

Calcolo Numerico con elementi di programmazione

(A.A. 2014-2015)

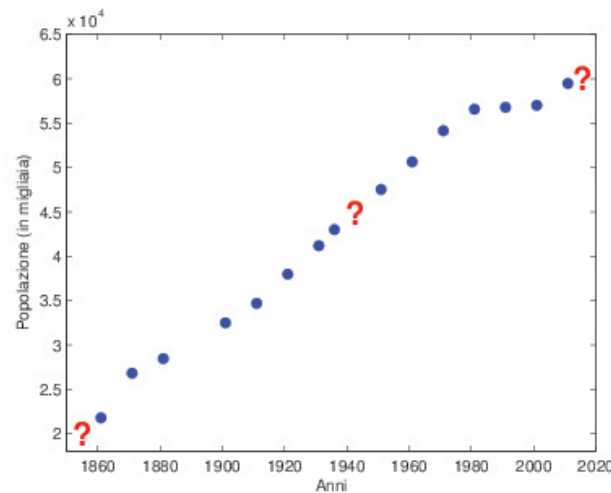
Appunti delle lezioni
sull'Approssimazione di dati e funzioni

Esempio 1

Nella **tavola** seguente è riportata la **popolazione** (in migliaia) dell'Italia censita ogni **10 anni** tra il **1861** e il **2011** (dati ISTAT).

Anno	1861	1871	1881	1901	1911	1921	1931	1936
Popolazione (in migliaia)	21.777	26.801	28.460	32.475	34.671	37.974	41.177	42.994

Anno	1951	1961	1971	1981	1991	2001	2011
Popolazione (in migliaia)	47.516	50.624	54.137	56.557	56.778	56.996	59.464



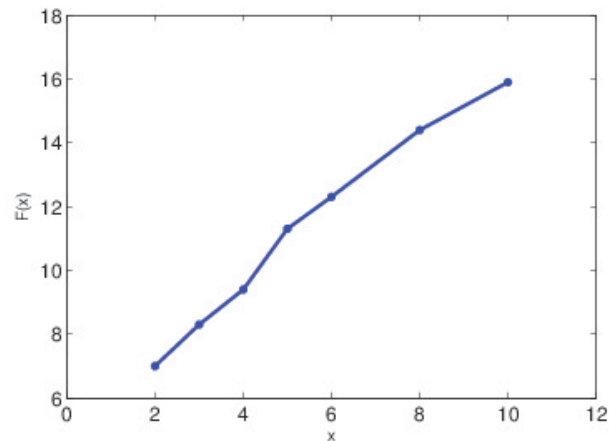
Determinare il **tasso di crescita** e approssimare la popolazione negli anni **1850**, **1943**, **2015**. Come si può stimare l'**accuratezza** dei valori ottenuti?

Esempio 2

Nella tabella sono riportate le **misure sperimentali** relative alla **forza** $F(x)$ necessaria per allungare una **molla** fino alla **lunghezza** x .

x	2	3	4	5	6	8	10
$F(x)$	7.0	8.3	9.4	11.3	12.3	14.4	15.9

Determinare la **costante elastica** della molla.



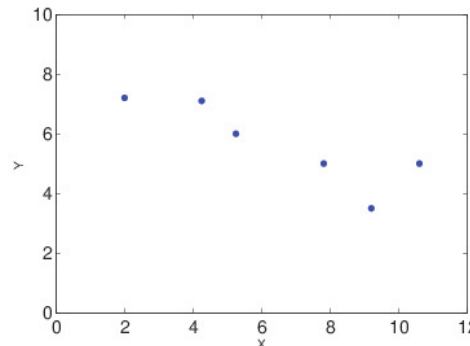
Esempio 3

Per **misurare il diametro** di alcuni fori di **coordinate**

x pollici	2.00	4.25	5.25	7.81	9.20	10.60
y pollici	7.2	7.1	6.0	5.0	3.5	5.0

praticati su una lastra di metallo di dimensione $15'' \times 10''$ viene utilizzato un laser collocato all'estremità di un **braccio meccanico** di un robot.

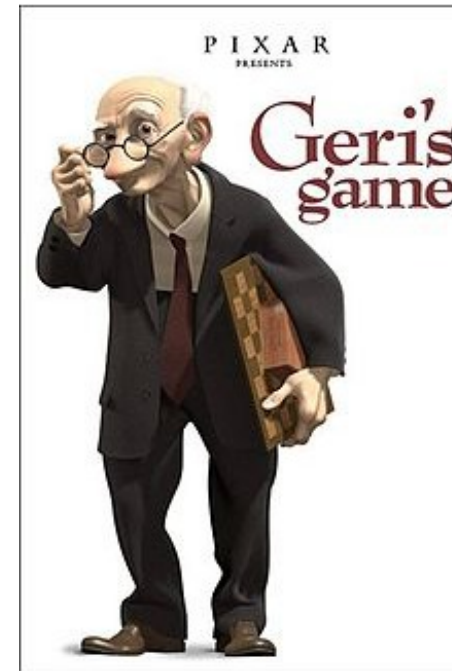
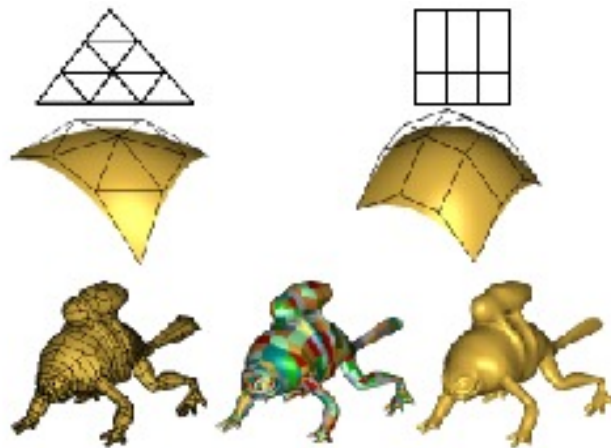
Determinare il **cammino ottimale** che il braccio deve compiere per collegare i fori. Il cammino deve essere sufficientemente **regolare**, in modo da impedire variazioni di direzione troppo brusche, e **breve**.



Esempio 4

Date le coordinate di alcuni punti nello spazio, disegnare una superficie che riproduca l'andamento dei punti.

→ Computer graphics, disegno di fonts ...



1998 Academy Award
Best Animated Short Film

Approssimazione di dati e funzioni

Problema

Data la **tabella** $\{x_i, y_i\}$, $i = 0, \dots, n$, si vuole trovare una **funzione analitica** φ_M che **approssimi** i dati.

La **tabella** $\{x_i, y_i\}$ può essere il risultato di **misure sperimentali** oppure può rappresentare i valori di una funzione la cui **espressione analitica** è nota ma **complicata** da calcolare direttamente.

Per poter costruire una **funzione approssimante** bisogna stabilire

- in quale **classe** di funzioni si vuole operare
- il **metodo di approssimazione**

Classi di funzioni approssimanti

- **Polinomi algebrici** di grado n a coefficienti reali ($n + 1$ parametri)

$$p_n(x) = a_0 + a_1x + \cdots + a_{n-1}x^{n-1} + a_nx^n \quad a_k \in \mathbf{R} \quad \forall k$$

adatti a rappresentare **funzioni appartenenti a** $C^n([a, b])$

- **Polinomi trigonometrici** di ordine n a coefficienti reali: ($2n + 1$ parametri)

$$t_n(x) = \sum_{k=0}^n (a_k \cos(kx) + b_k \sin(kx)) \quad a_k, b_k \in \mathbf{R} \quad \forall k$$

adatti per rappresentare **funzioni periodiche**

- **Funzioni razionali**: ($m + n + 2$ parametri)

$$r_{n,m}(x) = \frac{p_n(x)}{p_m(x)} \quad p_n, p_m \text{ polinomi}$$

adatte per rappresentare **funzioni con singolarità**

- **Funzioni esponenziali:** ($2n$ parametri)

$$g_n(x) = \sum_{k=1}^n a_k \exp(b_k x) \quad a_k, b_k \in \mathbf{R} \quad \forall k$$

adatte per rappresentare **fenomeni fisici ad andamento esponenziale**

- **Funzioni splines:** polinomi a tratti di grado n e regolarità C^{n-1}
adatte a rappresentare **funzioni polinomiali a tratti**

Metodi di approssimazione

- **Interpolazione**: si sceglie la funzione approssimante φ_n in modo che

$$\varphi_n(x_i) = y_i \quad i = 0, 1, \dots, n \quad \text{Condizioni di interpolazione}$$

Si usa quando i dati y_i sono **accurati**.

