Máster en Materiales y Sistemas Sensores para Tecnologías Medioambientales (Erasmus Mundus)

NOTAS DE CÁLCULO NUMÉRICO

Damián Ginestar Peiró

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DEL DISEÑO

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Capítulo 1

Integración numérica

La integración numérica es una operación frecuente en computación científica. Obtener la primitiva de una función puede ser complicado, incluso imposible. De hecho muchas funciones se definen a partir de integrales que no pueden calcularse de manera exacta. Por otro lado, cuando únicamente conocemos el valor de la función en un conjunto de puntos $(x_i, f(x_i))$, como ocurre con los resultados de un experimento o de simulaciones numéricas, sus integrales sólo se pueden obtener numéricamente, lo cual motiva aún más la necesidad de poder obtener derivadas e integrales a partir de conjuntos discretos de datos.

Discutiremos distintos métodos aproximados para calcular la integral definida de una función $f(x)$ en un intervalo $[a, b]$,

$$I = \int_a^b f(x)\,dx\ .$$

La función $f(x)$ puede estar definida analíticamente o mediante un conjunto de puntos $(x_0, f_0), \ldots, (x_N, f_N)$. Las fórmulas de inetegración numérica o *fórmulas de cuadratura* se expresan de la forma

$$I \approx \sum_{j=0}^{N} w_j f(x_j)\ ,$$

donde a los coeficiente w_j se les llama pesos. Distintos conjuntos de pesos dan lugar a distintas fórmulas de cuadratura.

1.1. Fórmulas de Newton-Cotes

Supongamos que se divide el intervalo de integración en N subintervalos

$$x_i = a + ih \ , \quad h = \frac{b-a}{N} \ , \quad i = 0, 1, \ldots , N \ .$$

Se construye el polinomio interpolador que pasa por los $N+1$ puntos (x_j, f_j)

$$P_N(x) = \sum_{j=0}^{N} P_{N,i}(x) f_j \ .$$

Para el cálculo de la integral definida se hace

$$\begin{aligned} I &= \int_a^b f(x)\, dx \approx \int_a^b P_N(x)\, dx \\ &= \sum_{j=0}^{N} f_j \int_a^b P_{N,i}(x)\, dx = (b-a) \sum_{j=0}^{N} C_j^N f_j \ , \end{aligned}$$

donde

$$C_j^N = \frac{1}{b-a} \int_a^b P_{N,j}(x)\, dx \ ,$$

son los números de Cotes, y los pesos

$$w_j = (b-a) C_j^N \ .$$

Por ejemplo, si utilizamos calculamos el polinomio interpoldor que pasa por (x_0, f_0), $(N = 0)$, se obtiene la *Regla del rectángulo*

$$I \approx (b-a) f_0 \ .$$

Si utilizamos calculamos el polinomio interpolador que pasa por $\left(\frac{x_0+x_1}{2}, f\left(\frac{x_0+x_1}{2}\right)\right)$, $(N = 0)$, se obtiene la *Regla del punto medio*

$$I \approx (x_1 - x_0) f\left(\frac{x_0 + x_1}{2}\right) \ .$$

Si se utiliza el polinomio interpolador que pasa por (x_0, f_0) y (x_1, f_1), $(N = 1)$, se obtiene la *Regla del trapecio*

$$I \approx \frac{1}{2}(x_1 - x_0)\left(f_0 + f_1\right) \ .$$

Si se divide el intervalo $[a, b]$ en N subintervalos, escribe

$$I = \sum_{i=0}^{N-1} I_i = \sum_{i=0}^{N-1} \int_{x_i}^{x_{i+1}} f(x)\, dx \ ,$$

usando la Regla del trapecio, se tiene

$$I \approx h \left(\frac{1}{2} f_0 + \frac{1}{2} f_N + \sum_{j=1}^{N-1} f_i \right) \ .$$

