Computação Científica I
Projeto 3: Integração usando Álgebra Linear

Professores: Bernardo Costa & Pedro Fonini

Monitores: Vitor Luiz & Luan Lima

22 de outubro de 2018

```
gauss quadrature weights and kronron quadrature abscissae and weights
as evaluated with 80 decimal digit arithmetic by l. w. fullerton,
bell labs, nov. 1981.
```

comment on `http://www.netlib.org/quadpack/gk15.f`

Calendário de entrega:

1. Teste 7: Interpolação. 20 de Outubro de 2020.
2. Projeto 3: Integração: 3 de Novembro de 2020.

1 Notações de Interpolação polinomial

- $\mathcal{F}$ é o conjunto de todas as funções contínuas;
- Pol_d, o conjunto dos polinômios de grau menor ou igual a d;
- Um polinômio $g \in \mathrm{Pol}_d$ se escreve $g(z) = \sum g_j z^j$ para coeficientes g_j.
- X é um conjunto de "nós de interpolação" fixos: $X = \{x_i\}$.
- L_X a aplicação linear de $\mathcal{F}$ em Pol_{n-1}, que dá o polinômio interpolador.
- π_X a aplicação linear de $\mathcal{F}$ em $\mathbf{R}^n$ que a cada função f dá o vetor $(f(x_i))$.
- $\phi_X : \mathbf{R}^n \to \mathrm{Pol}_{n-1}$ a aplicação inversa de π_X restrita aos polinômios de grau menor do que n.

2 Integração numérica como aplicação linear

Supomos nesta parte que $X \subset [-1, 1]$, o intervalo-padrão para integrais.

1. Mostre que a integral definida $I(g) = \int_{-1}^{1} g(t)\,dt$ é uma aplicação linear no espaço $\mathcal{F}$, logo em Pol_{n-1}.

2. Mostre que, para $g \in \mathrm{Pol}_{n-1}$ com coeficientes (g_j), $I(g) = \sum_{j=0}^{n-1} g_j \dfrac{1 - (-1)^{1+j}}{1 + j}$.

3. Considere a aplicação $I_X : \mathcal{F} \to \mathbf{R} : f \mapsto \displaystyle\int_{-1}^{1} \Big(L_X(f)\Big)(t)\,dt$, que calcula a integral do polinômio interpolador de f (dadas as abscissas $x_i \in X$) no intervalo $[-1, 1]$. Escreva I_X como a composta de duas aplicações lineares.

4. Use as decomposições de I_X e L_X para mostrar que $I_X(f)$ só depende de $\pi_X(f)$.

5. Deduza que existem reais w_i tais que $I_X(f) = \sum w_i f(x_i)$.

6. Queremos calcular w_i, para descobrir uma nova regra de integração. Escreva a equação matricial que determina g_j a partir dos x_i e dos $f(x_i)$.

7. Escreva a integral $I(g)$ como o produto interno do vetor de coeficientes g_j com um vetor v. Ou seja, encontre v tal que $I(g) = \langle (g_j), v\rangle$.

8. O vetor $g = (g_j)$ pode ser escrito como a solução de um sistema linear da forma $Mg = \pi_X(f)$. Demonstre que $I(g) = \Big\langle \pi_X(f), (M^{-1})^T v \Big\rangle$.

9. Deduza que $w = (M^{-1})^T v$.

2.1 Aplicações computacionais, mudanças de variáveis e erros numéricos

Vamos agora transformar estas equações em código!

1. Escreva uma função `matrix_nodes(xs)` que cria a matriz M^T correspondente aos pontos X.

2. Escreva uma função `weight_nodes(xs)` que retorna o vetor de pesos w para integrar no intervalo $[-1, 1]$. Verifique que w corresponde aos pesos certos para os métodos do ponto médio, trapézio e Simpson.

3. Escreva uma função `rule_nodes(xs)` que retorna uma função `rule(f,ai,bi)` para integrar f no intervalo $[a_i, b_i]$. Faça as contas primeiro no intervalo-padrão $[-1, 1]$, e depois "normalize" de volta para o intervalo $[a_i, b_i]$.

4. Escolha algumas funções e faça o gráfico do erro de integração em função do número de nós de interpolação escolhidos. (Integre pelo menos três funções "bem diferentes" para não analisar apenas um caso particular!)

5. Para as mesmas funções, aumente e/ou reduza o tamanho do intervalo de integração: será que o erro de integração continua com o mesmo comportamento?

6. Para integrar funções em intervalos muito grandes, pode ser melhor dividir em blocos do que "só" aumentar o grau do polinômio interpolador. Escreva uma função `integra(f,a,b,n,rule)` que aplica a regra `rule` em n sub-intervalos de $[a, b]$, para aproximar a integral.

7. Faça o gráfico do erro de integração em função do número n de subdivisões $[a_i, b_i]$ do intervalo $[a, b]$, para regras que usam, em cada sub-intervalo, 10, 15, 20 e 30 pontos.

8. Refaça os mesmos gráficos, agora com o erro em função do *número total de pontos* do intervalo $[a, b]$ onde foi avaliada a função f. Até quando aumentar o número de pontos em X acelera a convergência? Como você explica este equilíbrio entre grau do interpolador e número de sub-intervalos?

   Sabemos que usar pontos de Chebyshev é mais interessante para interpolação, em comparação com pontos igualmente espaçados.

9. Refaça a análise dos dois métodos de integração, agora para nós de Chebyshev: primeiro, apenas aumentando o grau do polinômio interpolador, e, em seguida, também aumentando o número de sub-divisões. Qual a melhor combinação de grau/sub-divisões que você encontra para cada função?

10. Compare com o uso de interpoladores em pontos igualmente espaçados.