

Prova por Contradição, Horn Clauses e Diagnóstico por Consistência

1 Motivação: Provar por Contradição em Bases de Conhecimento

Exposição.

- Em muitas aplicações queremos saber *que combinações de fatos são impossíveis*.
- Exemplo: no domínio elétrico, queremos proibir situações como
“a mesma lâmpada está, ao mesmo tempo, lit e dark”.
- Para isso, precisamos representar contradições e raciocinar *por absurdo*.

Coneectar com lógica clássica.

- Prova por contradição: supomos algo, derivamos false, concluímos que a suposição é impossível.
- Em bases de conhecimento, vamos capturar isso com uma constante especial *false* e com *integrity constraints*.

2 Horn Clauses e Integrity Constraints

Definição 1 (Integrity constraint). Uma restrição de integridade é uma cláusula da forma

$$\text{false} \leftarrow a_1 \wedge \dots \wedge a_k,$$

em que cada a_i é um átomo e *false* é um átomo especial, que deve ser falso em toda interpretação.

Definição 2 (Horn clause). Uma Horn clause é:

- ou uma cláusula definida usual ($h \leftarrow a_1 \wedge \dots \wedge a_m$),
- ou uma restrição de integridade ($\text{false} \leftarrow a_1 \wedge \dots \wedge a_k$).

Ou seja, a cabeça de uma Horn clause é um átomo comum ou o átomo especial *false*.

2.1 Equivalência lógica

Exposição.

- A integridade

$$\text{false} \leftarrow a_1 \wedge \dots \wedge a_k$$

é logicamente equivalente à cláusula

$$\neg a_1 \vee \dots \vee \neg a_k.$$

- Interpretação: “pelo menos um dos a_i tem que ser falso”.
- Diferentemente de uma base apenas com cláusulas definidas, uma KB com Horn clauses pode *implicar negações* de átomos.

2.2 Exemplo simples: KB_1 (Exemplo 5.17)

Considere:

$$\begin{aligned} & \text{false} \leftarrow a \wedge b. \\ & a \leftarrow c. \\ & b \leftarrow c. \end{aligned}$$

Raciocínio em sala.

- Suponha que c seja verdadeiro em um modelo I .
- Então a e b também são verdadeiros em I (pelos duas regras).
- Como *false* deve ser sempre falso, a cláusula

$$\text{false} \leftarrow a \wedge b$$

fica violada em I (corpo verdadeiro, cabeça falsa) $\Rightarrow I$ não é modelo.

- Logo, **não** existe modelo de KB_1 em que c seja verdadeiro. Conclusão:

$$KB_1 \models \neg c.$$

2.3 Exemplo: disjunções de negações (Exemplo 5.18)

Considere:

$$\begin{aligned} \textit{false} &\leftarrow a \wedge b. \\ a &\leftarrow c. \\ b &\leftarrow d. \\ b &\leftarrow e. \end{aligned}$$

Discussão.

- Se c e d fossem ambos verdadeiros em um modelo I , então a e b seriam verdadeiros, violando a integridade. Logo, em todo modelo:

$$KB_2 \models \neg c \vee \neg d.$$

- Pelo mesmo raciocínio:

$$KB_2 \models \neg c \vee \neg e.$$

- Isto mostra que, mesmo sem escrever disjunções e negações na entrada, conseguimos derivar sentenças desse tipo.

2.4 Satisfatibilidade e inconsistência

Pontos importantes.

- Um conjunto de cláusulas é *insatisfatível* se não tem modelos.
- Ele é *provavelmente inconsistente* em relação a um procedimento de prova se *false* pode ser derivado.
- Para bases apenas com *cláusulas definidas*, sempre existe modelo (por exemplo, a interpretação que faz todos os átomos verdadeiros).
- Já um conjunto de *Horn clauses* pode ser insatisfatível, por exemplo:

$$\{a, \textit{false} \leftarrow a\}.$$

3 Assumíveis e Conflitos

Definição 3 (Assumível). *Um assumível (assumable) é um átomo que pode ser assumido em uma prova por contradição. Ao provar false a partir de assumíveis, obtemos disjunções de negações desses assumíveis.*