- **Approssimazione ai minimi quadrati** di dati discreti: si sceglie la funzione approssimante φ_M in modo che **minimizzi** la quantità

$$\sum_{i=0}^n [\varphi_M(x_i) - y_i]^2 \quad \text{Scarto quadratico}$$

oppure, introducendo i **pesi** w_i ,

$$\sum_{i=0}^n w_i [\varphi_M(x_i) - y_i]^2 \quad \text{Scarto quadratico pesato}$$

Si usa quando i dati y_i sono poco **accurati** e in **numero elevato**.

Esempio: retta di regressione

Interpolazione

Tabella: $\{x_i, y_i\}$ $i = 0, 1, \dots, n$

Funzione approssimante:

$\varphi_n(x)$ dipende **linearmente** da $n + 1$ **parametri**:

$$a_0, a_1, \dots, a_n$$



$$V = \begin{bmatrix} \psi_0(x_0) & \psi_1(x_0) & \psi_2(x_0) & \cdots & \psi_n(x_0) \\ \psi_0(x_1) & \psi_1(x_1) & \psi_2(x_1) & \cdots & \psi_n(x_1) \\ \dots & \dots & \dots & \dots & \dots \\ \psi_0(x_n) & \psi_1(x_n) & \psi_2(x_n) & \cdots & \psi_n(x_n) \end{bmatrix} \in \mathbf{R}^{(n+1) \times (n+1)}$$

$$\underbrace{V^T V}_H \mathbf{A} = \underbrace{V^T Y}_B \Rightarrow \boxed{V \mathbf{A} = Y}$$

Risolvere il **problema dell'interpolazione** equivale a risolvere il sistema lineare $V \mathbf{A} = Y$

Condizioni di interpolazione

$$VA = Y$$



$$\left\{ \begin{array}{l} \underbrace{a_0 \psi_0(x_0) + a_1 \psi_1(x_0) + a_2 \psi_2(x_0) + \cdots + a_n \psi_n(x_0)}_{\varphi_n(x_0)} = y_0 \\ \underbrace{a_0 \psi_0(x_1) + a_1 \psi_1(x_1) + a_2 \psi_2(x_1) + \cdots + a_n \psi_n(x_1)}_{\varphi_n(x_1)} = y_1 \\ \vdots \\ \underbrace{a_0 \psi_0(x_n) + a_1 \psi_1(x_n) + a_2 \psi_2(x_n) + \cdots + a_n \psi_n(x_n)}_{\varphi_n(x_n)} = y_n \end{array} \right.$$



$$\varphi_n(x_0) = y_0, \quad \varphi_n(x_1) = y_1, \quad \cdots \quad \varphi_n(x_n) = y_n$$

⇒ La funzione interpolante **passa** per i valori $\{x_i, y_i\}$.

Interpolazione polinomiale

Tabella: $\{x_i, y_i\} \quad i = 0, \dots, n$

Intervallo di interpolazione: $[a, b] = [x_0, x_n]$

Funzione approssimante: $p_n(x) = a_0 + a_1x + \dots + a_{n-1}x^{n-1} + a_nx^n$

Metodo di approssimazione: $p_n(x_i) = y_i$
 $(i = 0, 1, \dots, n) \rightarrow$ **Interpolazione**

Risolvere il **problema dell'interpolazione** vuol dire individuare il polinomio p_n , cioè i **coefficienti reali** a_k , che soddisfano le **condizioni di interpolazione**. Questo equivale a risolvere il **sistema lineare**

$$\begin{cases} p_n(x_0) = a_0 + a_1x_0 + a_2x_0^2 + \dots + a_nx_0^n = y_0 \\ p_n(x_1) = a_0 + a_1x_1 + a_2x_1^2 + \dots + a_nx_1^n = y_1 \\ \vdots \\ p_n(x_n) = a_0 + a_1x_n + a_2x_n^2 + \dots + a_nx_n^n = y_n \end{cases} \rightarrow VA = Y$$

Unicità del polinomio interpolatore

$$\boxed{VA = Y} \text{ con } V = \underbrace{\begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix}}_{\text{Matrice di Vandermonde}} \quad A = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} \quad Y = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}$$

La **matrice di Vandermonde** di $n+1$ nodi **distinti** $\{x_i\}$, $i = 0, \dots, n+1$, è **regolare** poiché

$$\det V = \prod_{j>i} (x_i - x_j) \neq 0$$

\Rightarrow esiste un'**unica** soluzione A del sistema.



Esiste **uno e uno solo** polinomio p_n di grado n che verifica le **condizioni di interpolazione**

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

Unicità : se $\exists p_1, p_2 : p_1(x_i) = p_2(x_i) = y_i \Rightarrow (p_1 - p_2)(x_i) = 0 \Rightarrow p_1 - p_2$ si annulla in $n+1$ punti \Rightarrow assurdo perchè è un polinomio di grado n .

Condizionamento della matrice di Vandermonde

La **matrice di Vandermonde** può essere **malcondizionata**.

Esempio 1: $n + 1$ nodi **equispaziati** in $[0, 1]$, $x_i = \frac{i}{n}$, $i = 0, \dots, n$

$n + 1$	1	2	3	4	5	6
$K_1(V)$	4	24	216	1.7067e+003	1.2500e+004	9.8784e+004

$n + 1$	7	8	9	10
$K_1(V)$	8.1271e+005	6.2915e+006	4.8184e+007	4.0042e+008

Esempio 2: $n + 1$ nodi di **Chebyshev** in $[0, 1]$, $x_i = \frac{1}{2} \cos\left(\frac{2i+1}{n+1} \frac{\pi}{2}\right) + \frac{1}{2}$

$n + 1$	1	2	3	4	5	6
$K_1(V)$	2.4142	6.9761	18.637	46.951	1.3157e+002	3.7648e+002

$n + 1$	7	8	9	10
$K_1(V)$	1.0200e+003	2.6550e+003	6.7024e+003	1.6514e+004

Nota. Poiché la matrice di Vandermonde è **malcondizionata** per n elevato e, inoltre, il **costo computazionale** per la soluzione del sistema lineare $VA = Y$ può essere **elevato**, si preferisce ricorrere ad **altre strategie** per costruire l' **unico polinomio interpolatore** p_n .

Polinomi di base di Lagrange

Base dei monomi: $\{1, x, x^2, \dots, x^{n-1}, x^n\}$

- x^k è un monomio di grado k

$$\Rightarrow p_n(x) = a_0 + a_1x + \dots + a_{n-1}x^{n-1} + a_nx^n$$

Polinomi di base di Lagrange: $\{\ell_0(x), \ell_1(x), \dots, \ell_{n-1}(x), \ell_n(x)\}$

$$\ell_k(x) := \prod_{\substack{j=0 \\ j \neq k}}^n \frac{x - x_j}{x_k - x_j} = \frac{x - x_0}{x_k - x_0} \cdot \frac{x - x_1}{x_k - x_1} \dots \frac{x - x_{k-1}}{x_k - x_{k-1}} \cdot \frac{x - x_{k+1}}{x_k - x_{k+1}} \dots \frac{x - x_n}{x_k - x_n}$$

- $\ell_k(x)$, $k = 0, \dots, n$, è un **polinomio di grado n**

$$\bullet \ell_k(x_i) = \begin{cases} 1 & \text{se } i = k \\ 0 & \text{se } i \neq k \end{cases}$$

- l'espressione di $\ell_k(x)$ dipende solo dai **nodi** $\{x_i\}$

Espressione di Lagrange del polinomio interpolatore

Il **polinomio interpolatore** sui nodi x_i si definisce come combinazione lineare dei polinomi di base di Lagrange

$$L_n(x) = \sum_{k=0}^n y_k \ell_k(x) = y_0 \ell_0(x) + y_1 \ell_1(x) + \cdots + y_{n-1} \ell_{n-1}(x) + y_n \ell_n(x)$$

dove y_k sono i valori nei nodi e $\ell_k(x)$ dipende solo dai nodi.

