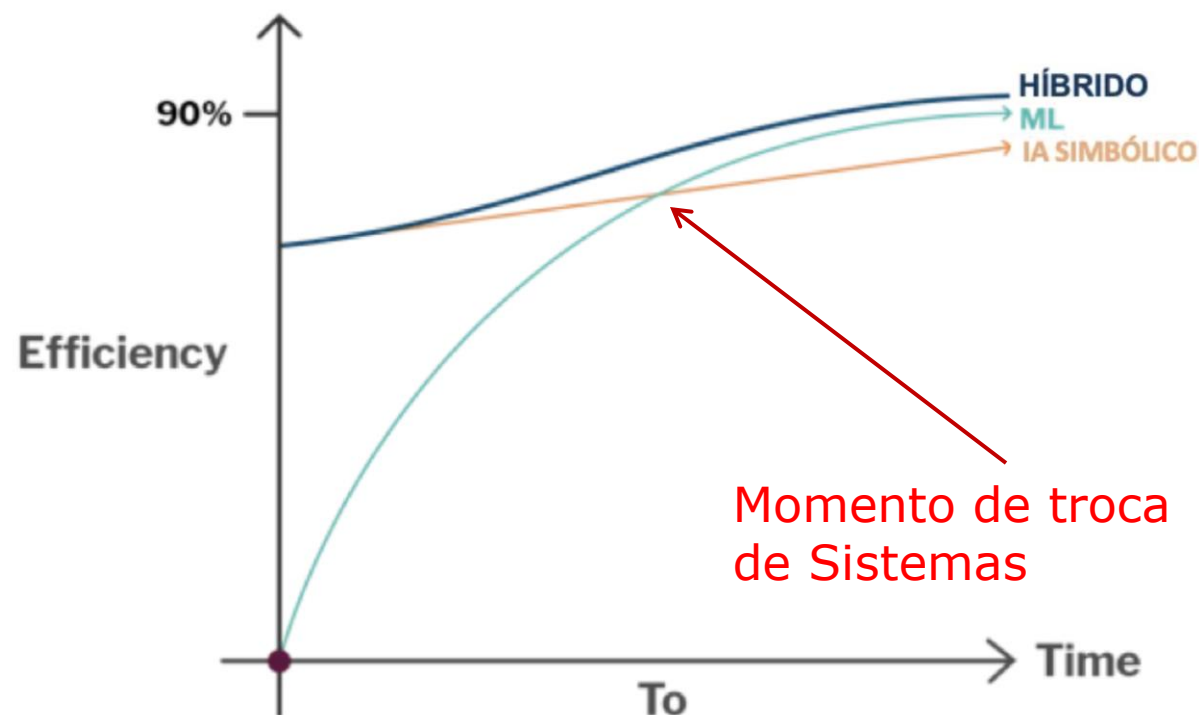
1. Regras absolutas – p.e. física, termodinâmica, etc

2. Pouca capacidade de processamento

3. Não existem dados iniciais

- ColdStart

4. Sistema Híbridos



Representação do Conhecimento com Símbolos

Imagine que temos um sistema que precisa diagnosticar uma doença com base em sintomas.

Símbolos:

S1: Febre.

S2: Tosse.

S3: Dor de garganta.

D1: Gripe.

D2: Constipação.

Lógica Formal (Raciocínio Lógico para Resolver o Problema)

1. O sistema recebe os seguintes sintomas de um paciente:

- febre (**S1**)
- tosse (**S2**)
- dor de garganta (**S3**)

2. O sistema aplica regras lógicas para realizar o diagnóstico:

- Regra 1: **$D1 = S1 \wedge S2 \wedge S3$**
- Regra 2: **$D2 = \neg S1 \wedge S2 \wedge S3$** (\neg é not)

Parece um sistema **if...else** mas não é!

Pode ter **if...else** ou não, mas tem que ser desenhado para ser **explicável!** – O que é **muito difícil de atingir!**

3. O sistema apresenta o resultado:

- D1 (Gripe) ←
- ~~D2 (Constipação)~~

(**Explicabilidade**) - O sistema pode explicar a sua decisão!

