

Matematica applicata

Un goliardico riassunto

Ollari Dmitri

4 aprile 2023

Indice

1	Introduzione	3
1.1	Argomenti del corso	3
1.2	Modalità d'esame	4
2	Interpolazione	5
2.1	Teorema	5
2.2	Dimostrazione	5
2.3	Costruzione polinomio interpolatore	6
3	Interpolazione - Laplace	7
3.1	Polinomio di Newton	7
3.1.1	Polinomio interpolatore lineare	8
3.1.2	Polinomio interpolatore quadratico	8
3.1.3	Algoritmo di Horner	8
3.1.4	Newton con derivate(Hermite)	8
3.2	Interpolazione inversa	9
4	Interpolazione - Infiniti nodi interpolanti	10
4.1	Nodi <i>quasi</i> Chebichev	10
4.2	Convergenza del polinomio di interpolazione	11
5	Interpolazione - Splines	12
5.1	Decomposizione	12
5.2	Ricerca binaria	12
5.3	Splines	12
5.4	Stima dell'errore	13
5.5	Polinomio di Hermite generalizzato	13
5.6	Spline Interpolante	14
5.6.1	Definizione di spline interpolante	14
5.6.2	Spline cubiche	14
5.6.3	Momenti della spline	14
5.7	Spline cubica naturale	16
5.8	Spline cubica vincolata	16

Elenco delle figure

3.1	Schema iterazione polinomio Newton	7
3.2	Hermite	8
3.3	Esempio hermite	9
4.1	Chebichev con 5 nodi	10
4.2	Chebichev con 9 nodi	10
4.3	Chebichev con 17 nodi	11

Capitolo 1

Introduzione

1.1 Argomenti del corso

- Approssimazione di dati e funzioni:
 - interpolazione polinomiale
 - matrice di vandermonde
 - interpolazione di Lagrange
 - interpolazione di Hermite
 - definizione di differenza divisa
 - interpolazione (alla Newton) alle differenze divise
 - convergenza
 - controesempio di Runge su nodi equispaziati
 - rappresentazione dell'errore
 - funzioni a tratti splines
 - interpolazione con funzioni splines
 - metodo dei minimi quadrati
 - Cenno curve di Bézier
 - cenno interpolazione in più dimensioni
- Integrazione numerica
 - formula quadratica di interpolazione
 - formule di Newton-Cotes
 - studio dell'errore e della convergenza
 - routines automatiche
 - uso di formule per integrali in più dimensioni
- Sistemi lineari
 - metodi diretti
 - sistemi a matrice triangolare
 - metodo di eliminazione di Gauss
 - pivoting
 - decomposizione di Gauss e fattorizzazione a LU
 - matrice inversa
 - raffinamento iterativo
 - sistemi complessi
 - Cenni a metodi iterativi di Jacobi e di Gauss-Seidel
 - studio della convergenza dei metodi iterativi e criteri di arresto

- Equazioni non lineari
 - radici reali
 - metodo di Newton-Raphson
- Matlab
- Prerequisiti
 - operazioni tra matrici
 - matrici non singolari
 - determinante
 - cramer
 - regola di Laplace
 - matrice inversa
 - concetto di norma
 - norma di un vettore e di una matrice
 - lineare dipendenza e indipendenza

1.2 Modalità d'esame

Prova scritta su esercizi e prova orale di 1 ora :(

Capitolo 2

Interpolazione

Avendo alcuni punti, si considerano i seguenti problemi:

- Ricostruire una traiettoria passanti per i punti assegnati
- approssimare una funzione complessa nota in alcuni punti una più semplice come un polinomio
- calcolare il valore di un integrale definito di una funzione di cui non conosciamo facilmente una primitiva ad esempio approssimandola con un polinomio

2.1 Teorema

Se (x_i, y_i) con $i = 0, \dots, n$, sono $n + 1$ punti tali che $x_i \neq x_j$ se $i \neq j$, esiste ed è unico il polinomio $p_n(x)$ di grado al più n tale che:

$$p_n(x_i) = y_i, \quad i = 0, \dots, n \quad (2.1)$$

2.2 Dimostrazione

Consideriamo il generico polinomio di grado n :

$$p_n(x) = a_0 + a_1x + \cdots + a_nx^n \quad (2.2)$$

ed imponiamo le $n + 1$ condizioni (vincoli)