Da $\ell_k(x_i) = \delta_{ki}$, segue

$$L_n(x_i) = \sum_{k=0}^n y_k \ell_k(x_i) = y_i \ell_i(x_i) = y_i$$

per cui è verificata la condizione di interpolazione in ogni nodo e

$$V = \begin{bmatrix} \ell_0(x_0) & \ell_1(x_0) & \ell_2(x_0) & \cdots & \ell_n(x_0) \\ \ell_0(x_1) & \ell_1(x_1) & \ell_2(x_1) & \cdots & \ell_n(x_1) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \ell_0(x_n) & \ell_1(x_n) & \ell_2(x_n) & \cdots & \ell_n(x_n) \end{bmatrix} = I \quad \Rightarrow \quad A = Y$$

$L_n(x)$ è un polinomio di grado n che verifica le condizioni di interpolazione \Rightarrow per l'**unicità del polinomio interpolatore**, $L_n(x) \equiv p_n(x)$

Osservazioni:

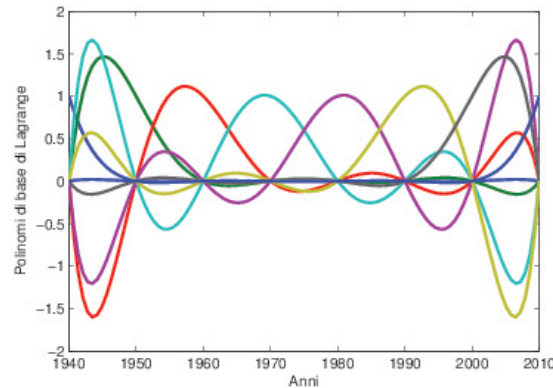
- non è necessario risolvere il sistema lineare $VA = Y$ per determinare i coefficienti a_k del polinomio interpolatore;
- i polinomi di base $\ell_k(x)$ dipendono solo dai nodi, per cui una volta fissata una distribuzione di nodi possono essere valutati una volta per tutte;
- nell'espressione del polinomio interpolatore $L_n(x)$ compaiono esplicitamente i valori di y_k con i quali si esegue l'interpolazione;
- la rappresentazione di Lagrange costituisce la base per la costruzione di **formule di quadratura** e di metodi per l'approssimazione della soluzione di problemi differenziali.

Esempio 1

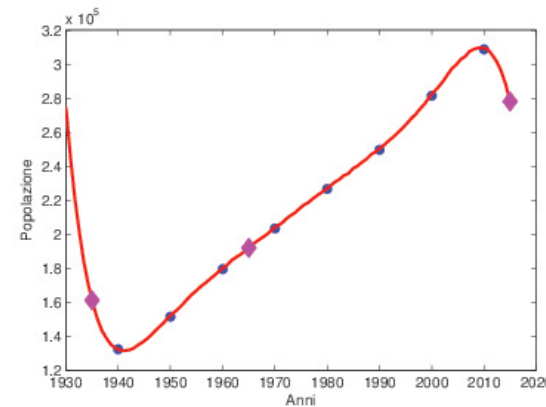
Data la tabella

Anno	1940	1950	1960	1970	1980	1990	2000	2010
Popolazione (in migliaia)	132'165	151'326	179'323	203'302	226'542	249'633	281'421	308'745

approssimare la popolazione negli anni 1930, 1965, 2015. Quanto sono **accurati** i valori ottenuti?



Polinomi di base di Lagrange



Polinomio interpolatore
(il simbolo \diamond indica i punti
di interpolazione)

Interpolazione di polinomi

L'interpolazione è **esatta** per i polinomi q_m di grado $m \leq n$.

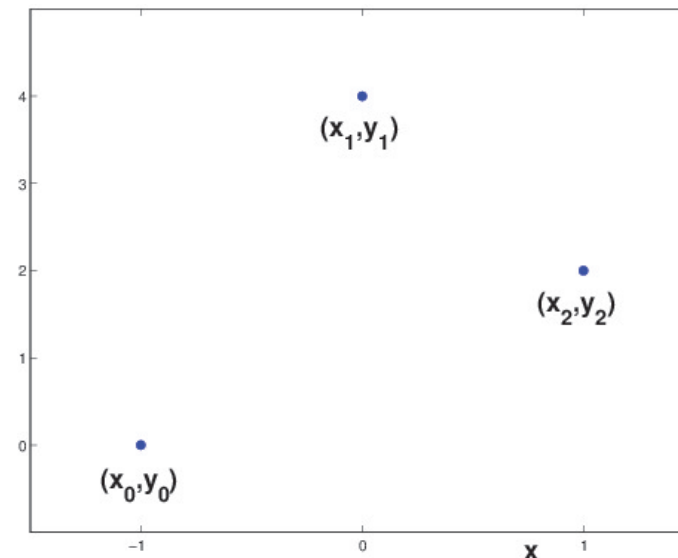
$$\{x_k, q_m(x_k)\} \quad k = 0, \dots, n \Rightarrow L_n(x) = \sum_{k=0}^n q_m(x_k) \ell_k(x) \equiv q_m(x)$$

Esempio.

Data la tavola

i	0	1	2
x_i	-1	0	1
y_i	0	4	2

costruire il relativo **polinomio interpolatore**.



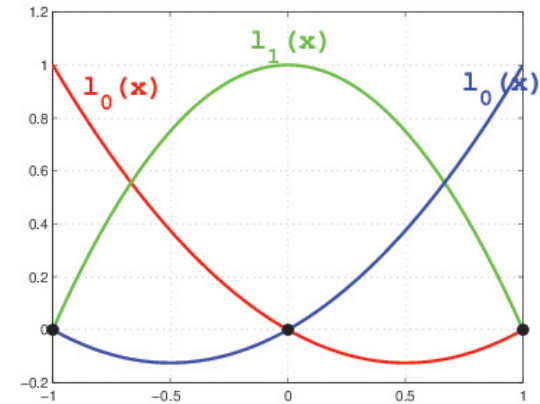
Soluzione

I dati sono $n + 1 = 3$ quindi il polinomio interpolatore è un polinomio di **grado 2**.

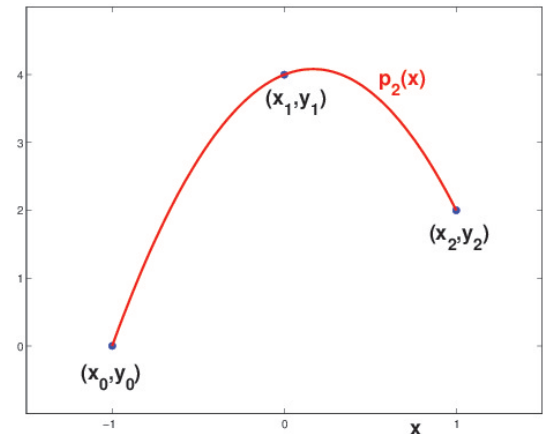
$$l_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} = \frac{x(x - 1)}{(-1)(-1 - 1)} = \frac{1}{2}x(x - 1)$$

$$l_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} = \frac{(x + 1)(x - 1)}{(1)(-1)} = -(x + 1)(x - 1)$$

$$l_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)} = \frac{(x + 1)x}{(1 + 1)(1)} = \frac{1}{2}(x + 1)x$$



$$\begin{aligned} \Rightarrow p_2(x) &= \sum_{k=0}^2 y_k l_k(x) = 0 \cdot l_0(x) + 4 \cdot l_1(x) + 2 \cdot l_2(x) \\ &= -4(x + 1)(x - 1) + (x + 1)x = -3x^2 + x + 4 \end{aligned}$$



Aggiungiamo alla tabella la coppia di valori $(x_3, y_3) = (2, -6)$ e calcoliamo il nuovo polinomio interpolatore.