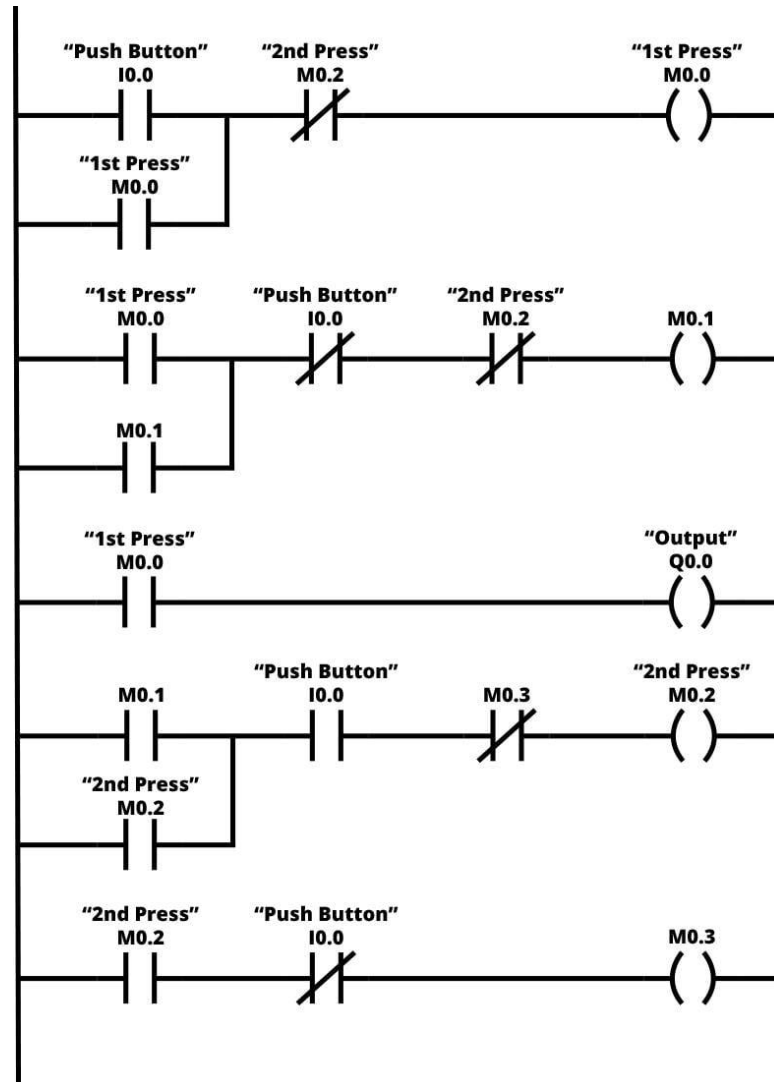
"Diagnóstico: Gripe. Justificação: O paciente tem febre, tosse e dor de garganta, o que corresponde às condições da Regra 1."

Exemplos

1. Que roupa vestir dependendo da temperatura?
2. O que comer ao pequeno-almoço?
3. O que fazer à tarde?
4. Quando ligar as luzes em casa?
5. Quando fazer a manutenção do carro?

Sistema similar (embora não igual)

