Universidade de São Paulo

EP 2 - MAP3121

Decomposição LU para Matrizes Tridiagonais

Nome	NUSP
Diogo Vaccaro	8803195
Otávio Henrique Monteiro	10774159

5 de Junho de 2022

Conteúdo

Introdução

O segundo exercício programa proposto pela equipe de Métodos Numéricos e Aplicações diz respeito a implementação de fórmulas de integração numérica para cálculo de algumas integrais duplas.

Fórmulas de Gauss

A fórmula por integração numérica de Gauss de uma função f permite aproximar a integral de f por uma somatória de n pesos multiplicados por valores de f. Quando f for um polinômio de grau 2n-1 essa aproximação é exata!

Os nós e pesos são bem conhecidos para o intervalo [-1,1] e são encontrados em tabelas. Por exemplo, a fórmula

$$\int_{-1}^{1} f(x)\,dx \approx \frac{5}{9} f\left(-\sqrt{0.6}\right) + \frac{8}{9} f(0) + \frac{5}{9} f\left(\sqrt{0.6}\right)$$

é exata para a integral de -1 a 1 de polinômios de grau menor igual a 5 (pois tem 3 nós). Uma verificação desse resultado pode ser encontrada no anexo.

Podemos obter as fórmulas para qualquer intervalo [a,b] por meio de uma mudança de variável.

$$t \in [-1, 1]\; para\; x \in [a, b] \quad | \quad x = a + \frac{b-a}{2}(t+1) \Leftrightarrow t = \frac{2x - a - b}{b-a}$$

$$\int_{-1}^{1} f(t)\,dt = \int_{a}^{b} f(x(t)) \frac{2}{b-a} dx, \text{ sendo } dt = \frac{2}{b-a} dx.$$

Assim:

$$\int_{a}^{b} f(x)\,dx = \frac{b-a}{2} \int_{-1}^{1} f(t)\,dt.$$

Para 3 nós temos que:

$t_1 = -\sqrt{0.6}$, $t_2 = 0$, $t_3 = \sqrt{0.6}$ & $\omega_1 = 5/9$, $\omega_2 = 8/9$, $\omega_3 = 5/9$

Logo:

$x_1 = a + \frac{b-a}{2}(1 - \sqrt{0.6})$, $x_2 = \frac{a+b}{2}$, $x_3 = a + \frac{b-a}{2}(1 + \sqrt{0.6})$

Portanto, a fórmula de Gauss exata para polinômios de grau menor ou igual a 5 em um intervalo [a,b] é dada por:

$$\int_a^b f(x)dx \approx \frac{b-a}{2}[\frac{5}{9}f(a + \frac{b-a}{2}(1 - \sqrt{0.6}) + \frac{8}{9}f(\frac{a+b}{2}) + \frac{5}{9}f(a + \frac{b-a}{2}(1 + \sqrt{0.6})]$$

Analisando a seguinte igualdade:

$$\int_a^b (x - x_1)(x - x_2)\ldots(x - x_n)q(x)\,dx = 0$$

Podemos concluir que ela é verdadeira para qualquer polinômio q(x) de grau menor ou igual a n-1, pois nesse caso f(x) é igual a zero nos nós.

Integrais Duplas

O exercício programa se baseia, primariamente, no cálculo de integrais duplas em regiões R do plano. Utilizamos uma aproximação das integrais por meio da fórmula de Gauss para *n* nós.

$$I = \iint_R f(x, y)\,dx\,dy$$

Com R da forma: $\{(x, y) \mid a \leq x \leq b, c(x) \leq y \leq d(x)\}$

A equação aproximada para a integral dupla, calculada de maneira iterada, que utilizamos é:

$$I = \sum_{i=1}^{n} u_i \sum_{j=1}^{n} v_{ij} f(x_i, y_{ij})$$
,

onde onde xi e ui são os nós e os pesos no intervalo [a, b], e yij e vij são os nós e os pesos nos intervalos [c(xi), d(xi)].

Desenvolvimento

Este exercício programa foi desenvolvido em C++, seguindo as especificações detalhadas no enunciado fornecido. Foi criado um documento *LEIAME.txt* contendo uma explicação breve de compilação, interação, funcionamento e saídas do programa.