i	0	1	2	3
x_i	-1	0	1	2
y_i	0	4	2	-6

⇒ il **polinomio interpolatore** è di **grado 3**

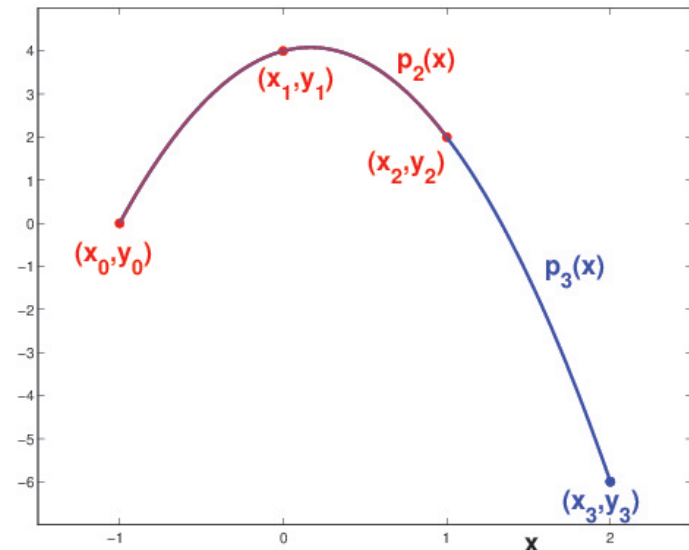
$$\Rightarrow p_3(x) = \sum_{k=0}^3 y_k l_k(x) = -3x^2 + x + 4 = p_2(x)!!!$$

$$l_0(x) = \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)} = -\frac{1}{3}x(x - 1)(x - 2)$$

$$l_1(x) = \frac{(x - x_0)(x - x_2)(x - x_3)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)} = \frac{1}{2}(x + 1)(x - 1)(x - 2)$$

$$l_2(x) = \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)} = -\frac{1}{2}(x + 1)x(x - 2)$$

$$l_3(x) = \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)} = \frac{1}{6}(x + 1)x(x - 1)$$



Se $y_i = q_m(x_i)$, dove q_m è un polinomio di grado minore o uguale a $n \Rightarrow$ esso coincide con il suo polinomio interpolatore $p_n(x)$ data la sua unicità .

Interpolazione di funzioni costanti

L'interpolazione è **esatta** per i polinomi q_m di grado $m \leq n$.

$$\text{Se } m = 0 \Rightarrow L_n(x) = \sum_{k=0}^n q_0(x_k) \ell_k(x) \equiv q_0(x) = \text{costante}$$

In particolare, se $q_0(x) \equiv 1$, allora $q_0(x_k) = 1$ per $k = 0, 1, \dots, n$

$$\Rightarrow L_n(x) = \sum_{k=0}^n \underbrace{q_0(x_k)}_1 \ell_k(x) = q_0(x) = 1$$

$$\sum_{k=0}^n \ell_k(x) = 1, \quad \forall x \in \mathbf{R}$$

Errore nell'interpolazione polinomiale

Tabella: $\{x_i, y_i\} \quad i = 0, \dots, n$

Errore di troncamento: causato dall'aver **approssimato** la funzione con il polinomio interpolatore

Errore di propagazione: dovuto agli **errori sui dati** di input (errori di misura o di arrotondamento); i dati y_i possono non rappresentare gli esatti valori assunti nei nodi dalla funzione f da ricostruire

Errore totale

polinomio "ideale"

$$\begin{aligned} E_t(x) = f(x) - p_n(x) &= \underbrace{f(x) - p_n^*(x)}_{\text{errore di troncamento}} + \underbrace{p_n^*(x) - p_n(x)}_{\text{errore di propagazione}} = \\ &= E_n(x) + E_n^*(x) \end{aligned}$$

Espressione dell'errore di troncamento

Polinomio "ideale":
$$p_n^*(x) = \sum_{k=0}^n f(x_k) \ell_k(x)$$

Errore di troncamento:
$$E_n(x) = f(x) - p_n^*(x)$$

- L'**errore di troncamento** deve essere **nullo nei nodi** poiché il polinomio interpolatore soddisfa le condizioni di interpolazione:

$$p_n^*(x_i) = f(x_i) \Rightarrow E_n(x_i) = f(x_i) - p_n^*(x_i) = 0 \quad i = 0, \dots, n$$

$$\Rightarrow E_n(x) = \prod_{i=0}^n (x - x_i) R_n(x)$$

dove $R_n(x)$ è un'opportuna funzione da individuare.

- L'**errore di troncamento** deve essere **nullo** se $f(x)$ è un **polinomio di grado $\leq n$**

Definendo il

Polinomio nodale: $\pi_n(x) := \prod_{i=0}^n (x - x_i) = (x - x_0)(x - x_1) \cdots (x - x_n)$

si trova la seguente espressione per

Errore di troncamento: $E_n(x) = \pi_n(x) \frac{f^{(n+1)}(\xi(x))}{(n+1)!} \quad \xi(x) \in [x_0, x_n]$

L'errore dipende dall'andamento del polinomio nodale e delle derivate della funzione.

Limitazioni dell'errore di troncamento - 1

Errore di troncamento:

$$E_n(x) = \pi_n(x) \frac{f^{(n+1)}(\xi(x))}{(n+1)!} \quad \xi(x) \in [a, b]$$

Il punto $\xi(x)$ in cui calcolare la derivata **non è noto** \rightarrow la formula dell'errore di troncamento ci permette di ricavare delle limitazioni

Limitazioni per eccesso e per difetto

$$m \leq f^{(n+1)}(x) \leq M \quad x \in [a, b]$$



$$m \frac{\pi_n(x)}{(n+1)!} \leq E_n(x) \leq M \frac{\pi_n(x)}{(n+1)!} \quad \text{se } \pi_n(x) > 0$$

$$m \frac{\pi_n(x)}{(n+1)!} \geq E_n(x) \geq M \frac{\pi_n(x)}{(n+1)!} \quad \text{se } \pi_n(x) < 0$$

Limitazioni dell'errore di troncamento - 2

Limitazione sul modulo

$$|f^{(n+1)}(x)| \leq \mu, \quad x \in [a, b] \quad \Rightarrow \quad |E_n(x)| \leq \frac{\mu}{(n+1)!} |\pi_n(x)|$$

Esempio: nodi equispaziati $x_i = x_0 + ih \quad (i = 0, \dots, n)$

$$\pi_n(x) = \prod_{i=0}^n (x - x_i) = (x - x_0)(x - x_1) \cdots (x - x_n)$$

Se $x = x_0 + sh \Rightarrow \pi_n(x) = s(s-1) \dots (s-n)h^{n+1}$

π_n tende a crescere agli estremi dell'intervallo. L'approssimazione migliore si ha nella parte centrale dell'intervallo di interpolazione.

I nodi migliori sono quelli che minimizzano la funzione $\frac{\mu}{(n+1)!} |\pi_n(x)|$ che risulta essere "grande" per i nodi equispaziati. I nodi che la minimizzano sono quelli di **Chebyshev**.

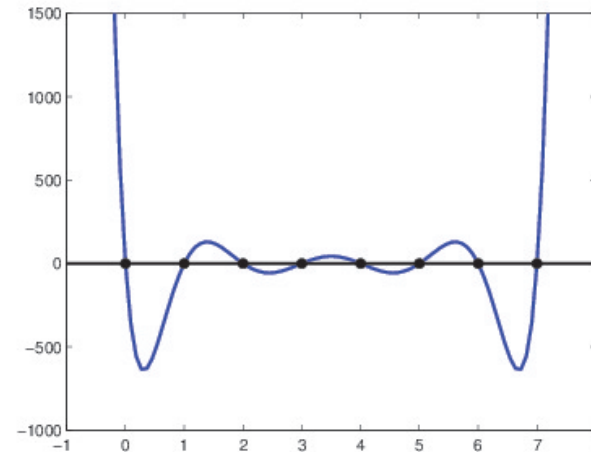


Grafico di $\pi_7(x)$

Errore di propagazione

Errore sui dati: $\begin{cases} \varepsilon_i = f(x_i) - y_i & \text{errore di arrotondamento} \\ \varepsilon_i = \delta_i & \text{errore di misura} \end{cases}$

↓

$$E_n^*(x) = p_n^*(x) - p_n(x) = \sum_{i=0}^n f(x_i) l_i(x) - \sum_{i=0}^n (f(x_i) + \varepsilon_i) l_i(x)$$

$$\Rightarrow E_n^*(x) = - \sum_{i=0}^n \varepsilon_i l_i(x)$$

Se $|\varepsilon_i| \leq \varepsilon$ si ottiene la limitazione

$$|E_n^*(x)| \leq \varepsilon \sum_{i=0}^n |l_i(x)| = \varepsilon \Lambda(x)$$

↓

Funzione di Lebesgue:

$$\Lambda(x) = \sum_{i=0}^n |l_i(x)|$$

coefficiente di amplificazione degli errori sui dati, si valuta nel punto in cui si vuole stimare l'errore di interpolazione