Definição 4 (Conflito). *Se KB é um conjunto de Horn clauses, um conflito de KB é um conjunto de assumíveis*

$$C = \{c_1, \dots, c_r\}$$

tal que

$$KB \cup \{c_1, \dots, c_r\} \models \textit{false}.$$

Nesse caso, podemos concluir:

$$KB \models \neg c_1 \vee \dots \vee \neg c_r.$$

Definição 5 (Conflito mínimo). *Um conflito C é mínimo se nenhum subconjunto próprio de C também é conflito.*

3.1 Exemplo: conflitos em KB_2 (Exemplo 5.20)

Se o conjunto de assumíveis é $\{c, d, e, f, g, h\}$, então:

- $\{c, d\}$ é um conflito mínimo de KB_2 .
- $\{c, e\}$ também é um conflito mínimo.
- $\{c, d, e, h\}$ é um conflito, mas **não** mínimo.

4 Diagnóstico Baseado em Consistência (CBD)

Ideia geral.

- Temos uma descrição de *como o sistema deveria funcionar* (modelo nominal).
- Temos observações (por exemplo, lâmpadas apagadas, apesar de interruptores ligados).
- Fazemos suposições de normalidade (`ok_cb1`, `ok_s1`, ...) e vemos quais combinações são *incompatíveis* com as observações.

Noções-chave.

- **Assumíveis** representam componentes supostamente normais (ou eventualmente falhos).
- **Conflitos** são conjuntos de suposições que não podem ser todas verdadeiras.
- A partir dos conflitos, obtemos *diagnósticos*: conjuntos de componentes que devem estar falhos para explicar tudo.

4.1 Exemplo: circuito elétrico (Exemplos 5.21–5.22)

Contexto.

- Domínio: casa com disjuntores $cb1, cb2$, chaves $s1, s2, s3$ e lâmpadas $l1, l2$.
- A KB contém:
 - Regras que definem quando fios estão `live_wX`.
 - Regras que definem quando lâmpadas estão `lit_l1, dark_l1`, etc.
 - Restrições de integridade para proibir uma lâmpada de ser, ao mesmo tempo, `lit` e `dark`:

$$\text{false} \leftarrow \text{dark_l1} \wedge \text{lit_l1}, \quad \text{false} \leftarrow \text{dark_l2} \wedge \text{lit_l2}.$$

- Assumíveis de normalidade:

$$\text{ok_cb1}, \text{ok_cb2}, \text{ok_s1}, \text{ok_s2}, \text{ok_s3}, \text{ok_l1}, \text{ok_l2}.$$

- Observações:

$$\text{up_s1}, \text{up_s2}, \text{up_s3}, \text{dark_l1}, \text{dark_l2}.$$

Conflitos encontrados. O sistema encontra dois conflitos mínimos:

$$\begin{aligned} &\{\text{ok_cb1}, \text{ok_s1}, \text{ok_s2}, \text{ok_l1}\}, \\ &\{\text{ok_cb1}, \text{ok_s3}, \text{ok_l2}\}. \end{aligned}$$

Interpretando:

- Em qualquer modelo compatível com as observações, *não* é possível que todos esses componentes estejam ok ao mesmo tempo.
- Em termos lógicos:

$$\begin{aligned} KB \models \neg \text{ok_cb1} \vee \neg \text{ok_s1} \vee \neg \text{ok_s2} \vee \neg \text{ok_l1}, \\ KB \models \neg \text{ok_cb1} \vee \neg \text{ok_s3} \vee \neg \text{ok_l2}. \end{aligned}$$

4.2 Diagnósticos mínimos (Exemplo 5.22)

Ideia.

- Dado o conjunto de conflitos, um **diagnóstico** é um conjunto de assumíveis que tem pelo menos um elemento de cada conflito.
- Um diagnóstico *mínimo* é aquele em que nenhum subconjunto também é diagnóstico.

Resultado do exemplo. No caso do circuito, os diagnósticos mínimos correspondem a:

$$\{\text{ok_cb1}\}, \quad \{\text{ok_s1}, \text{ok_s3}\}, \quad \{\text{ok_s1}, \text{ok_l2}\}, \quad \{\text{ok_s2}, \text{ok_s3}\}, \quad \{\text{ok_s2}, \text{ok_l2}\}, \quad \{\text{ok_l1}, \text{ok_s3}\}, \quad \{\text{ok_l1}, \text{ok_l2}\}.$$

Em palavras: pelo menos uma dessas combinações de componentes *não* está ok no mundo real.

