DIFERENÇAS DIVIDIDAS - NEWTON

INSTITUTO POLITÉCNICO DE BRAGANÇA
Escola Superior de Comunicação, Administração e Turismo

Redney Monteiro, 46398, Informática e Comunicações, EsACT- IPB

Jumara Fernandes, 43691, Informática e Comunicações, EsACT- IPB

25 de maio de 2022

INTRODUÇÃO

Este trabalho foi desenvolvimento no âmbito da unidade curricular Cálculo II, por Redney Monteiro e Jumara Fernandes. Tem como objetivo apresentar o Método Newton - diferenças divididas, este método é utilizado para determinar um polinómio interpolador dados alguns objetos e suas imagens.

1. DESCRIÇÃO DO MÉTODO

Este serve é utilizado para determinar um polinómio interpolador para um dado conjunto de pontos. Os coeficientes são obtidos através do cálculo das diferenças divididas.

Suponha que P(x) seja o n-ésimo polinómio interpolador que coincide com uma função f nos pontos $x_0, x_1, \dots, x_n$. Onde o polinômio obtido é único (apenas para um dado grau, isto é, para cada grau de uma equação vai ter um polinômio diferente). As diferenças divididas de f em relação a $x_0, x_1, \dots, x_n$ são usadas para representar P(x) na forma:

$$P(x) = C_0 + C_1(x - x_0) + C_2(x - x_0)(x - x_1) + \cdots + C_n(x - x_0)(x - x_1)(x - x_{n-1})$$

Para determinar o valor de x_0, calculamos o $P(x_0)$, temos

$$C_0 = P(x_0) = f(x_0)$$

Da mesma maneira, se calcula o $P(x_1)$, temos

$$f(x_0) + C_1(x_1 - x_0) = P(x_1) = f(x_1)$$

Temos que o valor de x_1:

$$a_1 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

1.1 O ALGORITMO

Algoritmo das diferenças divididas.
PARA i = 1, …, n-1 FAÇA:
    PARA J = 1, …, n-1-i:
        $F_{i,j} \leftarrow \frac{F_{i,j-1} - F_{i-1,j-1}}{x_i - x_{i-j}}$
PARE
DEVOLVE cf

```python
#função que calcula as doferenças dividivas
def DividedDifferences(x, y, valid):
    delta = [item for item in y] #copia os valores das imagens para a lista delta
    cf = [] #cria a lista dos coeficiente vazia
    cf.append(y[0]) #adiciona a imagem do primeiro objeto (x1)
    n = len(y)
    # vai percorrer até o total do polimonio menos um
    for i in range(n - 1): #se sabemos quatro ponto, vamos obter um polimonio do 3°grau
        if(valid): #se for para mostrar
            print('-' * 20)
            print(f'\033[7;34;40mOrdem {i + 1}\033[m')
        for j in range(n - 1 - i): #em cada iteração (ordem), calcular as diferenças
            number = delta[j + 1] - delta[j] #calcula o numerador
            denom = x[j + 1 + i] - x[j] #calcula o denominador
            if (valid):
                print(f'({delta[j + 1]} - {delta[j]}) / ({x[j + 1 + i]} - {x[j]}) = {(number / denom)}')
            delta[j] = number / denom # calcula a diferença
        cf.append(delta[0]) #adiciona apenas o primeiro valor da ordem que é o um coeficiente do polimonio interpolador
    return cf #retorna os operadores
```

A diferença dividida de ordem 0:

$$f[x_i, x_{i+1}] = \frac{f[x_{i+1}] - f[x_i]}{x_{i+1} - x_i}$$

A diferença dividida de ordem 1:

$$f[x_i, x_{i+1}, x_{i+2}] = \frac{f[x_{i+1}, x_{i+2}] - f[x_i, x_{i+1}]}{x_{i+2} - x_i}$$

A diferencia dividida de ordem n:

$$f[x_i, x_{i+1}, \dots, x_{i+k-1}, x_{i+k}] = \frac{f[x_{i+1}, x_{i+2}, \dots, x_{i+k}] - f[x_i, x_{i+1}, \dots, x_{i+k-1}]}{x_{i+k} - x_i}$$

Vai decorrer até o n-ésima ordem:

$$f[x_0, x_1, \dots, x_n] = \frac{f[x_1, x_2, \dots, x_n] - f[x_0, x_1, \dots, x_{n-1}]}{x_n - x_0}$$