La siguiente función de Matlab,implementa la Regla del Trapecio para el cálculo de una integral

```
function I = trapecio(fun,x)
%
% I=trapecio(fun,x)
%
% Calcula la integral de fun sobre la malla definida por x
%
% Variables de entrada:
% fun: nombre de la funcion
% x: Malla sobre la que se realiza el calculo
%
% Variable de salida:
% I: valor de la integral
%
h=x(2)-x(1);
M = length(x);
f=feval(fun,x);
I = (2*sum(f)-f(1)-f(M))*h/2;
```

Si se utiliza el polinomio que pasa por (x_0, f_0), (x_1, f_1), y (x_2, f_2), $(N = 2)$ se obtiene la *Regla de Simpson*

$$I \approx \frac{1}{6} (x_2 - x_0) (f_0 + 4 f_1 + f_2) \ .$$

Para N subintervalos, se tiene la Regla de Simpson

$$I \approx \frac{h}{3} \left(f_0 + f_N + 4 \sum_{\substack{j=1 \\ j=\text{impar}}}^{N-1} f_j + 2 \sum_{\substack{j=2 \\ j=\text{par}}}^{N-2} f_j \right) \ .$$

Para utilizar la Regla de Simpson, el número de puntos de la partición, $N+1$, debe ser impar.

La siguiente función de Matlab implementa la Regla de Simpson,

```
function I = simpson(fun,x)
%
% I=simpson(fun,x)
%
% Calcula la integral de fun sobre la malla definida por x
%
% Variables de entrada:
% fun: nombre de la funcion
% x: Malla sobre la que se realiza el calculo
%
% Variable de salida:
% I: valor de la integral
%
h=x(2)-x(1); M = length(x);
f=feval(fun,x);
I = (2*sum(f(2:2:length(f)-1))+2*sum(f)-f(1)-f(M))*h/3;
```

1.2. Integración adaptativa

Tal y como hemos visto anteriormente, la forma más sencilla de evaluar numéricamente una integral es dividir el intervalo de integración en pasos equiespaciados de anchura h, evaluar la función en esos puntos y construir una suma de coeficientes por valores de la función en los puntos para aproximar el valor de la integral. La cuadratura adaptativa involucra la selección cuidadosa de los puntos donde la función va a ser evaluada, de manera que podamos calcular la integral con una precisión especificada realizando el mínimo número posible de evaluaciones de la función.

Si c es cualquier punto entre a y b, entonces se cumple

$$\int_a^b f(x)\,dx = \int_a^c f(x)\,dx + \int_c^b f(x)\,dx \ .$$

La idea es que si podemos aproximar cada uno de los integrandos de la parte derecha con una precisión dentro de una tolerancia especificada, la suma de ambos dará entonces el resultado deseado. Si no, podemos aplicar recursivamente la propiedad aditiva a cada uno de los intervalos $[a, c]$ y $[c, b]$. De

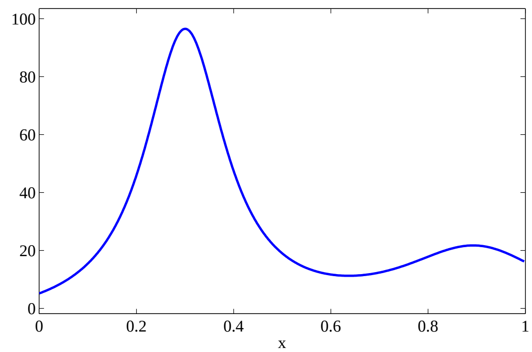

Figura 1.1: Función humps

este modo, el algoritmo resultante se adapta automáticamente al integrando, partiendo el intervalo en subintervalos con un espaciado fino en las partes donde el integrando varía rápidamente y con espaciados mayores donde el integrando varía lentamente

1.2.1. Reglas de cuadratura adaptativa en Matlab

La función de Matlab `quad` utiliza una regla de Simpson junto con una estrategia de adaptación recursiva como la descrita anteriormente. En cambio, la función `quadl` usa una cuadratura adaptativa basada en métodos de orden superior que la regla de Simpson, por lo que a menudo `quadl` requiere un menor número de evaluaciones de la función que `quad` para obtener una misma precisión.