O programa interage com o usuário inicialmente pedindo o número da questão utilizada como exemplo. O programa realiza iterativamente o cálculo para os diferentes valores de n especificados (i.e. 6, 8 e 10) e imprime individualmente esses resultados.

As funções e equações de cada questão proposta são determinadas por meio da variável global “questao” que armazena o número da questão e, utilizando-se do método case-switch, determina a porção de código referente à execução dessa. Para as questões com mais de uma pergunta, é passado internamente um novo indicador para a iteração seguinte (do segundo item do exemplo).

A aproximação de integral dupla por meio da fórmula de Gauss foi realizada por meio da chamada de função com dois laços *for* aninhados, referentes às somatórias presentes na equação explicitada anteriormente, com o devido ajuste de intervalo de integração após laço.

Questão 1

O volume do cubo calculado, de arestas de comprimento 1, é igual a 1.0. Já o do tetraedro de vértices (0, 0, 0), (1, 0, 0), (0, 1, 0) e (0, 0, 1) é 0.16666666666666669.

Os resultados são exatos (exceto por aproximações) devido ao grau do polinômio e valores de *n* escolhidos. Isso se deve ao fato da aproximação pelas fórmulas de Gauss serem exatas para polinômios de grau k, quando $0 \leq k \leq 2n-1$.

```
Para o tetraedro:

Para n igual a 6
O resultado da Integracao e':0.16666666666666663
Para n igual a 8
O resultado da Integracao e':0.16666666666666666
Para n igual a 10
O resultado da Integracao e':0.16666666666666669
```

Questão 2

Para três valores de n usados o resultado para a área da região A foi muito próximo. Esse valor foi avaliado também com a mudança de d(x) de [1-x²] para [sqrt(1-y)], com os resultados sendo muito próximos ao valor exato de ⅔ (~0.6666666666666667), especialmente os do primeiro intervalo. A seguir encontram-se os valores encontrados para os diferentes *n*, primeiro para o intervalo de [0] a [1-x²] e logo abaixo o intervalo de [0] a [sqrt(1-y)].

```
Para n igual a 6
O resultado da Area da regiao calculado por dydx e':0.66666666666666674
O resultado da Area da regiao calculado por dxdy e':0.66704643791561358
Para n igual a 8
O resultado da Area da regiao calculado por dydx e':0.66666666666666663
O resultado da Area da regiao calculado por dxdy e':0.66683558010017663
Para n igual a 10
O resultado da Area da regiao calculado por dydx e':0.66666666666666674
O resultado da Area da regiao calculado por dxdy e':0.66675604293650881
```

Questão 3

Calculados externamente as equações referentes às derivadas parciais da função, podemos então calcular o valor da área da superfície e seu volume. Os valores diferem bem ligeiramente nas últimas casas de precisão de acordo com o valor de n, assim como em outros exemplos.

d/dx(e^(y/x)) = -(y e^(y/x))/x^2 ,   d/dy(e^(y/x)) = e^(y/x)/x

```
Para n igual a 6
O resultado da Area da Superficie e':0.1054978824004979
O resultado do Volume da Regiao e':0.033305566116237188
Para n igual a 8
O resultado da Area da Superficie e':0.10549788240051997
O resultado do Volume da Regiao e':0.033305566116232081
Para n igual a 10
O resultado da Area da Superficie e':0.10549788240051994
O resultado do Volume da Regiao e':0.033305566116232081
```

Questão 4

Após estudados os casos, chegamos aos valores por meio da integração e por outras ferramentas nos seguintes resultados:

$$V_{revolucao} = \int_{-1}^{1} \int_{0}^{e^{-y^2}} x \cdot dxdy$$

$$V_{calota} = \int_{3/4}^{1} \int_{0}^{\sqrt{1-y^2}} x \; dxdy$$

```
Para n igual a 6
O resultado do Volume do Solido de Revolucao e':3.7581650328967093
O resultado do Volume da calota esferica de altura 1/4 e':0.17998707911191519
Para n igual a 8
O resultado do Volume do Solido de Revolucao e':3.7582492624394384
O resultado do Volume da calota esferica de altura 1/4 e':0.17998707911191522
Para n igual a 10
O resultado do Volume do Solido de Revolucao e':3.7582496332093873
O resultado do Volume da calota esferica de altura 1/4 e':0.17998707911191525
```

Conclusão

Podemos concluir que o método de integração de Gauss constitui um importante, simples e elegante e preciso método de implementação algorítmica numérica de resolução de integrais simples e dupla. Ademais, notamos como o número de nós pode influenciar o resultado final da integração, apesar da diferença ser pouco expressiva para os valores utilizados. Com esse Exercício Computacional verificamos o funcionamento e a eficácia do método para diferentes questões de integração, em especial integrais duplas.