[illegible]

I parametri (coefficienti) incogniti a_0, a_1, \dots, a_n sono soluzione del sistema lineare 2.3 di ordine $n+1$, in forma matriciale si può scrivere come:

$$\begin{pmatrix} 1 & x_0 & \cdots & x_0^n \\ 1 & x_1 & \cdots & x_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}, \quad Va = y \quad (2.4)$$

La matrice V è detta **matrice di Vandermonde**, essa risulta **non singolare** se e soltanto se **il vettore nullo 0 è la sola soluzione** del sistema omogeneo.

$$Va = 0 \tag{2.5}$$

Quindi se il vettore $a = [a_0, a_1, \dots, a_n] \neq 0$ possiamo costruire il polinomio di grado n :

$$p_n(x) = a_0 + a_1x + \cdots + a_nx^n \quad (2.6)$$

Questo polinomio soddisfa anche la condizione:

$$p_n(x_i) = 0, \quad i = 0, 1, \dots, n \quad (2.7)$$

Cioè il polinomio $p_n(x)$ di grado n avrebbe $n + 1$ zeri (assurdo, vedi teorema fondamentale dell'algebra), di conseguenza **il vettore a deve essere il vettore nullo e quindi la matrice V è non singolare.**

In alternativa si può verificare che la matrice V è non singolare mediante il determinante, che se non nullo, il sistema ha una e una sola soluzione, quindi il polinomio $p_n(x)$ esiste ed è unico.

2.3 Costruzione polinomio interpolatore

Per ottenere una base che riduca il numero di calcoli cerco una matrice $V \equiv I$.

Parto con il costruire i polinomi Lagrangiani:

$$L_0(x_i) = \delta_{0,i} = \begin{cases} 1 & \text{se } i = 0 \\ 0 & \text{se } i \neq 0 \end{cases} \quad i = 0, 1, 2 \quad (2.8)$$

$$L_1(x_i) = \delta_{1,i} = \begin{cases} 1 & \text{se } i = 1 \\ 0 & \text{se } i \neq 1 \end{cases} \quad i = 0, 1, 2 \quad (2.9)$$

$$L_2(x_i) = \delta_{2,i} = \begin{cases} 1 & \text{se } i = 2 \\ 0 & \text{se } i \neq 2 \end{cases} \quad i = 0, 1, 2 \quad (2.10)$$

$L_0(x)$ si deve annullare in x_1 e x_2 , quindi:

$$L_0(x) = a_0(x - x_1)(x - x_2) \quad (2.11)$$

Con a_0 costante arbitraria non nulla e imponendo che $L_0(x_0) = 1$ ottengo:

$$L_0(x_0) = a_0(x_0 - x_1)(x_0 - x_2) = 1 \quad \rightarrow \quad a_0 = \frac{1}{(x_0 - x_1)(x_0 - x_2)} \quad (2.12)$$

Ripetendo lo stesso procedimento per $L_1(x)$ e $L_2(x)$ ottengo:

$$L_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}, \quad L_0(x_0) = 1, L_0(x_1) = 0, L_0(x_2) = 0 \quad (2.13)$$

$$L_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)}, \quad L_1(x_0) = 0, L_1(x_1) = 1, L_1(x_2) = 0 \quad (2.14)$$

$$L_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}, \quad L_2(x_0) = 0, L_2(x_1) = 0, L_2(x_2) = 1 \quad (2.15)$$

Ciascun polinomio è di secondo grado, inoltre i polinomi $L_0(x)$, $L_1(x)$ e $L_2(x)$ sono linearmente indipendenti.