Consideramos la función `humps`, que está definida por

$$h(x) = \frac{1}{(x-0{,}3)^2 + 0{,}01} + \frac{1}{(x-0{,}9)^2 + 0{,}04} - 6 \ ,$$

cuya gráfica en el intervalo $[0, 1]$ se presenta en la Figura 1.1.

La integral

$$I = \int_0^1 h(x)\, dx = 5 \arctan\left(\frac{16}{13}\right) - 6 + 10\pi \approx 29{,}85832539549867$$

Podemos escribir el siguiente programa de Matlab

```
for k = 1:12
```

```
  tol = 10^(-k);
  [Q,fcount] = quadl(@humps,0,1,tol);
  err = Q - Qexact;
  fprintf('%8.0e %21.14f %7d %13.3e \n',tol,Q,fcount,err)
end
```

Con lo que se obtiene el resultado:

```
tol        Q              fcount err
1e-01 29.85803034289291 48    -2.951e-04
1e-02 29.85832829257638 78     2.897e-06
1e-03 29.85832548167252 108    8.617e-08
1e-04 29.85832548167252 108    8.617e-08
1e-05 29.85832539547769 168   -2.099e-11
1e-06 29.85832539568428 198    1.856e-10
1e-07 29.85832539550114 348    2.466e-12
1e-08 29.85832539549867 438   -3.553e-15
1e-09 29.85832539549866 618   -1.066e-14
1e-10 29.85832539549867 768   -7.105e-15
1e-11 29.85832539549867 1008  -7.105e-15
1e-12 29.85832539549867 1608   0.000e+00
```

Se pueden usar también métodos de paso fijo. Utilizando las siguientes instrucciones se obtiene la integral mediante la Regla del Trapecio dividiendo el intervalo en subintervalos de longitud 0.001.

```
x = 0:0.001:1;
f = humps(x);
It = trapz(x,f)
err=
```

1.3. Integración de Gauss

Para obtener las fórmulas de cuadratura de Gauss, se utilizan nodos o puntos de evaluación de la función, $x_1, x_2, \ldots, x_n$, que no están igualmente espaciados, y unos coeficientes $c_1, c_2, \ldots, c_n$, de forma que se minimice el error al hacer la aproximción

$$\int_a^b f(x)\,dx \approx \sum_{i=1}^{n} c_i f\left(x_i\right) \ . \tag{1.1}$$

Los coeficientes $c_1, c_2, \ldots, c_n$, son, en principio arbitrarios y la única restricción sobre los nodos es que deben estar en $[a, b]$. Esto supone que para utilizar la aproximación del tipo dado por (1.1) hay que determinar $2n$ parámetros. Para calcular estos parámetros supondremos que la aproximación (1.1) es exacta cuando se considera como función $f(x)$ un polinomio de grado $2n - 1$.

Veamos como calcular los parámetros cuando $n = 2$ y el inetervalo de integración es el $[-1, 1]$. O sea, se han de calcular c_1, c_2, x_1 y x_2 de forma que

$$\int_{-1}^{1} f(x)\, dx \approx c_1 x_1 + c_2 x_2 \ , \tag{1.2}$$

es exacta para polinomios de grado 3. Se tiene que cumplir que la relación sea exacta para $f(x) = 1, x, x^2, x^3$. Se obtienen las igualdades

$$\begin{aligned} c_1 + c_2 &= \int_{-1}^{1} dx = 2 \ , \\ c_1 x_1 + c_2 x_2 &= \int_{-1}^{1} x\, dx = 0 \ , \\ c_1 x_1^2 + c_2 x_2^2 &= \int_{-1}^{1} x^2\, dx = \frac{2}{3} \ , \\ c_1 x_1^3 + c_2 x_2^3 &= \int_{-1}^{1} x^3\, dx = 0 \ . \end{aligned}$$

Resolviendo el sistema se llega a

$$c_1 = 1 \ , \quad c_2 = 1 \ , \quad x_1 = -\frac{\sqrt{3}}{3} \ , x_2 = \frac{\sqrt{3}}{3} \ .$$

Se puede hacer razonamientos similares para n=3,4, ..., pero el sistema de ecuaciones a resolver se hace muy complicado. Por ello, se suele utilizar otra formulación basada en polinomios ortogonales.