Comas diferenças divididas calculadas, vamos montar o polinômio interpolador

$$P(x) = f[x_0] + C_1(x - x_0) + C_2(x - x_0)(x - x_1) + a_n(x - x_0)(x - x_1) \dots (x - x_{n-1})$$

```python
#função que retorna a expressão do polimonio interpolador
def Equation(x, cf):
    n = len(x)
    equation = '' #começa a expressão vazia
    for i in range(n):
        #em cada valor de xn, adiciona na equação o sinal, seguido do sinal de multiplicação caso tiver um numero ainda colocar na expressão
        equation += f'{cf[i]:+}' + '*'.join([f"(x{-xj:+})" for j, xj, in enumerate(x) if j < i])
    return equation #retorna a expressão
```

Para o programa receber vários conjuntos de pontos, criamos uma função Menu, que pergunta ao utilizador o valor de x e f(x) e para cada iteração pergunta ao utilizador se pretende continuar ou não (é feita uma verificação, só aceita se digitar S ou N), e quando for digitado N, é perguntado ao utilizado se quer que apareça o resultado das diferenças divididas (feita uma verifica para aceitar apenas S ou N).

```python
def Menu(x, y):
    cont = True
    valid = True
    while cont:
        x.append(float(input('x = '))) #adiciona o valor de x, recebido na lista
        y.append(float(input('f(x) = '))) #adiciona o valor de y, recebido na lista
        while valid: #enquanto for valido
            resp = input('Quer continuar[S/N]? ').upper() #pergunta-mos e valida-mos
            if(resp[0] == 'N' or resp[0] == 'S'): # se for um opção valida
                valid = False #falso, para sair do ciclo(perguntar se quer continuar)
                if(resp[0] == 'N'): #se for não, sai do ciclo, o utilizador não vai digitar mais numeros
                    cont = False
            else: # se não for vália
                print('\033[31mErro!!! Por favor digite S/N\033[m')
        valid = True
    while valid:
        resp = input('Mostrar as diferenças dividivas[S/N]? ').upper() #perguntar se que mostar as diferenças divididas
        if (resp[0] == 'N' or resp[0] == 'S'): #se for uma opção válida
            if(resp[0] == 'N'): #se for não, retorna falso
                return False
            else: #se for sim, retorna verdadeiro
                return True
        else: #se não for
            print('\033[31mErro!!! Por favor digite S/N\033[m')
```

1.2 EXERCÍCIO DE TESTE

Dado esses pontos calcula o polinômio interpolador para cada conjunto de pontos:

Exercício 01

x	-1	0	2
F(x)	4	1	-1

Exercício 02

x	1	2	3	4
F(x)	2	1	6	47

Exercício 03

x	0	2	3	4
F(x)	7	11	28	63

Exercício 04

x	1	2	3	4
F(x)	2	5	1	3

Exercício 05

x	-1	0	2
F(x)	4	1	-1

2. RESOLUÇÃO DO EXERCÍCIO 01 PELO MÉTODO DIFERENÇAS DIVIDIDAS - NEWTON

Dados o conjunto de pontos, executamos o programa. Aparecerá um menu par informar o valor de x e para f(x) e para cada pergunta apareça uma mensagem a perguntar se quer continuar e no final aparecerá uma mensagem a perguntar se quer mostrar os passos da diferença divididas.

```
x = -1
f(x) = 4
Quer continuar[S/N]? s
x = 0
f(x) = 1
Quer continuar[S/N]? s
x = 2
f(x) = -1
Quer continuar[S/N]? n
Mostrar as diferenças dividivas[S/N]? s
--------------------
Ordem 1
(1.0 - 4.0) / (0.0 - -1.0) = -3.0
(-1.0 - 1.0) / (2.0 - 0.0) = -1.0
--------------------
Ordem 2
(-1.0 - -3.0) / (2.0 - -1.0) = 0.6666666666666666

p(x) = +4.0-3.0(x+1.0)+0.6666666666666666(x+1.0)*(x-0.0)

Process finished with exit code 0
```

