nemo
ontology & conceptual
modeling research group

Introdução à Computação

Jordana Sarmenghi Salamon
jssalamon@inf.ufes.br
jordanasalamon@gmail.com
http://inf.ufes.br/~jssalamon

Departamento de Informática
Universidade Federal do Espírito Santo

Agenda

- Noções de Álgebra Booleana
  - Álgebra de Boole
  - Axiomas da Álgebra de Boole
  - Álgebra de Boole de dois valores literais
  - Teoremas da Álgebra de Boole
  - Simplificação de expressões booleanas

Álgebra de Boole - George Boole (1815-1864)

- **Ideias de Boole**

- Usar **símbolos algébricos** como x, y, z, p, q, r para denotar palavras, frases, ou proposições;
- O que Boole estava pensando era em criar um **sistema algébrico** com operações como adição e multiplicação e métodos de resolução de equações;
- A Álgebra de Boole exigia a formulação de uma linguagem simbólica do pensamento;
- Resolver uma equação em tal linguagem não levaria a uma resposta numérica, mas sim a uma **conclusão lógica**.
- Sua álgebra seria a “álgebra do pensamento”.

Álgebra de Boole

- **Álgebra Booleana (noções)**
- Álgebra Booleana é uma variante de álgebra ordinária como ensinado no ensino médio;
- Ela difere da ordinária, basicamente, em três coisas:
- (i) nos **valores que as variáveis podem assumir**, que são de caráter lógico (zero e um indicando falso e verdadeiro, respectivamente);
- (ii) nas **operações** aplicáveis a esses valores;
- (iii) e nas **propriedades** dessas operações, i.e., as leis que elas obedecem.

Álgebra de Boole

- **Álgebra Booleana (noções)**
- Álgebra "Tradicional":
  - Variáveis representam números reais;
  - Operadores são aplicados às variáveis e o resultado é um número real;
- Álgebra Booleana:
  - Variáveis representam apenas 0 ou 1
  - Operadores retornam apenas 0 ou 1

Álgebra de Boole

- **Álgebra Booleana (noções)**
- Por exemplo, supondo:
  - x = jovem;
  - y = faz Ciência da Computação;
- (1 - x) iria então representar a operação de selecionar todas as coisas no mundo exceto jovens, isto é, todas as coisas que não são jovens;
- (xy) representaria o conjunto dos jovens que fazem Ciência da Computação;
- (1 - x) (1 - y) seria todas as coisas que não são jovens nem fazem Ciência da Computação;
- (x + y) seria o conjunto das coisas que são jovens ou que fazem Ciência da Computação.

Álgebra de Boole

- **Operações booleanas básicas**
- $x \wedge y = xy$ (conjunção)
- $x \vee y = x + y$ (disjunção)
- $\neg x = 1 - x$ (negação)