Quindi posso esprimere un generico polinomio di secondo grado come combinazione dei tre nuovi polinomi costruiti:

$$p_2(x) = a_0L_0(x) + a_1L_1(x) + a_2L_2(x) \quad a_0, a_1, a_2 \in R \quad (2.16)$$

Assegnati tre valori di y possiamo costruire il polinomio interpolatore $p_2(x)$ usando la nuova base Lagrangiana:

$$\begin{cases} a_0L_0(x_0) + a_1L_1(x_0) + a_2L_2(x_0) = y_0 \\ a_0L_0(x_1) + a_1L_1(x_1) + a_2L_2(x_1) = y_1 \\ a_0L_0(x_2) + a_1L_1(x_2) + a_2L_2(x_2) = y_2 \end{cases} \quad (2.17)$$

Mediante il polinomio interpolatore posso scrivere i punti (x_0, y_0) , (x_1, y_1) e (x_2, y_2) come:

$$p_2(x) = y_0L_0(x) + y_1L_1(x) + y_2L_2(x) \quad (2.18)$$

Capitolo 3

Interpolazione - Laplace

Siano x_0, x_1, \dots, x_n , $n + 1$ **punti distinti** e siano $L_0(x), L_1(x), \dots, L_n(x)$ polinomi di grado n tali che:

$$L_j(x_i) = \delta_{j,i} = \begin{cases} 1 & \text{se } i = j \\ 0 & \text{se } i \neq j \end{cases} \quad i = 0, 1, \dots, n \quad (3.1)$$

Dove:

$$L_n(x) = \prod_{k=0, k \neq n}^n \frac{x - x_k}{x_n - x_k} \quad (3.2)$$

Trovare la funzione interpolatrice dati i seguenti valori

$$\begin{array}{ccc} x_0 & x_1 & x_2 \\ y_0 & y_1 & y_2 \\ y'_0 & & \end{array}$$

Quindi devo usare costruire il sistema:

$$\begin{cases} a_0 + a_1 x_0 + a_2 x_0^2 + a_3 x_0^3 = y_0 \\ a_1 + 2a_2 x_0^1 + 3a_3 x_0^2 = y'_0 \\ a_0 + a_1 x_1 + a_2 x_1^2 + a_3 x_1^3 = y_1 \\ a_0 + a_1 x_2 + a_2 x_2^2 + a_3 x_2^3 = y_2 \end{cases} \quad (3.3)$$

E poi da qui posso risolvere il sistema.

3.1 Polinomio di Newton

Si chiama differenza divisa di ordine k della funzione $f(x)$ relativa ai punti x_0, \dots, x_{k-1} la funzione $f[x_0, \dots, x_{k-1}, x]$ definita per $x \neq x_i$ con $i = 0, \dots, k - 1$ ricorsivamente come:

$$\begin{array}{ccccccc} x_0 & f(x_0) & & & & & \\ & f[x_0, x_1] & & & & & \\ x_1 & f(x_1) & f[x_0, x_1, x_2] & & & & \\ & f[x_1, x_2] & f[x_0, x_1, x_2, x_3] & & & & \\ x_2 & f(x_2) & f[x_1, x_2, x_3] & f[x_0, x_1, x_2, x_3, x_4] & & & \\ & f[x_2, x_3] & f[x_1, x_2, x_3, x_4] & \dots & & & \\ x_3 & f(x_3) & f[x_2, x_3, x_4] & & & & \\ & f[x_3, x_4] & & \dots & & & \\ x_4 & f(x_4) & & & & & \\ \dots & \dots & & & & & \end{array}$$

Figura 3.1: Schema iterazione polinomio Newton

Per calcolare il polinomio finale di Newton si deve iterare come segue:

Se siamo nella prima colonna(dove ci sono gli x_n):

$$f[x] = f(x) \quad (3.4)$$

Se ci troviamo nella colonna 1(dove si trovano $f(x_n)$), prendo due x alla volta e eseguo:

$$f[x_0, x_1] = \frac{f[x_1] - f[x_0]}{x_1 - x_0} \quad (3.5)$$

Se mi trovo nelle colonne dalla numero 2 in poi (andando verso destra):

$$f[x_0, \dots, x_{k-2}, x_{k-1}, x] = \frac{f[x_1, \dots, x_{k-1}, x] - f[x_0, \dots, x_{k-1}, x_{k-1}]}{x - x_0} \quad (3.6)$$

Il polinomio ottenuto con Newton è al più di grado n , nel polinomio interpolatore di newton, i coefficienti $[x_0, \dots, x_n]$ sono indipendenti da x .