Costante di Lebesgue:

$$\Lambda_n = \max_{a \leq x \leq b} \Lambda(x)$$

è associata ai nodi e all'intervallo $[a, b]$ ed è una **maggiorazione dell'amplificazione degli errori sui dati**

Proprietà della funzione di Lebesgue

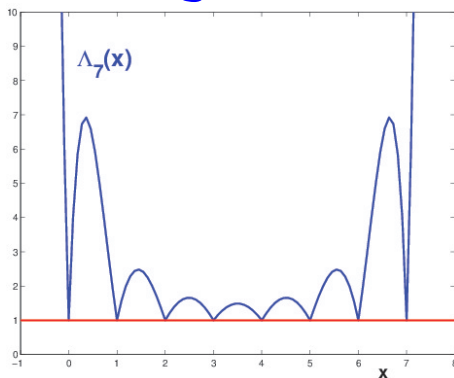
- Poiché $\sum_{i=0}^n l_i(x) = 1 \Rightarrow \Lambda(x) \geq 1$

- $\Lambda(x)$ dipende solo dai polinomi fondamentali di Lagrange e, quindi, dalla **distribuzione dei nodi**; l'**aumento del numero dei nodi** produce un aumento consistente dell'amplificazione dell'errore sui dati

Nodi equispaziati in $[a, b]$

$$x_i = a + ih \quad h = \frac{b-a}{n}, \quad i = 0, \dots, n$$

$$\Lambda_n \sim \frac{2^{n+1}}{e n \log n} \text{ per } n \rightarrow \infty$$

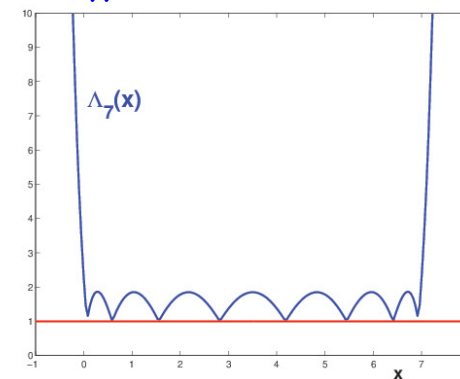


$n = 7$ Intervallo: $[0, 7]$

Nodi di Chebyshev in $[a, b]$

$$x_i = \frac{b-a}{2} \cos \frac{2i+1}{n+1} \frac{\pi}{2} + \frac{b+a}{2}, \quad i = 0, \dots, n$$

$$\Lambda_n \sim \frac{2}{\pi} \log n \text{ per } n \rightarrow \infty$$



$n = 7$ Intervallo: $[0, 7]$

Esercizio 1

Data la tabella

i	0	1	2	3
x_i	0.0	0.4	0.8	1.2
y_i	0.00000	0.42839	0.74210	0.91031

- a) scrivere l'espressione del **polinomio interpolatore di Lagrange**;
- b) sapendo che $0 < f^{(4)}(x) \leq 5.657$, dare una limitazione per eccesso e una per difetto dell'**errore di troncamento** nei punti $t_1 = 0.2$, $t_2 = 0.6$, $t_3 = 1.0$;
- c) valutare la **costante di Lebesgue** e dare una stima dell'errore di propagazione.

Soluzione

a) La tabella contiene **4 nodi**, quindi il **polinomio interpolatore** è di **grado 3**.

$$l_0(x) = \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)} = -\frac{1}{0.384}(x - 0.4)(x - 0.8)(x - 1.2)$$

$$l_1(x) = \frac{(x - x_0)(x - x_2)(x - x_3)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)} = \frac{1}{0.128}x(x - 0.8)(x - 1.2)$$

$$l_2(x) = \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)} = -\frac{1}{0.128}x(x - 0.4)(x - 1.2)$$

$$l_3(x) = \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)} = \frac{1}{0.384}x(x - 0.4)(x - 0.8)$$

$$\Rightarrow p_3(x) = \sum_{k=0}^3 y_k l_k(x) \Rightarrow \left\{ \begin{array}{l} p_3(t_0) = p_3(0.2) = \sum_{k=0}^3 y_k l_k(0.2) = 0.226606 \\ p_3(t_1) = p_3(0.6) = \sum_{k=0}^3 y_k l_k(0.6) = 0.601508 \\ p_3(t_2) = p_3(1.0) = \sum_{k=0}^3 y_k l_k(1.0) = 0.846320 \end{array} \right.$$

b) Errore di troncamento: $E_3(x) = \frac{\pi_3(x)}{4!} f^{(4)}(\tau) \quad \tau \in (0, 1.2)$

$$\pi_3(x) = (x - x_0)(x - x_1)(x - x_2)(x - x_3) = x(x - 0.4)(x - 0.8)(x - 1.2)$$

$$\begin{cases} \pi_3(t_0) = \pi_3(0.2) = -0.024 \\ \pi_3(t_1) = \pi_3(0.6) = 0.0144 \\ \pi_3(t_2) = \pi_3(1.0) = -0.024 \end{cases} \quad m = 0 < f^{(4)}(x) \leq M = 5.657$$

Stima di $E_3(t_0) \rightarrow -0.00566 \simeq \frac{\pi_3(t_0)}{4!} M \leq E_3(t_0) \leq \frac{\pi_3(t_0)}{4!} m < 0$

Stima di $E_3(t_1) \rightarrow 0 < \frac{\pi_3(t_1)}{4!} m \leq E_3(t_1) \leq \frac{\pi_3(t_1)}{4!} M \simeq 0.00339$

Stima di $E_3(t_2) \rightarrow -0.00566 \simeq \frac{\pi_3(t_2)}{4!} M \leq E_3(t_2) \leq \frac{\pi_3(t_2)}{4!} m < 0$

Nota. Le approssimazioni di t_0 e t_2 sono per **eccesso**, mentre l'approssimazione di t_1 è per **difetto**

c) **Errore di propagazione:** $|E_3^*(x)| \leq \varepsilon \sum_{i=0}^3 |l_i(x)| = \varepsilon \Lambda(x)$

Poiché i dati di input hanno **5 decimali** l'errore di arrotondamento sui dati è : $|\varepsilon_i| \leq 0.5 \cdot 10^{-5} = \varepsilon$.

$$\Lambda(t_0) = \sum_{i=0}^3 |l_i(t_0)| = 1.625 \rightarrow |E_3^*(t_1)| \leq 0.8 \cdot 10^{-5}$$

$$\Lambda(t_1) = \sum_{i=0}^3 |l_i(t_1)| = 1.25 \rightarrow |E_3^*(t_2)| \leq 0.6 \cdot 10^{-5}$$

$$\Lambda(t_2) = \sum_{i=0}^3 |l_i(t_2)| = 1.625 \rightarrow |E_3^*(x)| \leq 0.8 \cdot 10^{-5}$$

Nota. L'errore di propagazione è **trascurabile** rispetto all'errore di troncamento.

Esercizio 2

Data la seguente tabella di valori di una funzione $f(x)$

x	1	2	3	4
$f(x)$	1	-1	1	-1

2.1) valutare $f(2.5)$ mediante **due polinomi interpolatori** $p_1(x)$ e $p_2(x)$, dove $p_1(x)$ interpola $f(x)$ nei nodi 2, 3 e $p_2(x)$ nei nodi 1, 2, 3;

2.2) posto $f(x) = \cos \pi(x - 1)$ dare una maggiorazione di $|E(2.5)|$.