3. RESOLUÇÃO DO EXERCÍCIO 02 PELO MÉTODO DIFERENÇAS DIVIDIDAS – NEWTON

```
x = 1
f(x) = 2
Quer continuar[S/N]? s
x = 2
f(x) = 1
Quer continuar[S/N]? s
x = 3
f(x) = 6
Quer continuar[S/N]? s
x = 4
f(x) = 47
Quer continuar[S/N]? n
Mostrar as diferenças dividivas[S/N]? s
--------------------
Ordem 1
(1.0 - 2.0) / (2.0 - 1.0) = -1.0
(6.0 - 1.0) / (3.0 - 2.0) = 5.0
(47.0 - 6.0) / (4.0 - 3.0) = 41.0
--------------------
Ordem 2
(5.0 - -1.0) / (3.0 - 1.0) = 3.0
(41.0 - 5.0) / (4.0 - 2.0) = 18.0
--------------------
Ordem 3
(18.0 - 3.0) / (4.0 - 1.0) = 5.0

p(x) = +2.0-1.0(x-1.0)+3.0(x-1.0)*(x-2.0)+5.0(x-1.0)*(x-2.0)*(x-3.0)

Process finished with exit code 0
```

4. RESOLUÇÃO DO EXERCÍCIO 03 PELO MÉTODO DIFERENÇAS DIVIDIDAS – NEWTON

```
x = 0
f(x) = 7
Quer continuar[S/N]? s
x = 2
f(x) = 11
Quer continuar[S/N]? s
x = 3
f(x) = 28
Quer continuar[S/N]? s
x = 4
f(x) = 63
Quer continuar[S/N]? n
Mostrar as diferenças dividivas[S/N]? n

p(x) = +7.0+2.0(x-0.0)+5.0(x-0.0)*(x-2.0)+1.0(x-0.0)*(x-2.0)*(x-3.0)

Process finished with exit code 0
```

5. RESOLUÇÃO DO EXERCÍCIO 04 PELO MÉTODO DIFERENÇAS DIVIDIDAS – NEWTON

```
x = 1
f(x) = 2
Quer continuar[S/N]? s
x = 2
f(x) = 5
Quer continuar[S/N]? s
x = 3
f(x) = 1
Quer continuar[S/N]? s
x = 4
f(x) = 3
Quer continuar[S/N]? n
Mostrar as diferenças dividivas[S/N]? n

p(x) = +2.0+3.0(x-1.0)-3.5(x-1.0)*(x-2.0)+2.1666666666666665(x-1.0)*(x-2.0)*(x-3.0)

Process finished with exit code 0
```

6. RESOLUÇÃO DO EXERCÍCIO 05 PELO MÉTODO DIFERENÇAS DIVIDIDAS – NEWTON

```
x = -1
f(x) = 4
Quer continuar[S/N]? s
x = 0
f(x) = 1
Quer continuar[S/N]? s
x = 2
f(x) = -1
Quer continuar[S/N]? n
Mostrar as diferenças dividivas[S/N]? s
--------------------
Ordem 1
(1.0 - 4.0) / (0.0 - -1.0) = -3.0
(-1.0 - 1.0) / (2.0 - 0.0) = -1.0
--------------------
Ordem 2
(-1.0 - -3.0) / (2.0 - -1.0) = 0.6666666666666666

p(x) = +4.0-3.0(x+1.0)+0.6666666666666666(x+1.0)*(x-0.0)

Process finished with exit code 0
```

CONCLUSÃO

Chegando ao fim deste trabalho com muito esforço. Foi rico em aprendizagemno metodo de diferencias divididas - newton, porém ao progredir no trabalho requeriu uma constante pesquisa de material e funções para satisfazer as dificuldades que iam aparecendo. No desenvolvimento do trabalho foi ganhando mais eficiência e como consequência, foi necessário buscar mais informações.

Foi muito muito satisfatório este trabalho, porque requeriu muita pesquisa

O código está todo explicado em forma de comentário.

Contudo, o básico do Metodo diferencias divididas ficou aprendido e aplicado neste trabalho.

REFERÊNCIAS

https://pt.wikipedia.org/wiki/Polin%C3%B3mio_de_Newton

https://sites.icmc.usp.br/andretta/ensino/aulas/sme0500-1-12/ipnewton.pdf

https://www.math.tecnico.ulisboa.pt/~calves/cursos/Interpola.HTM

https://www.youtube.com/watch?v=OxJwjpZxEkc

https://www.youtube.com/watch?v=pnf8UCJ2Gwg