Matemática

Funções

Prof. Edson Alves

Faculdade UnB Gama

Funções

Produtos Cartesianos

Definição

Sejam A e B dois conjuntos. O **produto cartesiano** de A por B é dado por

$$A \times B = \{(a, b) \mid a \in A, b \in B\}$$

- Em outras palavras, o produto cartesiano é o conjunto de todos os possíveis pares ordenados cujo primeiro elemento pertence a A e o segundo pertence a B
- Se A tem n elementos e B tem m elementos, o produto cartesiano terá nm elementos distintos
- Observe que se $A \neq B$ então $A \times B \neq B \times A$

Relações

Definição

Sejam A e B dois conjuntos. Dizemos que R é uma **relação** de A em B se $R \subset A \times B$, isto é, se R é um subconjunto do produto cartesiano de A por B.

- Se $|A| = n$ e $|B| = m$, existem 2^{nm} relações de A em B
- Se $(a, b) \in R$, dizemos que a se relaciona com b
- Observe que $(a, b) \in R$ não implica $(b, a) \in R$

Definição

Uma relação $f \subset A \times B$ é uma **função** de A em B (e escrevemos $f : A \to B$) se os dois critérios abaixo forem atendidos:

1. todo elemento a de A se relaciona com algum elemento b de B;
2. cada elemento a de A está relacionado com um único elemento b de B.

Injeção, sobrejeção e bijeção

Definição

Uma função $f : A \to B$ é dita **injetora** se $f(a) = f(b)$ implica em $a = b$, isto é, cada elemento do conjunto B está relacionado com um único elemento do conjunto A.

Uma função f é dita **sobrejetora** se, para qualquer elemento $b \in B$, existe um elemento $a \in A$ tal que $f(a) = b$, ou seja, cada elemento de B está relacionado a pelo menos um elemento de A.

Uma função que é, ao mesmo tempo, injetora e sobrejetora, é dita **bijetora**.

Função inversa

- A classificação de uma função como injetora ou sobrejetora está relacionada diretamente aos dois critérios da definição de funções
- Considere uma função $f : A \to B$ e seja $R \subset B \times A$ uma relação de B em A dada por

$$R = \{(b, a) \mid a \in A, b \in B, f(a) = b\}$$

- Se a relação R atender ao primeiro critério, então a função f é sobrejetora; se atender o segundo critério, f é injetora
- Se a relação R atende a ambos critérios, R é uma função, denominada função **inversa** da função f

Funções invertíveis

- Uma função f é invertível se ela for bijetora
- A função inversa de $f : A \to B$, se existir, é grafada como $f^{-1} : B \to A$
- Se for invertível, f estabelece uma relação um-a-um entre os elementos de A e B
- Se A e B forem conjuntos finitos, então ambos terão o mesmo número de elementos

Variáveis dependentes e independentes

- Na notação $y = f(x)$, x é a variável **independente** e y é a variável **dependente**
- Esta notação indica que y é função de x, ou que y depende de x
- Isto significa que, conhecido o valor de x, é possível determinar o valor de y
- Uma variável pode ser dependente de mais de uma variável
- Por exemplo, área A de um retângulo depende dos valores das medidas da base b e da altura h do retângulo, ou seja, $A = A(b, h)$

Zeros de funções

- Seja $f : A \to B$, onde 0 (zero) pertence a B
- Dizemos que $x \in A$ é um **zero** de f em A se $f(x) = 0$
- Uma função pode não ter, ter finitos ou ter infinitos zeros
- Exemplos:
  (a) a função $f(x) = 1/x$ não tem zeros nos reais
  (b) o Teorema Fundamental da Álgebra diz que todo polinômio de grau n tem n raízes complexas
  (c) a função $f(x) = \operatorname{sen}(x)$ tem infinitos zeros nos reais: qualquer múltiplo inteiro de 2π é zero de f

Método da bisseção

- Seja $f : \mathbb{R} \to \mathbb{R}$ é uma função contínua em um intervalo I dos reais, isto é, para qualquer elemento $a \in I$, o limite de $f(x)$ quanto x tende a a existe e é igual a $f(a)$
- Suponha que existam dois valores $a, b \in I$ tais que $f(a)f(b) < 0$, isto é, $f(a)$ e $f(b)$ tem sinais opostos
- Nestas condições, o Teorema de Valor Intermediário garante que existe ao menos um valor $c \in (a, b)$ tal que $f(c) = 0$
- O método da bisseção consiste em aproximar o valor de c por meio de uma busca binária

Implementação do método da bisseção em C++

```cpp
// Assuma que a função f(double) esteja definida, que a < b e que f(a)*f(b) < 0
// eps é a tolerância de erro
double root(double a, double b, double eps = 1e-6)
{
    while (fabs(a - b) > eps)
    {
        auto c = (a + b)/2;
        auto y = f(c);

        // c é uma boa aproximação para o zero
        if (fabs(y) < eps)
            return c;

        // Determina em qual dos intervalos ( (a,c) ou (c,b) ) está o zero
        f(a)*y < 0 ?  b = c : a = c;
    }

    return (a + b)/2;
}
```

Convergência

- Por conta de possíveis erros de precisão, o método da bisseção pode não convergir ou não melhorar sua precisão após um determinado número de iterações
- Implementações alternativas usam um número de passos N, pré-determinado, como critério de parada
- Há outros métodos com melhor convergência, como o método de Newton
- Porém o método da bisseção é notável por sua simplicidade e aplicabilidade

Variante da implementação do método da bisseção em C++

```cpp
// Assuma que a função f(double) esteja definida, a < b e que f(a)*f(b) < 0
// N é o número de iterações do algoritmo
double root(double a, double b, int N = 50)
{
    while (N--)
    {
        double c = (a + b)/2;

        // Determina em qual dos intervalos ( (a,c) ou (c,b) ) está o zero
        f(a)*f(c) < 0 ?  b = c : a = c;
    }

    return (a + b)/2;
}
```

Referências

1. **HALE**, Margie. *Essentials of Mathematics: Introduction to Theory, Proof, and the Professional Culture*. Mathematical Association of America, 2003.
2. **Wikipédia**. Bissection Method. Acesso em 15 de agosto de 2017.