3.1.1 Polinomio interpolatore lineare

$$p_1(x) = f[x_0] + (x - x_0) \frac{f(x_1) - f(x_0)}{x_1 - x_0} \quad (3.7)$$

3.1.2 Polinomio interpolatore quadratico

$$p_2(x) = f[x_0] + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2] \quad (3.8)$$

3.1.3 Algoritmo di Horner

Algoritmo ottimale per calcolare un polinomio in un punto:

Il polinomio:

$$p(x) = a_0 + a_1x + a_2x^2 + a_3x^3 \quad (3.9)$$

Si può riscrivere come:

$$p(x) = ((a_3x + a_2)x + a_1)x + a_0 \quad (3.10)$$

3.1.4 Newton con derivate(Hermite)

Se i punti $f(x)$ appartengono all'intervallo chiuso e limitato $[a, b]$, non necessariamente distinti, esiste un punto ϵ compreso tra il minimo e il massimo tale che:

$$f[x_0, \dots, x_k] = \frac{f^{(k)}(\epsilon)}{k!} \quad (3.11)$$

Se si conoscono anche le derivate prime di $f(x)$ si può fare:

x_0	$f(x_0)$	$f'(x_0)$		
x_1	$f(x_1)$	$f[x_0, x_1]$	$f[x_0, x_0, x_1]$	
x_1	$f(x_1)$	$f'(x_1)$	$f[x_0, x_1, x_1]$	
\vdots	\vdots	\vdots	\vdots	
x_n	$f(x_n)$	$f[x_{n-1}, x_n]$	$f[x_{n-1}, x_n, x_n]$	$\dots \dots f[x_0, x_0, x_1, x_1, \dots, x_n, x_n]$

Figura 3.2: Hermite

Esempio: Se nel punto x_0 ho tre vincoli:

- $f(x)$
- $f'(x)$
- $f''(x)$

Posso realizzare lo schema delle differenze divise come:

Il polinomio risulta così:

$$p(x) = f[x_0] + f[x_0, x_0](x - x_0) + f[x_0, x_0, x_0](x - x_0)^2 \quad (3.12)$$

$$\begin{array}{rcl}
x_0 & y_0 & \\
& f[x_0, x_0] = f'(x_0) & \\
x_0 & y_0 & f[x_0, x_0, x_0] = \frac{f''(x_0)}{2!} \\
& f[x_0, x_0] = f'(x_0) & \\
x_0 & y_0 &
\end{array}$$

Figura 3.3: Esempio hermite

3.2 Interpolazione inversa

Se la funzione da interpolare $f(x)$ è:

- monotona in senso stretto in $[a, b]$

quindi è invertibile, la formula di interpolazione di Newton può essere usata per ottenere l'inversa.

Basta scambiare i punti x, y .

Esempio: Dati i seguenti punti:

$$\begin{array}{r|rrrr}
x_i & 0.2 & 0.4 & 0.6 & 0.8 \\
f(x_i) & 0.203 & 0.423 & 0.684 & 1.030 \\
\\
y_i & 0.203 & 0.423 & 0.684 & 1.030 \\
x_i = f^{-1}(y_i) & 0.2 & 0.4 & 0.6 & 0.8
\end{array}$$

Capitolo 4

Interpolazione - Infiniti nodi interpolanti

Il teorema di Faber afferma che non tutti i nodi dell'intervallo convergono uniformemente alla funzione che si cerca di approssimare con infiniti nodi.

4.1 Nodi *quasi* Chebichev

$$x_k^n = \cos\left(\frac{(2k+1)\pi}{2(n+1)}\right) \quad k = 0, \dots, n \quad (4.1)$$