◆ladder

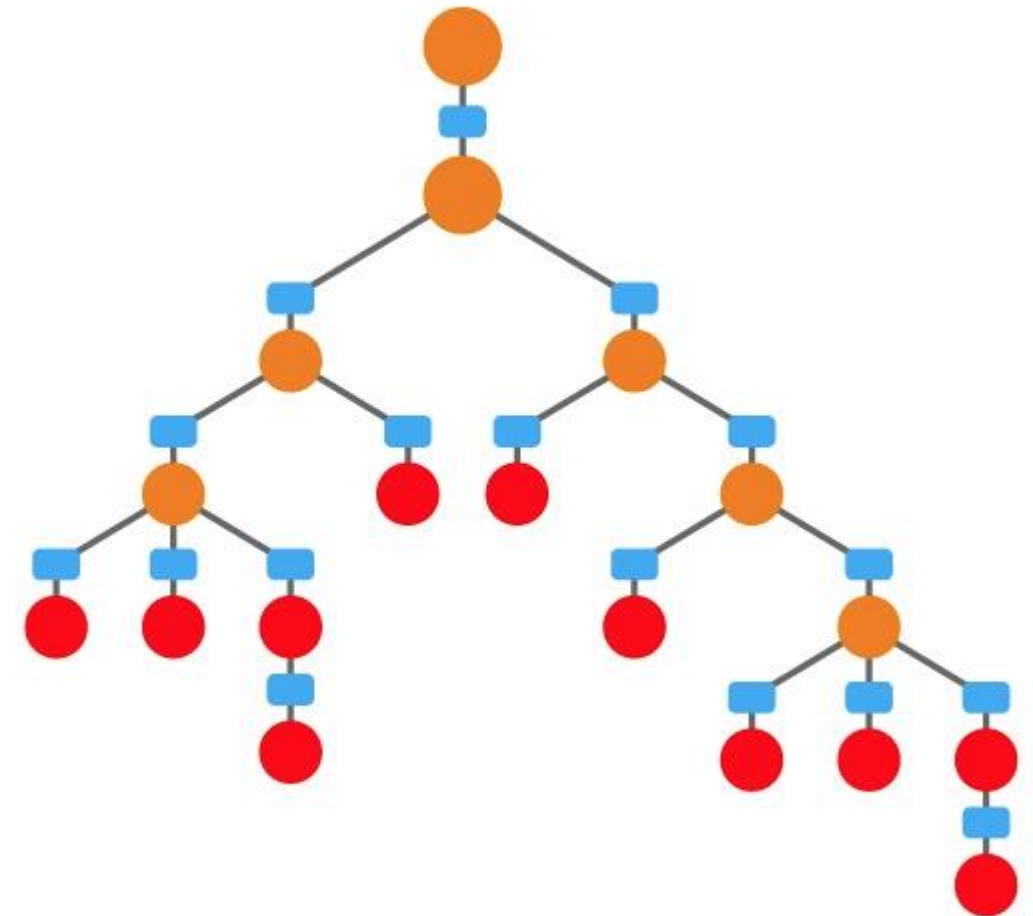


Regras

Regras são declarações condicionais que expressam o conhecimento de um domínio específico.

SE (**condições**) ENTÃO (**ações**)

- ♦ **Condições**: São os fatos ou padrões que precisam ser satisfeitos para que a regra seja aplicada.
- ♦ **Ações**: São as consequências ou conclusões que se seguem quando as condições são verdadeiras.



← **Ações** podem ser outras **Regras**

Simples if...elif...else vs IA Simbólica

SAFE STATE:

A decisão inicial da regra deve ser a que torna o sistema seguro

```
# Definir o tempo disponível atual
tempo_atual = "pressa"
# Pode ser "pressa", "tempo_disponivel"
# ou "tempo_moderado"

# Tomar decisão com base no tempo disponível
if tempo_atual == "pressa":
    print("Come uma barra de cereais.")
elif tempo_atual == "tempo_disponivel":
    print("Come um pequeno-almoço completo.")
elif tempo_atual == "tempo_moderado":
    print("Come uma refeição ligeira.")
else:
    print("Tempo desconhecido.")
```

```
# Representação e inicialização dos símbolos e ações
T1 = "pressa"
T2 = "tempo_disponivel"
T3 = "tempo_moderado" # Novo símbolo
A1 = "Come uma barra de cereais."
A2 = "Come um pequeno-almoço completo."
A3 = "Come uma refeição ligeira." # Nova ação
```

```
# Definir o tempo disponível atual
tempo_atual = T1 # Pode ser T1 (pressa), T2
                 # (tempo_disponivel) ou T3 (tempo_moderado)
```

```
# Inicializar a decisão com "Tempo desconhecido."
decisao = "Tempo desconhecido."
```

```
# Aplicar as regras simbolicamente
```

```
if tempo_atual == T1:
    decisao = A1
```

```
if tempo_atual == T2:
    decisao = A2
```