- Valores podem ser obtidos através da tabela verdade

x	y	$x \wedge y$	$x \vee y$
0	0	0	0
1	0	0	1
0	1	0	1
1	1	1	1

x	$\neg x$
0	1
1	0

- Ou com equações gerando os valores explicitamente

$0 \wedge 0 = 0$ $\quad$ $0 \vee 0 = 0$

$0 \wedge 1 = 0$ $\quad$ $0 \vee 1 = 1$

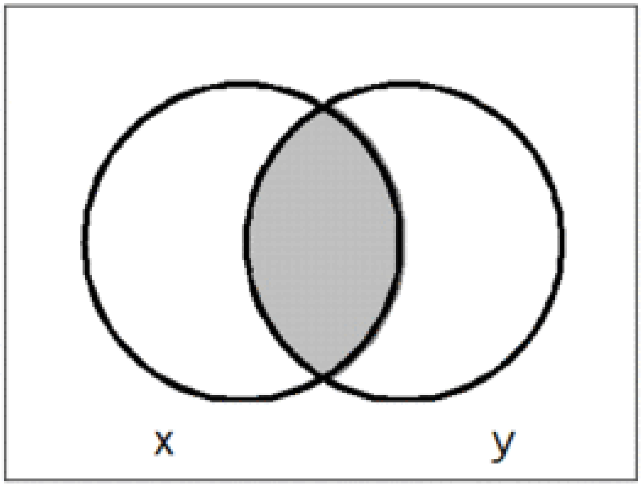
$1 \wedge 0 = 0$ $\quad$ $1 \vee 0 = 1$

$1 \wedge 1 = 1$ $\quad$ $1 \vee 1 = 1$

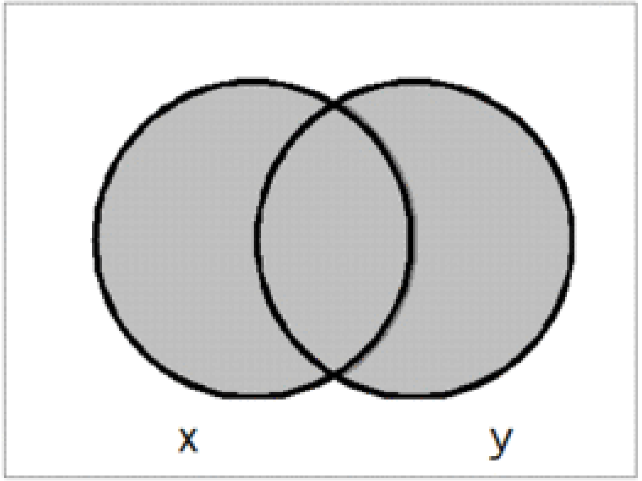
$\neg 0 = 1$ $\quad$ $\neg 1 = 0$

Álgebra de Boole

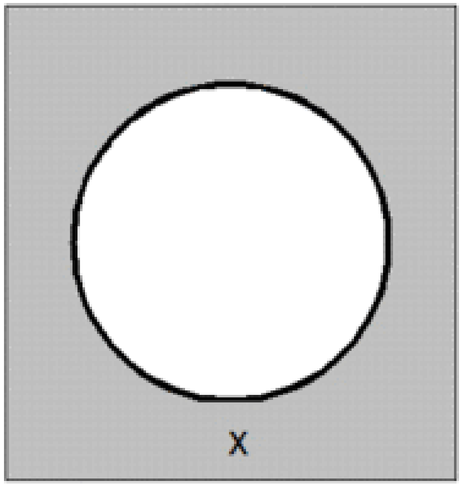
- **Representação diagramática**
- Diagramas de Venn para conjunção, disjunção e complemento.



x^y



x∨y



¬x

Álgebra de Boole

- **Operações booleanas derivadas**

$x \to y = \neg(x \wedge \neg y)$ (Condicional)

$x \oplus y = (x \vee y) \wedge \neg(x \wedge y)$ (ou exclusivo)

$x \equiv y = \neg(x \oplus y)$ (equivalência)

- Valores podem ser obtidos através da tabela verdade

x	y	$x \to y$	$x \oplus y$	$x \equiv y$
0	0	1	0	1
1	0	0	1	0
0	1	1	1	0
1	1	1	0	1

Definição de um sistema algébrico qualquer

- Chamamos de álgebra abstrata ou sistema algébrico um conjunto não vazio munido de um ou mais operadores binários sobre ele definidos;
- Denotando por A o conjunto e por * e Δ os operadores definidos sobre A, podemos ter:
  - (A, *) ou (A, Δ) com um operador
  - ou (A, *, Δ) com dois operadores

Definição da Álgebra de Boole

- O sistema algébrico (B, +, .) é uma Álgebra de Boole se e
- somente se ∀ a, b, c ϵ B valem os axiomas:
- A1. a + b ϵ B (oclusão ou encerramento)
- A2. a . b ϵ B (oclusão ou encerramento)
- A3. a + b = b + a (comutatividade)
- A4. a · b = b · a (comutatividade)
- A5. a + (b·c) = (a+b) · (a+c) (distributividade)
- A6. a · (b+c) = (a·b) + (a·c) (distributividade)

Definição da Álgebra de Boole

- A7. a + 0 = 0 + a = a (identidade)
- A8. a . 1 = 1 . a = a (identidade)
- A9. a + a’ = 1 e a . a’ = 0
- (o elemento a’ chama-se complemento de a)

Definição da Álgebra de Boole

- **Resumo:**
- **Operações binárias:{· , +}**
- **Operações unárias: {‘}, também pode ser utilizado o ¯**
- **Regras de prioridade: ‘, ·, +**

Definição da Álgebra de Boole

- **Literais**
- Cada aparecimento de uma variável ou do seu complemento numa expressão booleana é designado por um literal;
- Uma expressão booleana é mais simples do que outra se tiver menos literais (pode indicar que contém menor número de operações);
- Exemplo: ab’c+ab+a’bc’+b’c ----------> 10 literais

Álgebra de Boole de dois valores

- **Símbolos:**
- AND (E)
  - Símbolo: · ou nenhum
- OR (OU)
  - Símbolo: +
- NOT (COMPLEMENTO)
  - Símbolo: ‘ ou ¯
- Também são encontrados na literatura os seguintes símbolos: AND (&, ^); OR (|, v); NOT(~, !)