Esempio 5 nodi

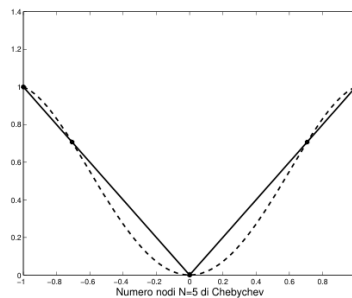


Figura 4.1: Chebichev con 5 nodi

Esempio 9 nodi

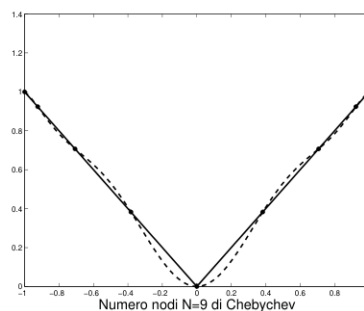


Figura 4.2: Chebichev con 9 nodi

Esempio 17 nodi

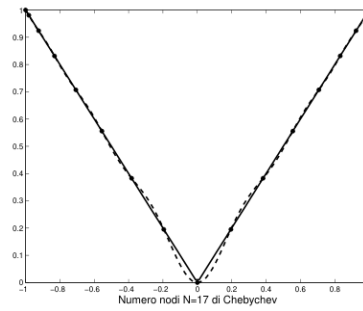


Figura 4.3: Chebichev con 17 nodi

4.2 Convergenza del polinomio di interpolazione

Il teorema di Bernstein dice che:

Se $f(x) \in C^1([a, b])$, il polinomio $p_n(x)$ di interpolazione della funzione f relativo agli zeri del polinomio di Chebichev di grado $n + 1$ converge uniformemente a $f(x)$ su $[a, b]$ per $n \rightarrow +\infty$.

Se la funzione $f \in C^2([a, b])$ si ha che la stima dell'errore è:

$$\|f(x) - p_n(x)\|_\infty = O\left(\frac{1}{\sqrt{n}}\right) \quad (4.2)$$

Se l'intervallo usato per il polinomio interpolatore è $[0, 1]$ e i nodi sono n :

$$B_{n,k}(x) = \binom{n}{k} x^k (1-x)^{n-k} \quad k = 0, \dots, n \quad (4.3)$$

Dove:

$$\binom{n}{k} = \frac{n!}{k!(n-k)!} \quad (4.4)$$

Il teorema di Hermite-Fejer dice che:

Sia $f(x) \in C^0([a, b])$, con $[a, b]$ limitato e chiuso e sia $p_{2n+1}(x)$ il polinomio di grado $2n + 1$ tale che:

- $p_{2n+1}(x_i) = f(x_i)$
- $p'_{2n+1}(x_i) = 0$

Si ha che gli zeri del polinomio di Chebichev sull'intervallo tendono a zero per infiniti nodi.

Interpolazione - Splines

5.1 Decomposizione

Sia $[a, b]$ un intervallo chiuso e limitato, Δ è una sua decomposizione:

$$\Delta = \{a = x_0 < x_1 < \cdots < x_i < \cdots < x_n = b\} \quad (5.1)$$

e siano dati i valori osservati y_0, \dots, y_n , vogliamo costruire una decomposizione su ciascun tratto $[x_{i-1}, x_i]$ della decomposizione Δ un polinomio lineare che interpoli i dati y_{i-1}, y_i .

[illegible]

Osservo che la continuità $S_1^{(i)} \in C^\infty([x_i, x_{i+1}])$ mache la continuità in $S_1(x) \in C^0[a, b]$. Quindi si hanno tanti polinomi di primo grado definiti a tratti.

5.2 Ricerca binaria

Per calcolare $S_1(x)$ in un generico punto bisogna prima determinare in che sottointervallo cade il punto, il modo più comodo è fare una ricerca binaria.

```
x = [1:0.1:10]
xd = 3.17
n = length(x);
sinistra = 1;
destra = n;

while destra > sinistra + 1
    meta = floor((sinistra + destra) / 2);
    if xd < x(meta);
        destra = meta;
    else
        sinistra = meta;
    end
end
```

5.3 Splines

Siano $\varphi(x), i = 0, 1, \dots, n$ le funzioni così definite:

$$\varphi_0(x) = \begin{cases} \frac{x_1 - x}{x_1 - x_0} & x \in [x_0, x_1] \\ 0 & x \notin [x_0, x_1] \end{cases} \quad (5.3)$$

$$\varphi_i(x) = \begin{cases} 0 & x \notin [x_{i-1}, x_{i+1}] \\ \frac{x - x_{i-1}}{x_i - x_{i-1}} & x \in [x_{i-1}, x_i] \\ \frac{x_{i+1} - x}{x_{i+1} - x_i} & x \in [x_i, x_{i+1}] \end{cases} \quad (5.4)$$

$$\varphi_n(x) = \begin{cases} \frac{x - x_{n-1}}{x_n - x_{n-1}} & x \in [x_{n-1}, x_n] \\ 0 & x \notin [x_{n-1}, x_n] \end{cases} \quad (5.5)$$