Soluzione

2.1) Utilizzando la rappresentazione di Lagrange per i due polinomi abbiamo

$$p_1(x) = y_2 \frac{(x-x_3)}{(x_2-x_3)} + y_3 \frac{(x-x_2)}{(x_3-x_2)} = 2x - 5$$

$$\begin{aligned} p_2(x) &= y_1 \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} + y_2 \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} + y_3 \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)} \\ &= 2x^2 - 8x + 7 \end{aligned}$$

E quindi

$$f(2.5) \cong p_1(2.5) = 0 \quad f(2.5) \cong p_2(2.5) = -0.5$$

2.2) La funzione $f \in C^\infty([1,4])$ e si può usare la stima dell'errore di troncamento

$$E_n(x) = \pi_n(x) \frac{f^{(n+1)}(t)}{(n+1)!} \quad t \in [x_0, x_n]$$

dove

$$\pi_n(x) := \prod_{i=0}^n (x - x_i) = (x - x_0)(x - x_1) \cdots (x - x_n)$$

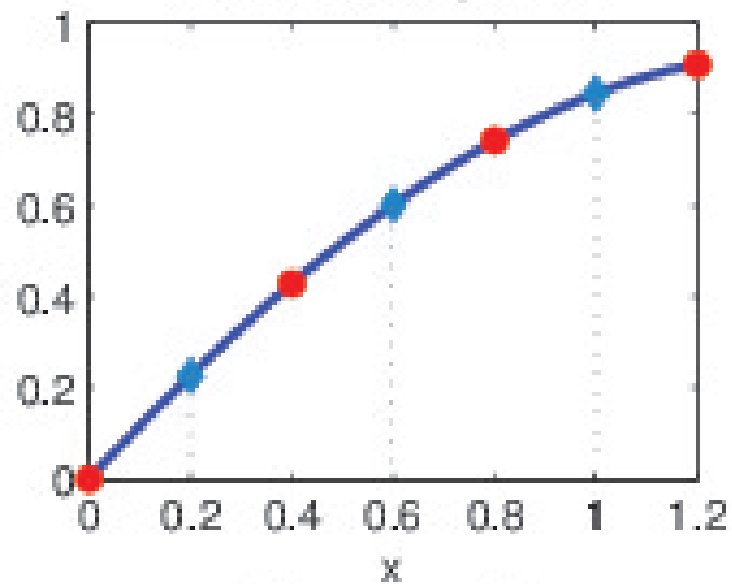
da cui

$$|E_1(2.5)| = \left| -\frac{(2.5 - 2)(2.5 - 3)}{2!} \pi^2 \cos \pi(t - 1) \right| \leq \frac{\pi^2}{8}$$

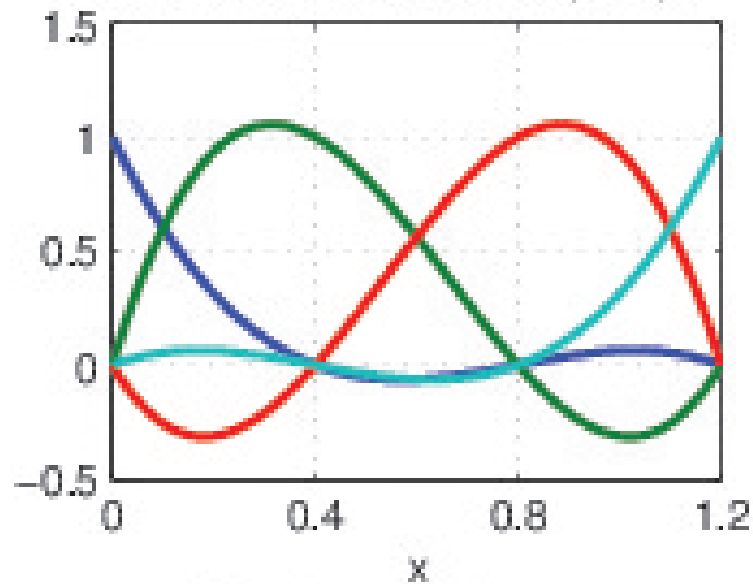
$$|E_2(2.5)| = \left| \frac{(2.5 - 1)(2.5 - 2)(2.5 - 3)}{6} \pi^3 \sin \pi(t - 1) \right| \leq \frac{\pi^3}{16}$$

Tali stime evidenziano che p_1 da una migliore approssimazione.

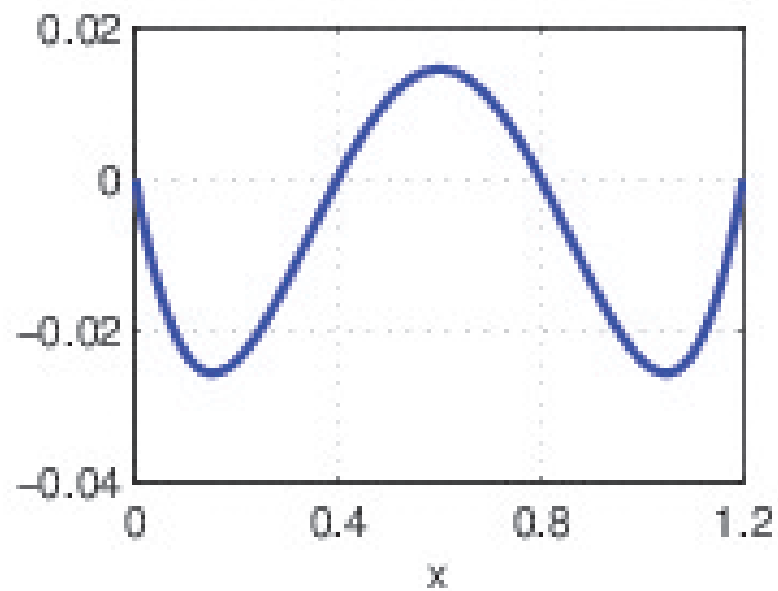
Polinomio interpolatore



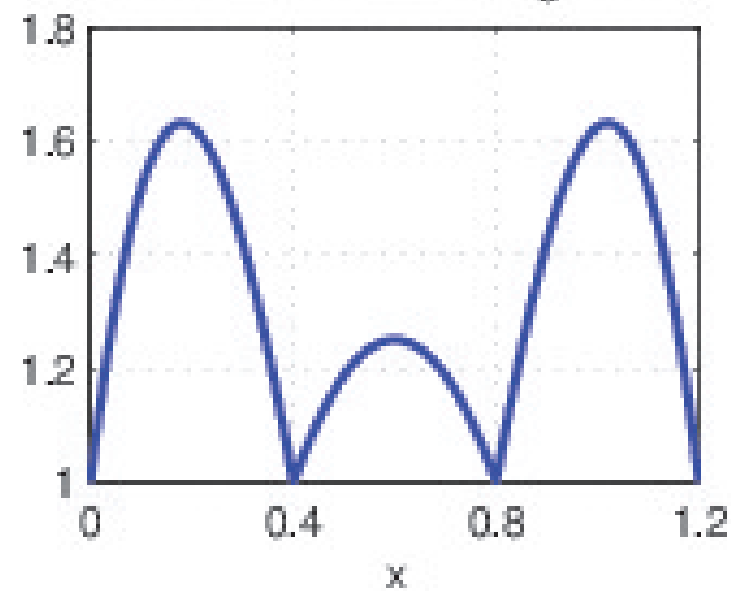
Polinomi di base di Lagrange



Polinomio nodale



Funzione di Lebesgue



Convergenza dei polinomi interpolatori

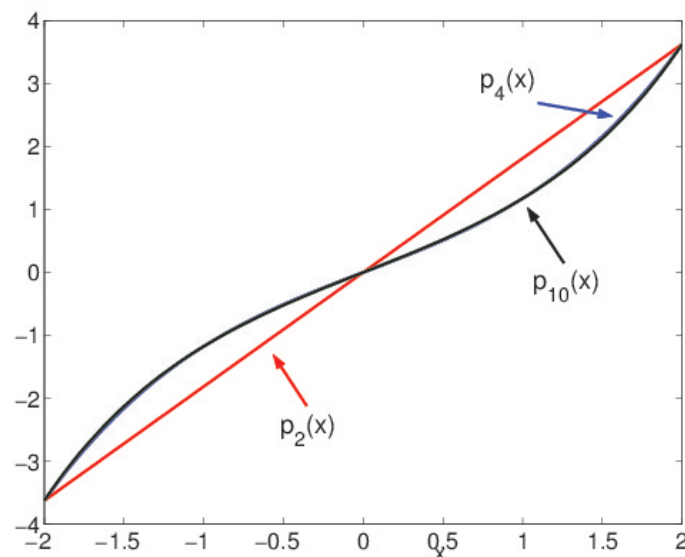
Si ha **convergenza** se, in **assenza** di **errori di arrotondamento**,