Referências

[1] Equipe de Métodos Numéricos. Fórmulas de Integração Numérica de Gauss - MAP3121, 2022.

[2] WOLFRAM. Wolfram|Alpha: Computational Intelligence. Disponível em: https://www.wolframalpha.com/. Acesso em: 01 junho 2022.

Anexos

Demonstração da (1):

$$\underset{(1)}{\int_{-1}^{1} f(x)\,dx} \approx \underset{(2)}{A_1 f(x_1)} + \underset{(3)}{A_2 f(x_2)} + \underset{(4)}{A_3 f(x_3)}$$

→ 6 incógnitas
→ 6 coeficientes

Seja $f(x) \approx c_0 + c_1x + c_2x^2 + c_3x^3 + c_4x^4 + c_5x^5$

substituindo:

$$(1) = \int_{-1}^{1} c_0 + c_1x + c_2x^2 + c_3x^3 + c_4x^4 + c_5x^5\,dx = c_0 \cdot x + c_1\frac{x^2}{2} + \frac{c_2 \cdot x^3}{3} + \frac{c_3 \cdot x^4}{4} + \frac{c_4 \cdot x^5}{5} + \frac{c_5 \cdot x^6}{6}\Big|_{-1}^{1}$$

$$= 2c_0 + \frac{2c_2}{3} + \frac{2c_4}{5}$$

$$(2) = A_1 \cdot (c_0 + c_1x_1 + c_2 \cdot x_1^2 + c_3 x_1^3 + c_4 x_1^4 + c_5 x_1^5)$$

$$(3) = A_2(c_0 + c_1 x_2 + c_2 x_2^2 \ldots + c_5 x_2^5)$$

$$(4) = A_3(c_0 + c_1 x_3 + \ldots + c_5 \cdot x_3^5)$$

$$\left.\begin{aligned} &A_1 + A_2 + A_3 = 2 \\ &A_1x_1 + A_2x_2 + A_3x_3 = 0 \\ &A_1x_1^2 + A_2 \cdot x_2^2 + A_3x_3^2 = 2/3 \\ &A_1x_1^3 + A_2x_2^3 + A_3x_3^3 = 0 \\ &A_1x_1^4 + A_2x_2^4 + A_3 \cdot x_3^4 = 2/5 \\ &A_1x_1^5 + A_2x_2^5 + A_3x_3^5 = 0 \end{aligned}\right.$$

solução:

$x_1, x_2, x_3 = [-\sqrt{0{,}6}, 0, \sqrt{0{,}6}]$

$A_1, A_2, A_3 = [5/9, 8/9, 5/9]$

Código do programa:

```
//EP2 MAP3121
//Diogo Vaccaro 8803195
//Otavio Henrique Monteiro 10774159

#include <stdio.h>
#include <iostream>
#include <cmath>
#include <iomanip>

#define MAX 11
const double PI = 3.141592653589793238463;
char questao = '2'; //Variavel universal para escolha de questao

using namespace std;

double calcula_integral(int n, double a, double b, double T[MAX],double W[MAX]);
double integral_dupla(int n, double a, double b, double T[MAX],double W[MAX]);
double funcao_escolhida(double x, double y);
double c_escolhido(double xi);
double d_escolhido(double xi);

int main(){
    // INICIALIZAR VARIAVEIS
    // Nmr de nos
    int n = 6;
    int m;
    int i;
    int N;
    //double a,b;
    cout.precision(17);

    cout<<"Bem-Vindo ao EP2 de MAP3121-2022 \n";
```

```
    cout<<"A seguir, digite de 1 a 4 a questao do enunciado a qual deseja o calculo das integrais duplas \n";
    cout<<"Numero da questao : \n";
    cin>>questao;


    m = 2*n-1; // maximo grau do polinomio
    double T[MAX]; //vetor de abscissas
    double W[MAX]; //vetor de pesos

    //DADOS DE ABCISSAS E PESOS, ja prontos para uso
    double x6[] = {0.0, -0.9324695142031520278123016, -0.6612093864662645136613996, -0.2386191860831969086305017,
                        0.2386191860831969086305017, 0.6612093864662645136613996, 0.9324695142031520278123016};
    double w6[] = {0.0, 0.1713244923791703450402961, 0.3607615730481386075698335, 0.4679139345726910473898703,
                        0.4679139345726910473898703, 0.3607615730481386075698335, 0.1713244923791703450402961};
    double x8[] = {0.0, -0.9602898564975362316835609, -0.7966664774136267395915539, -0.5255324099163289858177390, -0.1834346424956498049394761,
                        0.1834346424956498049394761, 0.5255324099163289858177390, 0.7966664774136267395915539, 0.9602898564975362316835609};
    double w8[] = {0.0, 0.1012285362903762591525314, 0.2223810344533744705443560, 0.3137066458778872873379622, 0.3626837833783619829651504,
                        0.3626837833783619829651504, 0.3137066458778872873379622, 0.2223810344533744705443560, 0.1012285362903762591525314};
    double x10[] = {0.0, -0.9739065285171717200779640, -0.8650633666889845107320967, -0.6794095682990244062343274, -0.4333953941292471907992659, -0.1488743389816312108848260,
                         0.1488743389816312108848260, 0.4333953941292471907992659, 0.6794095682990244062343274, 0.8650633666889845107320967, 0.9739065285171717200779640};
    double w10[] = {0.0, 0.0666713443086881375935688, 0.1494513491505805931457763, 0.2190863625159820439955349, 0.2692667193099963550912269, 0.2955242247147528701738930,
                         0.2955242247147528701738930, 0.2692667193099963550912269, 0.2190863625159820439955349, 0.1494513491505805931457763, 0.0666713443086881375935688};

    //ITERACAO PARA DIFERENTES VALORES DE N
    while(n<12){
        // Preencher T e W com os valores das abscissas e dos pesos para o n determinado (preencher do 1 ate n)
        switch (n)
        {
        case 2: //Extra   - testes
            T[1]=-sqrt(3)/3;
            T[2]=-T[1];
            W[1]=1;
            W[2]=W[1];
            break;
        case 6:
            for (size_t i = 0; i <= n; i++){
                T[i] = x6[i];
                W[i] = w6[i];
            }break;
        case 8:
            for (size_t i = 0; i <= n; i++){
                T[i] = x8[i];
                W[i] = w8[i];
            }break;
        case 10:
            for (size_t i = 0; i <= n; i++){
                T[i] = x10[i];
                W[i] = w10[i];
            }break;
        default:
            cout << "Valor de n inv�lido";
            char end; cin >> end;
            return 0;
        }
```

```
        // SOLUCAO INDIVIDUAL PARA CADA EXEMPLO
        double resultado = 0.0;
        switch (questao)
        {
        case '1':
            resultado = integral_dupla(n,0,1,T,W); // Cubo
            cout << "Para n igual a " << n <<endl;
            cout << "O resultado do Volume do Cubo e':" << resultado << endl;

            questao = '5'; // tetraedro
            resultado = integral_dupla(n,0,1,T,W); //Tetraedro
            // Imprimir resposta
            cout << "O resultado do Volume do Tetraedro e':" << resultado << endl;
            questao = '1';
            break;
        case '2':
            resultado = integral_dupla(n,0,1,T,W);//area dydx
            // Imprimir resposta