Le funzioni $\varphi_i(x)$ prendono il nome di funzioni **spline** di grado 1, e verificano le seguenti condizioni:

$$\varphi_i(x_j) = \begin{cases} 1 & j = i \\ 0 & j \neq i \end{cases} \quad (5.6)$$

Sono linearmente indipendenti sull'intervallo $[a, b]$ (facendo la combinazione lineare e uguagliandola a zero, i coefficienti sono tutti zero), sono base canonica.

Gli errori sui dati si riducono quando ci si allontana dal punto usato come intervallo locale della spline

5.4 Stima dell'errore

Siano $y_i = f(x_i), i = 0, 1, \dots, n$ dove $f(x)$ è definita nell'intervallo $[a, b]$. $S_1(x)$ è la spline lineare che interpola la funzione, l'errore è dato da:

$$r(x) = f(x) - S_1(x) \quad \forall x \neq x_i \quad (5.7)$$

Indico con h_i l'ampiezza dell'intervallo dove voglio calcolare il punto.

Se $f(x) \in C^2[a, b]$ allora posso scrivere:

$$|f(x) - S_1(x)| = \frac{|f''(\epsilon_x)|}{2} |(x - x_{i-1})(x - x_i)| \quad (5.8)$$

Quindi per una funzione $f(x) \in C^2[a, b]$ dopo vari passaggi tediosi posso scrivere:

$$|f(x) - S_1(x)| \leq \max_{x \in [x_0, x_n]} \frac{|f''(\epsilon_x)|}{8} h_2 \quad (5.9)$$

Dove $h = \max_{1 \leq i \leq n} h_i$ è la norma della decomposizione.

Con h che tende a zero, l'errore tende a zero e aumentano i polinomi per descrivere l'intervallo.

5.5 Polinomio di Hermite generalizzato

Vogliamo costruire un polinomio definito a tratti che in ogni intervallo coincide con la restrizione di un polinomio di grado minore o uguale a 3, soddisfacendo le seguenti condizioni:

$$p(x_i) = y_i \quad i = 0, 1, \dots, n \quad (5.10)$$

$$p'(x_i) = y'_i \quad i = 0, 1, \dots, n \quad (5.11)$$

Considerando la seguente forma del polinomio:

$$p_i(x) = a_i + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^2(x - x_i) \quad (5.12)$$

Dove la sua derivata è definita:

$$p'_i(x) = b_i + 2c_i(x - x_{i-1}) + 3d_i(x - x_{i-1})^2 \quad (5.13)$$

Per determinare i coefficienti imponiamo le 4 condizioni dettate da $h_i = x_i - x_{i-1}$:

1. $p_i(x_{i-1}) = y_{i-1}$ da cui ottengo $a_i = y_{i-1}$
2. $p'_i(x_{i-1}) = y'_{i-1}$ da cui ottengo $b_i = y'_{i-1}$
3. $p_i(x_i) = y_i = a_i + b_i h_i + c_i h_i^2$ da cui ottengo $c_i = \frac{y_i - y_{i-1}}{h_i^2} - \frac{y'_{i-1}}{h_i}$

4. $p'_i(x_i) = y'_i$ da cui ottengo $d_i = \frac{y'_i - y'_{i-1} - 2c_i h_i}{h_i^2}$

Sostituendo c_i si ottiene l'espressione di d_i :

$$d_i = \frac{y'_i + y'_{i-1}}{h_i^2} - 2 \frac{y_i - y_{i-1}}{h_i^3} \quad (5.14)$$

Il polinomio 5.12 prende il nome di polinomio di **Hermite generalizzato**.