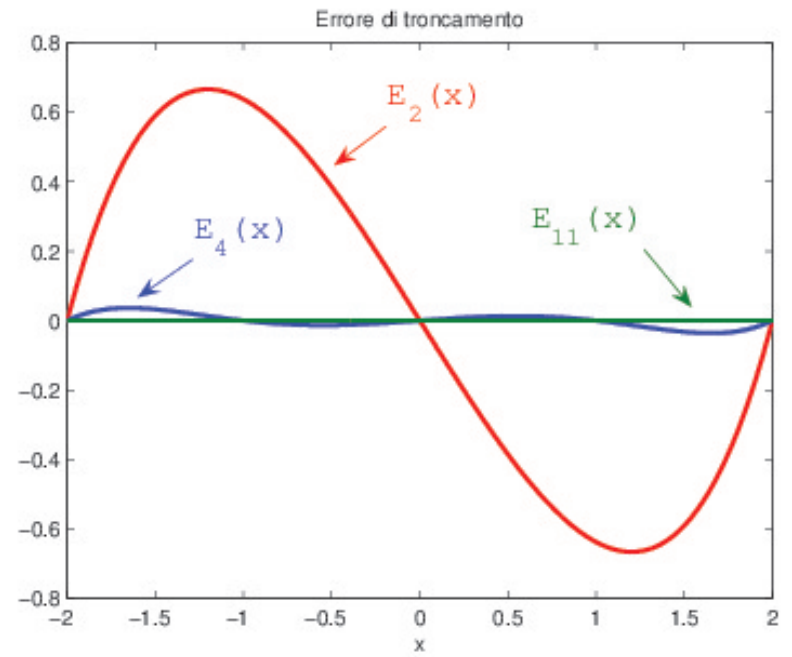
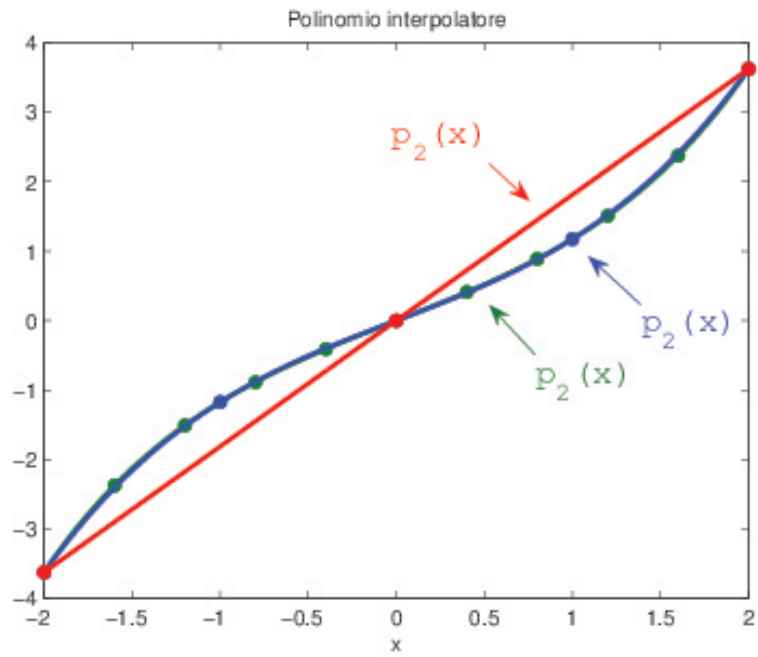
$$\lim_{n \rightarrow \infty} p_n(x) = f(x)$$

cioè se all'**umentare del numero dei nodi** il polinomio interpolatore approssima sempre **meglio** la funzione.

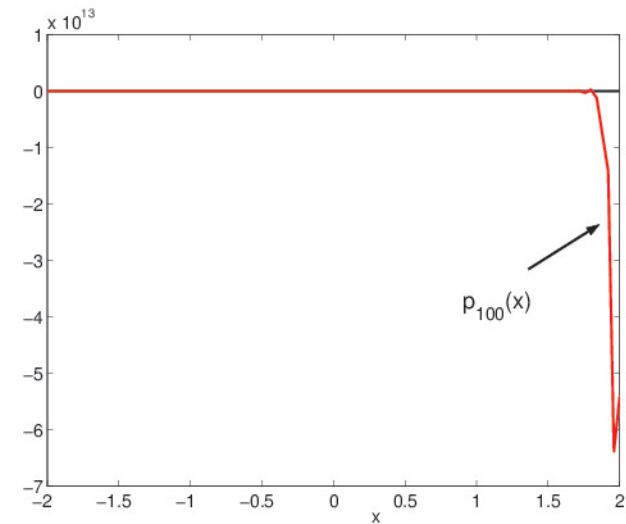
Esempio 1: $f(x) = \sinh(x)$ $x \in [-2, 2]$

All'aumentare di n il polinomio interpolatore approssima sempre meglio la funzione. Per $n = 10$ il grafico di $p_{10}(x)$ coincide con il grafico di $f(x)$.



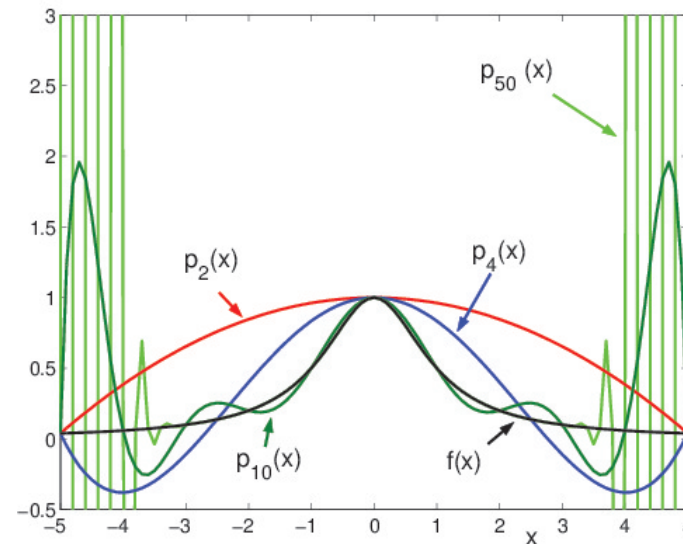


Nota. All'aumentare di n aumentano gli **errori di arrotondamento** che possono distruggere il risultato.



Esempio 2: $f(x) = \frac{1}{1+x^2}$ $x \in [-5, 5]$ (Funzione di Runge)

All'aumentare di n il polinomio interpolatore **oscilla** sempre di più in prossimità dei bordi (non dipende dagli errori di arrotondamento)



Teoremi di convergenza - 1

Teorema 1. Se $f \in C^\infty[a, b]$ e $\lim_{k \rightarrow \infty} \frac{(b-a)^k}{k!} \max_{a \leq x \leq b} |f^{(k)}(x)| = 0$, allora $\lim_{n \rightarrow \infty} p_n(x) = f(x)$ uniformemente in $[a, b]$.

Osservazioni.

- le funzioni con derivata equilimitata in $[a, b]$ soddisfano le ipotesi del teorema. Esempi: $\sin x$, $\cos x$, e^x ;
- la convergenza dei polinomi interpolatori è assicurata se i punti di singolarità della funzione da interpolare sono sufficientemente distanti dall'intervallo di interpolazione;
- il teorema non da alcuna condizione sulla distribuzione dei nodi.

Polinomi di Chebyshev

Si possono dare delle **condizioni di convergenza** con ipotesi meno restrittive su $f(x)$ se si scelgono nodi di interpolazione x_i **particolari**.

Ponendo $x = \cos \theta$ si definisce

$$T_n(x) := \cos n\theta = \cos n(\arccos x) \quad n = 0, 1, \dots$$

Ricordando che $2 \cos \theta \cos n\theta = \cos(n+1)\theta + \cos(n-1)\theta$

$$\Rightarrow \begin{cases} T_0(x) = 1 & T_1(x) = x \\ T_{n+1}(x) = 2x T_n(x) - T_{n-1}(x) & \text{(Relazione di ricorrenza)} \end{cases}$$

\Rightarrow per ogni n , $T_n(x)$ è un **polinomio** di grado n

I primi polinomi del sistema $\{T_n(x)\}$ sono

$$T_0(x) = 1 \quad T_1(x) = x \quad T_2(x) = 2x^2 - 1 \quad T_3(x) = 4x^3 - 3x$$

Nodi di Chebyshev

Per ogni n i **nodi di Chebyshev** sono gli $n + 1$ **zeri** del polinomio di Chebyshev di grado $n + 1$.

