

Interpolazione e Approssimazione

Dato un insieme di punti di ascisse e ordinate (x_j, f_j) mi serve qualche volta di avere a disposizione una funzione, di solito con proprietà particolari, che passi per tutti questi punti. La funzione che passa per N punti non è certo unica, ed il procedimento di trovare una di queste funzioni è detto **interpolazione**

Metodo di Lagrange

Un metodo semplice per interpolare N punti è quello dei polinomi di Lagrange. Per fare un esempio il polinomio che passa per tre punti può essere scritto come

$$P(x) = f_1 \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} + f_2 \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} + f_3 \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)}$$

Per un numero di punti grande (per esempio 20) il polinomio oscilla paurosamente e non rappresenta più, con ogni probabilità, l'andamento di una funzione fisicamente significativa.

Spline cubiche

Voglio una funzione che sia almeno continua con derivate prime e seconde; in ogni intervallo $[x_{j-1}, x_j]$ chiedo che la funzione sia un polinomio di terzo grado. Nei punti estremi devo imporre le condizioni di continuità per la funzione e le sue derivate prime e seconde

La funzione è cubica, quindi le sue derivate sono lineari. Chiamo M_j le derivate seconde in x_j . Allora

$$s_j''(x) = M_{j-1} \frac{x_j - x}{x_j - x_{j-1}} + M_j \frac{x - x_{j-1}}{x_j - x_{j-1}}$$

Posto $h_j = x_j - x_{j-1}$

$$s_j'(x) = -M_{j-1} \frac{(x_j - x)^2}{2h_j} + M_j \frac{(x - x_{j-1})^2}{2h_j} + C_j$$

$$s_j(x) = M_{j-1} \frac{(x_j - x)^3}{6h_j} + M_j \frac{(x - x_{j-1})^3}{6h_j} \\ + C_j(x - x_{j-1}) + E_j$$

Condizioni ai limiti

Devo imporre

$$s(x_{j-1}) = f_{j-1} \quad \text{e} \quad s(x_j) = f_j$$

e quindi

$$f_{j-1} = M_{j-1} \frac{h_j^2}{6} + E_j$$

$$f_j = M_j \frac{h_j^2}{6} + C_j h_j + E_j$$

$$E_j = f_{j-1} - M_{j-1} \frac{h_j^2}{6}$$

$$C_j = \frac{f_j - f_{j-1}}{h_j} - \frac{h_j}{6} (M_j - M_{j-1})$$

Impongo ora che la spline nell'intervallo j si raccordi con quella nell'intervallo $j+1$. Questo è già vero per la funzione (avendo imposto che sia uguale a f_{j-1} e f_j agli estremi) e per le derivate seconde (avendo scelto M_j in modo indipendente dall'intervallo). Rimane da imporre la continuità delle derivate prime.

Continuità delle derivate

In x_j

$$M_j \frac{h_j}{2} + C_j = -M_j \frac{h_{j+1}}{2} + C_{j+1}$$

$$M_j \frac{h_j}{2} + \frac{f_j - f_{j-1}}{h_j} - \frac{h_j}{6}(M_j - M_{j-1}) =$$
$$-M_j \frac{h_{j+1}}{2} + \frac{f_{j+1} - f_j}{h_{j+1}} - \frac{h_{j+1}}{6}(M_{j+1} - M_j)$$

Scrivendo questa come equazione per gli M_j ottengo

$$l_j M_{j-1} + 2M_j + u_j M_{j+1} = y_j$$

con

$$l_j = \frac{h_j}{h_j + h_{j+1}} \quad u_j = \frac{h_{j+1}}{h_j + h_{j+1}}$$

$$y_j = \frac{6}{h_j + h_{j+1}} \left(\frac{f_{j+1} - f_j}{h_{j+1}} - \frac{f_j - f_{j-1}}{h_j} \right)$$

Si possono anche imporre le due condizioni

$$2M_1 + u_1 M_2 = y_1 \quad l_N M_{N-1} + 2M_N = y_N$$

che è l'equazione lineare associata ad una matrice tridiagonale

Matrici tridiagonali

Una matrice che abbia elementi a_{ij} diversi da zero solo per $j = i, i \pm 1$ si dice tridiagonale. La matrice tridiagonale può essere memorizzata in tre array di dimensione al più N ed è particolarmente facile scomporla con un procedimento LU.

$$\begin{pmatrix} d_1 & u_1 & 0 & 0 \\ l_2 & d_2 & u_2 & 0 \\ 0 & l_3 & d_3 & u_3 \\ 0 & 0 & l_4 & d_4 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ l'_2 & 1 & 0 & 0 \\ 0 & l'_3 & 1 & 0 \\ 0 & 0 & l'_4 & 1 \end{pmatrix} \begin{pmatrix} d'_1 & u'_1 & 0 & 0 \\ 0 & d'_2 & u'_2 & 0 \\ 0 & 0 & d'_3 & u'_3 \\ 0 & 0 & 0 & d'_4 \end{pmatrix}$$

L'equazione si risolve facilmente osservando che $u'_j = u_j$ e $d'_1 = d_1$ le altre equazioni danno

$$l'_2 = l_2/d'_1 \quad d'_2 = d_2 - l'_2 u'_1$$

e, in generale,

$$l'_j = l_j/d'_{j-1} \quad d'_j = d_j - l'_j u'_{j-1}$$

A questo punto l'equazione lineare può essere risolta per doppia backsubstitution

Approssimazione

Se $f(x)$ è difficile da calcolare (impiega molto tempo) e mi servono i suoi valori solo con una certa precisione

Uso i polinomi di Chebychev

$$T_0(x) = 1 \quad T_1(x) = x$$
$$T_{n+1}(x) + T_{n-1}(x) = 2xT_n(x)$$

$$f(x) \approx \sum_{j=1}^N c_j T_{j-1}(x)$$

$$c_j = \sum_{k=1}^N f(x_k) \cos((j-1)x_k) \quad x_k = \frac{\pi(k - \frac{1}{2})}{N}$$