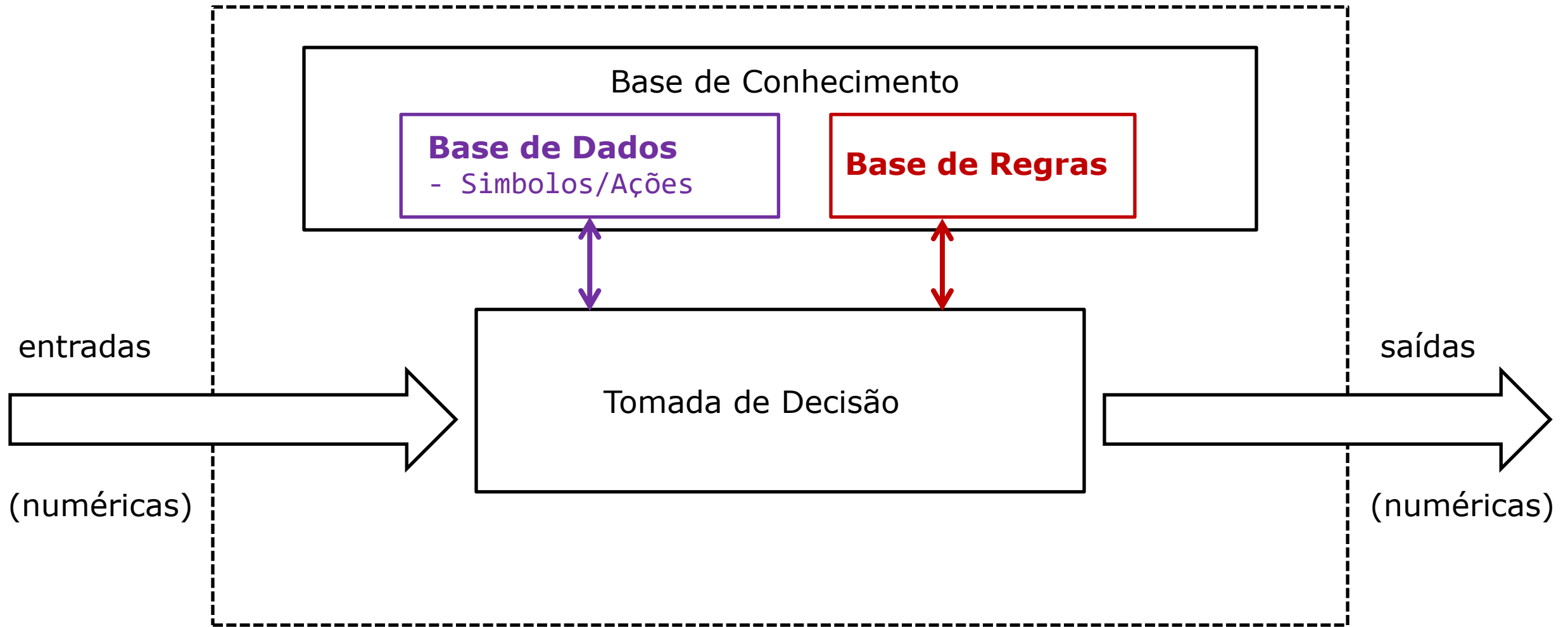
```
if tempo_atual == T3:
    decisao = A3 # Nova regra adicionada
```

```
print(decisao) # Imprimir a decisão final
```

Simples if...elif...else vs IA Simbólica

Aspeto	Simples (if...elif...else)	IA Simbólica
Conhecimento	Condições diretas e específicas	Representado por símbolos e regras explícitas
Regras de Decisão	Condições codificadas, difícil de expandir	Baseadas em regras, fácil de expandir
Escalabilidade	Complexidade crescente, difícil de manter	Modular e escalável
Raciocínio Explicável	Lógica implícita, difícil de explicar	Justificável com base nas regras
Generalização	Específico ao problema original	Adaptável a diferentes contextos