Consideramos el intervalo de integración $[-1, 1]$, lo cual no es una pérdida de generalidad ya que la relción

$$t = \frac{2x - a - b}{b - a} \ ,$$

pasa la variable x del intervalo $[a, b]$ a la variable t en $[-1, 1]$. Para el cálculo de integrales de la forma

$$I = \int_{-1}^{1} f(x)\, dx \ ,$$

se pueden usar los nodos y los coeficientes que se presentan en la Tabla 1.1

Tabla 1.1: Nodos y coeficientes para distintos valores de n.

Valor de n	Nodos (x_i)	Coeficientes (c_i)
2	0.577350269	1.0000000000
2	-0.577350269	1.0000000000
3	0.7745966692	0.5555555556
	0.0000000000	0.8888888889
	-0.7745966692	0.5555555556
4	0.8611363116	0.3478548451
	0.3399810436	0.6521451549
	-0.3399810436	0.6521451549
	-0.8611363116	0.3478548451
5	0.9061798459	0.2369268850
	0.5384693101	0.4786286705
	0.0000000000	0.0000000000
	-0.5384693101	0.4786286705
	- 0.9061798459	0.2369268850

1.4. Ejercicios

1. Se desea calcular la integral
$$\int_{-2}^{2} x^2 e^{-x^2}\, dx$$
con error menor de 10^{-3} mediante la fórmula de los trapecios compuesta. Indicar cúal es el número mínimo de divisiones del intervalo [-2, 2] que nos garantiza este resultado.

2. Calcular las integrales de las funciones $f(x) = \mathrm{sen}(4x)$ en $[0, \pi]$ y $f(x) = \frac{e^{-x}}{1+e^x}$ en $[-2, 2]$ para distintas particiones del intervalo mediante la regla de Simpson compuesta. Comparar los resultados con los valores exactos.

3. Obtener las fórmulas de Newton-Cotes para $N = 2$ y $N = 3$.

4. Obtener una fórmula de integración para el caso de tres puntos no equiespaciados x_0, x_1 y x_2 usando el polinomio de interpolación de Lagrange.

5. La función de error se define como
$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, dt\ ,$$

Tabular la función de error usando la regla del trapecio con $x = 0{,}0, 0{,}1, 0{,}2, 0{,}3, 0{,}4$ y $0{,}5$. Usar 20, 40 y 100 subintervalos y comparar los resultados con los valores exactos (Matlab incorpora la función de error `erf`).

6. La función de Bessel de orden cero puede definirse como

$$J_0(x) = \frac{1}{\pi} \int_0^{\pi} \cos(x \operatorname{sen}(t))dt.$$

Utilizar esta definición de $J_0(x)$ para calcular los siete primeros ceros positivos de esta función $r_0, r_1, \ldots, r_6$. Comprobar gráficamente que si x no es demasiado grande:

$$J_0(x) \approx \left(1 - \frac{x^2}{r_0^2}\right)\left(1 - \frac{x^2}{r_1^2}\right) \cdots \left(1 - \frac{x^2}{r_6^2}\right) .$$

7. Aproxima las integrales siguientes utilizando las fórmulas de cuadratura de Gauss usando $n = 2, 3, 4$.

$$\text{a)} \int_1^{1{,}5} x^2 \ln(x)\, dx \ , \quad \text{b)} \int_0^{\pi/4} x^2 \operatorname{sen}(x)\, dx \ .$$

8. Determina las constantes a, b, c y d que generan una fórmula de cuadratura de la forma

$$\int_{-1}^{1} f(x)\, dx = af(-1) + bf(1)cf'(-1) + df'(1) \ ,$$

que es exacta para polinomios de grado menor o igual que 3.