Álgebra de Boole de dois valores

X	Y	X AND Y
0	0	0
0	1	0
1	0	0
1	1	1

X	NOT X
0	1
1	0

X	Y	X OR Y
0	0	0
0	1	1
1	0	1
1	1	1

- B contem dois elementos {0, 1} e $0 \neq 1$
- Oclusão
- Comutatividade
- Identidade
- Complemento

Álgebra de Boole de dois valores

- **Prova de distributiva de .**

x	y	z	y+z	x(y+z)	xy	xz	xy+xz
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

Álgebra de Boole de dois valores

- **Prova de distributiva de +**

x	y	z	yz	x+(yz)	x+y	x+z	(x+y)(x+z)
0	0	0	0	0	0	0	0
0	0	1	0	0	0	1	0
0	1	0	0	0	1	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

Exemplo de uso das propriedades (1)

- Mostrar que abc’ equivale a c’ba
  - Usar a propriedade comutativa:
  - a.b.c’ = a.c’.b = c’.a.b = c’.b.a = c’ba
- Mostrar que abc + abc’ = ab
  - Usar a propriedade distributiva de .
  - abc + abc’ = ab(c+c’)
  - Complemento: trocar c+c’ por 1: ab(c+c’) = ab(1)
  - Identidade
  - ab(1) = ab.1 = ab

Exemplo de uso das propriedades (2)

- Mostrar que x + x'z equivale a x + z
  - Usar a propriedade distributiva de +
  - Trocar x+x'z por (x+x').(x+z)
  - Complemento: trocar (x+x') por 1
  - Identidade: trocar 1(x+z) por x+z

Formalização do pensamento humano

- Converter as seguintes afirmações em língua portuguesa para equações booleanas:
- Q1. Verdade se a é 1 e b é 1.
  - Resposta: $F = a . b$
- Q2. Verdade se a ou b forem 1.
  - Resposta: $F = a + b$
- Q4. Verdade se a é 1 e b é 0.
  - Resposta: $F = a . b'$

Avaliando equações booleanas

- Avaliar a equação booleana F = (a.b) + (c.d) para os seguintes valores de a, b, c, e d:
- Q1: a=1, b=1, c=1, d=0
  - Resposta: F = (1 . 1) + (1 . 0) = 1 + 0 = 1
- Q2: a=0, b=1, c=0, d=1.
  - Resposta: F = (0 . 1) + (0 . 1) = 0 + 0 = 0

Avaliando equações booleanas

- Ex: “Eu irei almoçar se Maria **ou** João forem **e** se Célia **não** for.”
- Suponha que F represente o meu comparecimento ao almoço (1 significa presença, 0 indica ausência)
- Do mesmo modo, (m) significa a presença de Maria, (j) a de João e (c) a de Célia
- F = (m OR j) AND NOT(c)
- $F = (m + j)\, c'$

Avaliando equações booleanas

- **Avaliação formal**
- Para m=1, j=0, c=1
- F = (m + j) c’
- F = (1 + 0) . 1’ = 1 . 0 = **0**

Teoremas da Álgebra de Boole

- **Idempotência**
- $a + a = a$
- $a . a = a$
- **Elemento Nulo**
- $a + 1 = 1$
- $a . 0 = 0$

- **Involução**
- $(a')' = a$

- **Associatividade**
- $(a+b)+c = a+(b+c)$
- $(a.b).c = a.(b.c)$

- **De Morgan**
- $(ab)' = a' + b'$
- $(a + b)' = a' b'$

Teorema de De Morgan

- As leis de De Morgan são muito importantes para fazer inferências válidas em provas e argumentos dedutivos;
- (i) $(a+b+c+...)'=a'.b'.c'....$
- (ii) $(a.b.c....)'=a'+b'+c'+...$

Simplificação de expressões booleanas

- Exemplo:
- Simplificar a expressão (abc) + (ac’) + (ab’)
- a(bc + c’ + b’)
- a(bc + (bc)’)
- Podemos chamar bc de X, então:
- a(X + X’)
- a(1)
- a

Simplificação de expressões booleanas

- Exercício:
- Simplificar a expressão a’b’c’ + a’bc + a’bc’ + ab’c’ + abc’

Simplificação de expressões booleanas

- Resolução:
- a’b’c’ + a’bc + a’bc’ + ab’c’ + abc’
- c’(a’b’ + a’b + ab’ + ab) + a’bc
- c’(a’(b’+b) + a(b’+b)) + a’bc
- c’(a’ + a) + a’bc
- c‘ + a’bc
- ((c‘ + a’bc)’)’
- ((c’)’ (a’bc)’)’

- (c . (a + b’ + c’))’
- (ca + cb’ + cc’)’
- (ca + cb’)’
- (c(a + b’))’
- c’ + (a+b’)’
- c’ + a’b

nemo
ontology & conceptual
modeling research group

Introdução à Computação

Jordana Sarmenghi Salamon
jssalamon@inf.ufes.br
jordanasalamon@gmail.com
http://inf.ufes.br/~jssalamon

Departamento de Informática
Universidade Federal do Espírito Santo