Sistemas Baseados em Regras



IA Simbólica mais condensada e mais aplicável ao sistema

```
# Representação e inicialização dos símbolos e ações
```

```
T1 = "pressa"
```

```
T2 = "tempo_disponivel"
```

```
T3 = "tempo_moderado"
```

```
A1 = "Come uma barra de cereais."
```

```
A2 = "Come um pequeno-almoço completo."
```

```
A3 = "Come uma refeição ligeira."
```

```
# Definir o tempo disponível atual
```

```
tempo_atual = T1 # Pode ser T1 (pressa), T2  
(tempo_disponivel) ou T3 (tempo_moderado)
```

```
# Inicializar a decisão com "Tempo desconhecido."
```

```
decisao = "Tempo desconhecido."
```

```
# Aplicar as regras simbolicamente
```

```
if tempo_atual == T1:  
    decisao = A1
```

```
if tempo_atual == T2:  
    decisao = A2
```

```
if tempo_atual == T3:  
    decisao = A3
```

```
print(decisao) # Imprimir a decisão final
```

```
# Dicionário de símbolos e ações e sua inicialização
```

```
sistema = {
```

```
    "T1": "pressa",
```

```
    "T2": "tempo_disponivel",
```

```
    "T3": "tempo_moderado",
```

```
    "A1": "Come uma barra de cereais.",
```

```
    "A2": "Come um pequeno-almoço completo.",
```

```
    "A3": "Come uma refeição ligeira."
```

```
}
```

```
def tomar_decisao(tempo_atual):
```

```
    # Inicializar a decisão com "Tempo desconhecido."
```

```
    decisao = "Tempo desconhecido."
```

```
    # Aplicar as regras simbolicamente
```

```
    if tempo_atual == sistema["T1"]:  
        decisao = sistema["A1"]
```

```
    elif tempo_atual == sistema["T2"]:  
        decisao = sistema["A2"]
```

```
    elif tempo_atual == sistema["T3"]:  
        decisao = sistema["A3"]
```

```
    return decisao
```

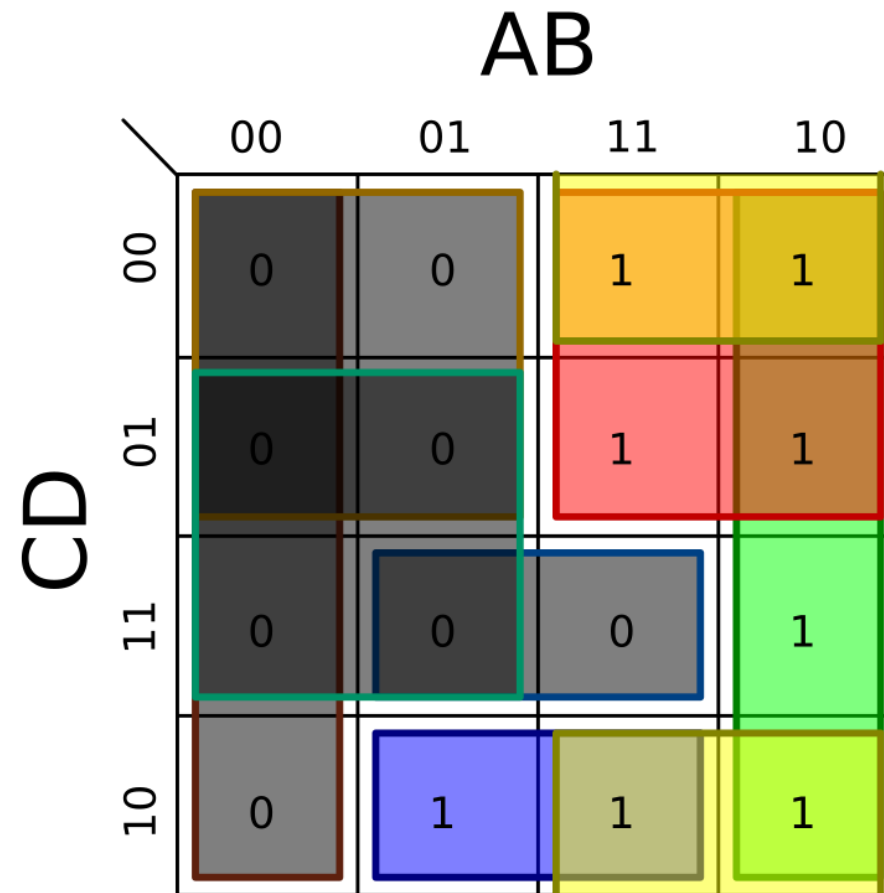
```
tempo_atual = sistema["T1"]
```

```
decisao_final = tomar_decisao(tempo_atual)
```

```
print(decisao_final) # Imprimir a decisão final
```

Método de simplificação gráfico das regras

♦ Mapa de Karnaugh



$$f(A,B,C,D) = E(6,8,9,10,11,12,13,14)$$

$$F = AC' + AB' + BCD' + AD'$$

$$F = (A+B)(A+C)(B'+C'+D')(A+D')$$

- ◆ Lab 02 - Principais abordagens em IA - abordagem simbólica