5 Implementações com Horn Clauses e Assumíveis

5.1 Visão geral do procedimento bottom-up (Figura 5.9)

Ideia.

- Estender o algoritmo bottom-up de cláusulas definidas.
- Em vez de guardar apenas “átomo é consequência”, guardamos pares

$$\langle a, A \rangle$$

em que a é um átomo e A é um conjunto de assumíveis que implicam a .

Passos.

1. Inicializar C com todos assumíveis:

$$C := \{\langle a, \{a\} \rangle : a \text{ é assumível}\}.$$

2. Enquanto possível, para cada cláusula

$$h \leftarrow b_1 \wedge \dots \wedge b_m$$

se para cada b_i existe $\langle b_i, A_i \rangle \in C$, então adicionamos

$$\langle h, A_1 \cup \dots \cup A_m \rangle.$$

3. Quando geramos $\langle \text{false}, A \rangle$, o conjunto A é um conflito.

Refinamento.

- Podemos podar supersets: se já temos $\langle a, A_1 \rangle$ e surge $\langle a, A_2 \rangle$ com $A_1 \subset A_2$, podemos descartar o segundo.

5.2 Visão geral do procedimento top-down (Figura 5.10)

Ideia.

- Versão top-down do raciocínio com Horn clauses.
- Queremos provar false , começando de $G = \{\text{false}\}$.

Passos.

1. Manter um conjunto G de átomos “a provar” que, juntos, implicam false .
2. Enquanto existir átomo $a \in G$ que não é assumível:
 - Escolher uma cláusula $a \leftarrow B$ em KB .
 - Atualizar $G := (G \setminus \{a\}) \cup B$.
3. Quando G ficar formado apenas por assumíveis, G é um conflito.

Observação.

- Como na derivação top-down de cláusulas definidas, escolhas diferentes de cláusulas podem levar a conflitos diferentes ou a falha.

6 Discussão Final e Exercícios

Pontos para discussão em sala.

- Qual a vantagem de ter explicitamente false na linguagem?
Permite escrever restrições de integridade e detectar inconsistências simplesmente tentando provar false .
- Por que é útil saber quais combinações de assumíveis geram contradição?
Porque essas combinações indicam quais conjuntos de hipóteses não podem ser verdadeiros ao mesmo tempo, ajudando em diagnóstico, planejamento e projeto.
- Como conflitos se relacionam com diagnósticos mínimos?
Conflitos são conjuntos de assumíveis incompatíveis; diagnósticos mínimos são conjuntos de assumíveis que intersectam todos os conflitos (têm pelo menos um elemento de cada conflito).

Sugestão de exercícios.

1. Considere uma KB simples com integridade:

$$\text{false} \leftarrow \text{alarm} \wedge \text{quiet}.$$

e regras para alarm e quiet . Proponha um conjunto de assumíveis e encontre um conflito mínimo.

2. A partir dos conflitos do circuito elétrico, peça aos alunos que:

- listem diagnósticos mínimos;
- expliquem em linguagem natural o significado de cada diagnóstico.

Abdução e Explicações em Bases de Conhecimento

7 Motivação: O que é Abdução?

Exposição.

- Enquanto **dedução** determina o que *segue logicamente* de axiomas,
- e **indução** infere generalizações a partir de exemplos,
- a **abdução** busca hipóteses que *expliquem observações*.

Exemplos introdutórios:

- Observar que uma lâmpada está apagada \Rightarrow considerar hipóteses (queimada, falta de energia, interruptor desligado).
- Em sistemas tutores, inferir o que o aluno sabe a partir de erros cometidos.

A noção foi introduzida por **Charles Peirce**.

8 Formalização da Abdução

Definição 6 (Assumables). *Seja A um conjunto de átomos que podem ser assumidos como hipóteses. Estes são os blocos básicos para construir explicações.*

Definição 7 (Cenário). *Dado $\langle KB, A \rangle$, um cenário é um conjunto $H \subseteq A$ tal que $KB \cup H$ é satisfatível, isto é, não contém contradições.*

Definição 8 (Explicação). *Uma explicação de g é um conjunto $H \subseteq A$ tal que:*

$$KB \cup H \models g \quad \text{e} \quad KB \cup H \not\models \text{false}.$$

Definição 9 (Explicação mínima). *Uma explicação H é mínima se nenhum subconjunto estrito de H também explica g.*

8.1 Comentário didático

- Observações **não são adicionadas** à KB — devem ser explicadas.
- Contradições são proibidas: qualquer conjunto que implique **false** é descartado.