5.6 Spline Interpolante

5.6.1 Definizione di spline interpolante

Sia $\Delta = \{a = x_0 < \dots < x_i = b\}$ una decomposizione dell'intervallo $[a, b]$. Una funzione spline di grado m con nodi x_i e' una funzione $S_m(x)$ in $[a, b]$ tale che su ogni sottointervallo $[x_{i-1}, x_i]$, $S_m(x)$ e' un polinomio di grado m ed e' derivabile $m - 1$ volte:

$$S_m(x) \in C^{m-1}([a, b]) \quad (5.15)$$

5.6.2 Spline cubiche

Avendo un intervallo $[a, b]$ con decomposizione Δ , assegnanti arbitrariamente i valori delle y , si dice spline cubica interpolante relativa alla decomposizione Δ la funzione $S_{3,\Delta}(x)$ tale che:

1. La spline cubica $S_{3,\Delta}(x)$ e' una funzione polinomiale definita a tratti e su ciascun tratto della decomposizione vale come un polinomio di terzo grado
2. $S_{3,\Delta}(x) \in C^2([a, b])$
3. $S_{3,\Delta}(x_i) = y_i$

La spline cubica si scrive:

[illegible]

I gradi di libert  delle incognite sono $4n$ (coefficienti della spline). I vincoli sono $3(n-1)$ per la regolarit  $C^2([x_0, x_n])$ e $n+1$ vincoli per l'interpolazione fra gli $n+1$ nodi.

I vincoli si sommano in $4n - 2$ che ci permette di costruire ∞^2 spline cubiche interpolanti.

5.6.3 Momenti della spline

Per ridurre la complessita' del calcolo della spline, si usano i **Momenti** della spline:

$$M_i := [S''_{3,\Delta} x]_{x=x_i} \quad (5.17)$$

Se fossero noti momenti M_{i-1} e M_i potrei scrivere:

$$[S_{3,\Delta}^i]'' = \frac{(x - x_{i-1}M_i + (x_i - x)M_{i-1})}{h_i} \quad (5.18)$$

Integrando due volte la funzione si ottiene:

$$S_{3,\Delta}^i(x) = \frac{M_i}{6h_i}(x - x_{i-1})^3 + \frac{M_{i-1}}{6h_i}(x_i - x)^3 + A_i(x - x_{i-1}) + B_i \quad (5.19)$$

Dove A_i e B_i sono costanti da determinare introdotte dalla doppia integrazione nell'intervallo $[x_{i-1}, x_i]$:

1. in x_{i-1} si ottiene $y_{i-1} = S_{3,\Delta}^i(x_{i-1}) \Rightarrow B_i = y_{i-1} - \frac{h_i^2 M_{i-1}}{6}$
2. in x_i si ottiene $y_i = S_{3,\Delta}^i(x_i) \Rightarrow A_i = \frac{y_i}{h_i} - \frac{h_i M_i}{6} - \frac{B_i}{h_i}$

Sostituendo A_i e B_i si ottiene la spline cubica nell'intervallo $[x_{i-1}, x_i]$ in funzione dei momenti M_{i-1} e M_i :

$$S_{3,\Delta}^i(x) = \frac{(x - x_{i-1})^3}{6h_i} M_i + \frac{(x_i - x)^3}{6h_i} M_{i-1} + (x - x_{i-1}) \left[\frac{y_i - y_{i-1}}{h_i} + \frac{h_i}{6} (M_{i-1} - M_i) \right] + y_{i-1} - h_i^2 \frac{M_{i-1}}{6} \quad (5.20)$$

La derivata prima della spline e':

$$[S_{3,\Delta}^i(x)]' = \frac{(x - x_{i-1})^2}{2h_i} M_i - \frac{(x_i - x)^2}{2h_i} M_{i-1} + \frac{y_i - y_{i-1}}{h_i} + \frac{h_i}{6} (M_{i-1} - M_i) \quad (5.21)$$

La derivata seconda e':

$$[S_{3,\Delta}^i(x)]'' = \frac{(x - x_{i-1})}{h_i} M_i + \frac{(x_i - x)}{h_i} M_{i-1} \quad (5.22)$$

La derivata terza e':

$$[S_{3,\Delta}^i(x)]''' = \frac{1}{h_i} (M_i + M_{i-1}) \quad (5.23)$$

Calcolare i coefficienti della spline partendo da i momenti

- $a_0^i = y_{i-1}$
- $a_1^i = \frac{y_i - y_{i-1}}{h_i} - \frac{h_i}{6} (2M_{i-1} + M_i)$
- $a_2^i = \frac{M_{i-1}}{2}$
- $a_3^i = \frac{M_i - M_{i-1}}{6h_i}$