            cout << "Para n igual a " << n <<endl;
            cout << "O resultado da Area da regiao calculado por dydx e':" << resultado << endl;

            questao = '6'; // area dxdy
            resultado = integral_dupla(n,0,1,T,W);
            // Imprimir resposta
            cout << "O resultado da Area da regiao calculado por dxdy e':" << resultado << endl;
            questao = '2';
            break;
        case '3':
            resultado = integral_dupla(n,0.1,0.5,T,W);//Area
            // Imprimir resposta
            cout << "Para n igual a " << n <<endl;
            cout << "O resultado da Area da Superficie e':" << resultado << endl;

            questao = '7';
            resultado = integral_dupla(n,0.1,0.5,T,W);//Volume
            // Imprimir resposta
            cout << "O resultado do Volume da Regiao e':" << resultado << endl;
            questao = '3';
            break;
        case '4':
            resultado = 2*PI*integral_dupla(n,-1,1,T,W);//Solido Revolucao
            // Imprimir resposta
            cout << "Para n igual a " << n <<endl;
            cout << "O resultado do Volume do Solido de Revolucao e':" << resultado << endl;

            questao = '8';
            resultado = 2*PI*integral_dupla(n,0.75,1,T,W);//Calota
            // Imprimir resposta
            cout << "O resultado do Volume da calota esferica de altura 1/4 e':" << resultado <<
endl;
            questao = '4';
            break;

        default:
            return 1;
        }


        //Proximo valor de n, se aplicavel
        n+=2;
    }


    //Resultado exato


    //Finalizar
    cout << "\n\nDigite algum caractere para finalizar.\n";
    char end;
```

```
    cin >> end;
    return 1;
}

//INTEGRAL SIMPLES
double calcula_integral(int n, double a, double b, double T[MAX],double W[MAX]){
    double integral=0; // valor da integral
    double ba2 = (b-a)/2; // valor medio do intervalo ab
    double x,y;
    for (int i=1;i<=n;i++){ // somatorio peso*funcao(abscissa)
        x = a + ba2*(T[i]+1);
        y = funcao_escolhida(x,0);
        integral += y*W[i];
    }
    integral = integral * ba2; // valor final
    return integral;
}

//INTEGRAL DUPLA (intervalo interno definido a parte)
double integral_dupla(int n, double a, double b, double T[MAX],double W[MAX]){
    double ba2 = (b-a)/2; // valor medio do intervalo ab
    double xi, yij, f;
    double sum_i = 0;

    for (int i=1;i<=n;i++){ // somatorio peso*funcao(abscissa)
        xi = a + ba2*(T[i]+1);

        double sum_j = 0;
        double dc2 = (d_escolhido(xi)-c_escolhido(xi))/2; // valor medio do intervalo cd
        for (int j =1;j<=n;j++)
        {
            yij = c_escolhido(xi) + dc2*(T[j]+1);
            f = funcao_escolhida(xi, yij);
            sum_j += W[j] * f ;
        }
        sum_i += W[i] * sum_j * dc2 ;
    }
    return sum_i * ba2 ;
}


//FUNCAO A SER INTEGRADA PARA CADA QUESTAO
double funcao_escolhida(double x, double y){//Funcao usada na integracao, pode ser escolhida de acordo com a questao
    //return pow(x,3)+1;// f(x)=x^3+1
    switch (questao)
    {
    case '1': // cubo
        return 1;
    case '5': //tetraedro
        return 1-x-y; //z=f(x,y)=1-x-y
    case '2':
        return 1;
    case '6':
        return 1;
    case '3': // area
        return sqrt(pow((-(y*exp(y/x))/(x*x)),2) + pow(exp(y/x)/x,2) + 1);
    case '7': // volume
        return exp(y/x);
    case '4':
        return y; //x (convencao invertida nesse item) - solido de revolucao
    case '8':
        return y;
    default:
        return 1;
    }
}

//VALOR DE c(x) DO INTERVALO DE INTEGRACAO
double c_escolhido(double xi){
```

```
    switch (questao)
    {
    case '1': // cubo
        return 0;
    case '5': // tetraedro
        return 0;
    case '2':
        return 0; //Ok
    case '6':
        return 0;
    case '3':
        return pow(xi, 3); //Ok
    case '7':
        return pow(xi,3);
    case '4':
        return 0;
    case '8':
        return 0;
    default:
        return 0;
    }
}

//VALOR DE d(x) DO INTERVALO DE INTEGRACAO
double d_escolhido(double xi){
    switch (questao)
    {
    case '1': // cubo
        return 1;
    case '5': // tetraedro
        return 1-xi; // 0<y<1
    case '2':
        return 1 - xi*xi; //Ok
    case '6':
        return sqrt (1-xi);
        //sqrt(1-xi)=sqrt(1-y)
    case '3':
        return pow(xi, 2); //Ok
    case '7':
        return pow(xi,2);
    case '4':
        return exp(-xi*xi);//e^-y^2
    case '8':
        return sqrt(1-xi*xi);
    default:
        return 0;
    }
}
```