9 Exemplo 1: Diagnóstico Médico Simples (Exemplo 5.31)

Considere a KB:

```
bronchitis ← influenza.  
bronchitis ← smokes.  
coughing ← bronchitis.  
wheezing ← bronchitis.  
fever ← influenza.  
fever ← infection.  
sore_throat ← influenza.  
false ← smokes ∧ nonsmoker.
```

Assumables:

$\{smokes, nonsmoker, influenza, infection\}$.

9.1 Exemplo A — Observação: wheezing

Para explicar *wheezing*:

```
wheezing ← bronchitis.  
bronchitis ← influenza ou bronchitis ← smokes.
```

Explicações mínimas:

$\{influenza\}$ ou $\{smokes\}$.

9.2 Exemplo B — Observação: wheezing \wedge fever

$$\text{fever} \leftarrow \text{influenza} \text{ ou } \text{infection}.$$

Explicações mínimas:

$$\{\text{influenza}\} \text{ e } \{\text{smokes}, \text{infection}\}.$$

9.3 Exemplo C — Observação: wheezing \wedge nonsmoker

Como assumir `smokes` leva a contradição:

$$\{\text{smokes}, \text{nonsmoker}\} \models \text{false},$$

a única explicação mínima possível é:

$$\{\text{influenza}, \text{nonsmoker}\}.$$

10 Exemplo 2: Sistema de Alarme (Exemplo 5.32)

Base de conhecimento:

$$\begin{aligned} \text{alarm} &\leftarrow \text{tampering}. \\ \text{alarm} &\leftarrow \text{fire}. \\ \text{smoke} &\leftarrow \text{fire}. \end{aligned}$$

10.1 Observação: alarm

Explicações mínimas:

$$\{\text{tampering}\}, \quad \{\text{fire}\}.$$

10.2 Observação: alarm \wedge smoke

A presença de `smoke` “explica” o alarme:

$$\{\text{fire}\}.$$

Não há necessidade de assumir `tampering`: hipótese explicada por outra evidência.

11 Abdução vs. Diagnóstico Baseado em Consistência (CBD)

Principais diferenças.

- Em CBD:
 - Assume-se comportamento **normal**.
 - Observações são adicionadas à KB.
 - Procura-se um conjunto de componentes que, se defeituosos, explique a inconsistência.
- Em Abdução:
 - Hipóteses incluem **normalidade e falhas**.
 - Observações devem ser explicadas, não adicionadas.
 - Requer modelagem mais detalhada (regra para cada forma de comportamento).

Consequência. Abdução produz diagnósticos mais detalhados, mas exige modelagem mais rica.

12 Exemplo 3 — Diagnóstico Elétrico (Exemplo 5.33)

Regras simplificadas:

$$\begin{aligned} \text{lit_l1} &\leftarrow \text{live_w0} \wedge \text{ok_l1}. \\ \text{dark_l1} &\leftarrow \text{broken_l1}. \\ \text{dark_l1} &\leftarrow \text{dead_w0}. \\ \text{false} &\leftarrow \text{ok_l1} \wedge \text{broken_l1}. \end{aligned}$$

Assumíveis:

$$\{\text{ok_l1}, \text{broken_l1}, \text{live_w0}, \text{dead_w0}\}.$$

Observações como `dark_l1` ou `lit_l1` são explicadas por hipóteses que tornam essas regras verdadeiras.

13 Discussão Final e Exercícios

Perguntas para debate.

- Por que permitir hipóteses é útil, mas perigoso?
- O que impede que o sistema simplesmente assuma tudo?
- Como garantir minimalidade das explicações?

Exercício para casa.

- Modele um domínio simples (ex.: funcionamento de um computador) com:
 - KB com Horn clauses,
 - assumables (normais e defeituosos),
 - observações.
- Encontre explicações mínimas para duas observações distintas.