Calcolo dei momenti

Prendo due intervalli contigui e impongo la derivabilita' prima nel nodo in comune:

$$[S_{3,\Delta}^i(x)]'_{x=x_i} = [S_{3,\Delta}^{i+1}(x)]'_{x=x_i}, \quad i = 1, \dots, n-1 \quad (5.24)$$

Facendo riferimento alla derivata prima della spline (5.21), pongo le seguenti variabili per comodita' a:

$$\alpha_i = \frac{h_i}{h_i + h_{i+1}} \quad (5.25)$$

$$\beta_i = \frac{h_{i+1}}{h_i + h_{i+1}} \quad (5.26)$$

$$d_i = \frac{6}{h_i + h_{i+1}} \left(\frac{y_{i+1} - y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i} \right) \quad (5.27)$$

Posso riscrivere la spline come:

$$\alpha_i M_{i-1} + 2M_i + \beta_i M_{i+1} = d_i, \quad i = 1, \dots, n-1 \quad (5.28)$$

Scrivo due righe di esempi perche' il valore della i ed il suo indice rendono tutto sbatti:

$$i = 1 \quad \alpha_1 M_0 + 2M_1 + \beta_1 M_2 = d_1$$

$$i = 2 \quad \alpha_2 M_1 + 2M_2 + \beta_2 M_3 = d_2$$

Possiamo cosi' costruire la matrice:

$$\begin{bmatrix} \alpha_1 & 2 & \beta_1 & 0 & \dots & \dots & 0 \\ 0 & \alpha_2 & 2 & \beta_2 & 0 & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots & 0 \\ 0 & \dots & \dots & 0 & \alpha_{n-1} & 2 & \beta_{n-1} \end{bmatrix} \begin{bmatrix} M_0 \\ M_1 \\ \dots \\ M_n \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \dots \\ d_{n-1} \end{bmatrix} \quad (5.29)$$

5.7 Spline cubica naturale

Una spline cubica e' naturale se:

- $M_0 = 0$
- $M_n = 0$

Che ci porta ad avere:

$$\begin{bmatrix} 2 & \beta_1 & 0 & \cdots & 0 \\ \alpha_2 & 2 & \beta_2 & 0 & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & 0 & \alpha_{n-1} & 2 \end{bmatrix} \begin{bmatrix} M_1 \\ \vdots \\ M_{n-1} \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_{n-1} \end{bmatrix} \quad (5.30)$$

Otengo cosi' una matrice triangolare con le seguenti proprieta':

- $\alpha_i + \beta_i = 1$
- $i = 2, \dots, n-2$
- $0 < \beta_1 \leq 1$
- $0 < \alpha_{n-1} \leq 1$

Quindi stiamo parlando di una matrice diagonal dominante che ci permette di identificare la matrice come **non singolare**.

Quindi possiamo risolvere il sistema, determinando i momenti e successivamente i coefficienti della spline:

- $a_0^i = y_{i-1}$
- $a_1^i = \frac{y_i - y_{i-1}}{h_i} - \frac{h_i}{6} (2M_{i-1} + M_i)$
- $a_2^i = \frac{M_{i-1}}{2}$
- $a_3^i = \frac{M_i - M_{i-1}}{6h_i}$

5.8 Spline cubica vincolata

Le condizioni per questa famiglia di spline sono:

- $[S_{s,\Delta}(x)]'_{x=x_0} = y'_0$
- $[S_{s,\Delta}(x)]'_{x=x_n} = y'_n$

Quindi si parte dalla formula della derivata prima della spline(5.21) e la si equaglia a y'_0 .
L'equazione ottenuta e':

$$2M_0 + M_1 = d_0 \quad (5.31)$$

Dove d_0 :

$$d_0 = \frac{6}{h_1} \left(\frac{y_1 - y_0}{h_1} - y'_0 \right) \quad (5.32)$$

E con il secondo vincolo si ottiene:

$$M_{n-1} + 2M_n = d_n \quad (5.33)$$

Dove d_n :

$$d_n = \frac{6}{h_n} \left(y'_n - \frac{y_n - y_{n-1}}{h_n} \right) \quad (5.34)$$