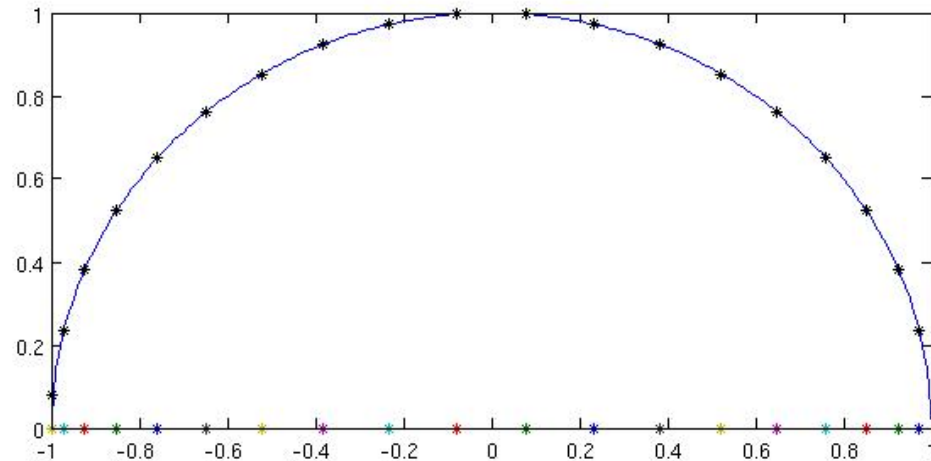
Dalla definizione di T_n si ottiene immediatamente

$$T_{n+1}(x) = 0 \iff (n + 1)\theta = (2i + 1)\frac{\pi}{2} \quad i = 0, 1, \dots, n$$

- **Nodi di Chebyshev** dell'intervallo $[-1, 1]$

$$x_i^C = \cos\left(\frac{2i + 1}{n + 1} \cdot \frac{\pi}{2}\right) \quad i = 0, 1, \dots, n$$

sono le proiezioni sull'asse x di nodi distribuiti uniformemente sul semicerchio di centro l'origine e raggio unitario. Si addensano in prossimità degli estremi dell'intervallo



- **Nodi di Chebyshev** dell'intervallo $[a, b]$

Cambiamento di coordinate: $x = \frac{b-a}{2}t + \frac{a+b}{2}$

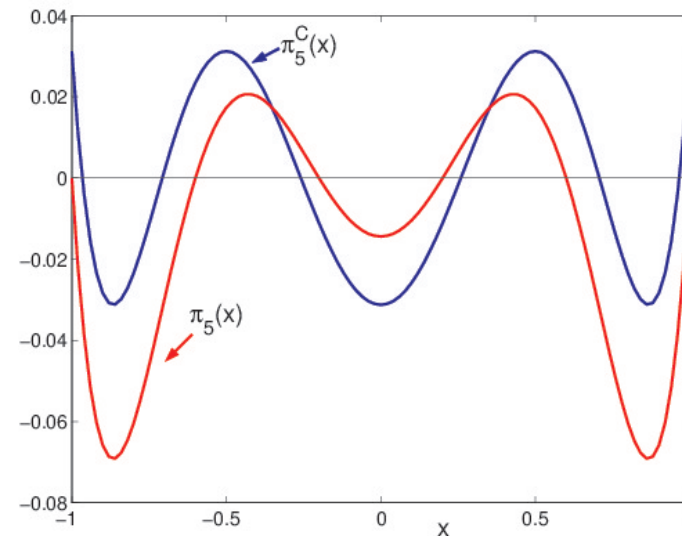
$$x_i^C = \frac{b-a}{2} \cos\left(\frac{2i+1}{n+1} \cdot \frac{\pi}{2}\right) + \frac{b+a}{2} \quad i = 0, 1, \dots, n$$

Teoremi di convergenza - 2

Teorema 3. Se f è **lipschitziana** in $[a, b]$, la successione dei polinomi interpolatori sui nodi di Chebyshev **converge** a f uniformemente in $[a, b]$.

Il **polinomio nodale** $\pi_n^C(x)$ costruito sui nodi di Chebyshev ha le seguenti proprietà :

- $\max_{x \in [a, b]} |\pi_n^C(x)| = \frac{(b-a)^{n+1}}{2^{2n+1}}$
- $\max_{x \in [a, b]} |\pi_n(x)| > \max_{x \in [a, b]} |\pi_n^C(x)|$



Nota. I **nodi di Chebyshev** sono tutti **interni** all'intervallo $[a, b]$

$$a < x_i^C < b \quad i = 0, 1, \dots, n$$

Svantaggi dell'espressione di Lagrange del polinomio interpolatore

- L'espressione di **ciascun polinomio di base di Lagrange** $l_k(x)$ dipende da **tutti** i nodi x_i .

Esempio: $\{x_0, x_1, x_2\}$

$$l_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)} \quad l_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)} \quad l_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}$$

- Se si **aggiunge un nodo** bisogna **ricalcolare** tutti i polinomi l_k e, quindi, p_n .

Esempio: $\{x_0, x_1, x_2, x_3\}$

$$l_0(x) = \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)} \quad l_1(x) = \frac{(x - x_0)(x - x_2)(x - x_3)}{(x_1 - x_0)(x_1 - x_2)(x_1 - x_3)}$$
$$l_2(x) = \frac{(x - x_0)(x - x_1)(x - x_3)}{(x_2 - x_0)(x_2 - x_1)(x_2 - x_3)} \quad l_3(x) = \frac{(x - x_0)(x - x_1)(x - x_2)}{(x_3 - x_0)(x_3 - x_1)(x_3 - x_2)}$$

Un'**espressione** del polinomio interpolatore p_n più efficiente è data dalla **formula di Newton alle differenze divise**.

Differenze divise

Tabella: $\{x_i, f_i\}$ $i = 0, 1, \dots, n$ (**Nodi distinti**)

Differenze divise di **ordine zero**: $f[x_i] := f_i$ $i = 0, 1, \dots, n$

Differenze divise prime: $f[x_i, x_j] = \frac{f[x_i] - f[x_j]}{x_i - x_j}$ $i, j = 0, 1, \dots, n$ $i \neq j$

Differenze divise seconde:

$$f[x_i, x_j, x_k] = \frac{f[x_i, x_j] - f[x_j, x_k]}{x_i - x_k} \quad i, j, k = 0, 1, \dots, n \quad i \neq j \neq k \neq i$$

La **differenze divisa** k -esima relativa a $k+1$ **nodi distinti** x_0, x_1, \dots, x_k è definita da

$$f[x_0, x_1, \dots, x_k] = \frac{f[x_0, x_1, \dots, x_{k-1}] - f[x_1, x_2, \dots, x_k]}{x_0 - x_k}$$

Tavola delle differenze divise

- Per calcolare la **differenza divisa** k -esima servono $k + 1$ valori della funzione
- Con $n + 1$ **nodi** si possono costruire $n - k + 1$ **differenze divise** k -esime, e pertanto **una sola** differenza divisa n -esima

x_0	f_0				
		$f[x_0, x_1]$			
x_1	f_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_2		$f[x_1, x_2, x_3]$		$f[x_0, x_1, x_2, x_3, x_4]$
		$f[x_2, x_3]$		$f[x_1, x_2, x_3, x_4]$	
x_3	f_3		$f[x_2, x_3, x_4]$		
		$f[x_3, x_4]$			
x_4	f_4				

Tavola delle differenze divise

- Se si **aggiunge** un nodo la **tavola** alle differenze divise viene modificata solo per l'**aggiunta** di una nuova riga

x_0	f_0					
		$f[x_0, x_1]$				
x_1	f_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_2		$f[x_1, x_2, x_3]$		$f[x_0, x_1, x_2, x_3, x_4]$	
		$f[x_2, x_3]$		$f[x_1, x_2, x_3, x_4]$		$f[x_0, x_1, x_2, x_3, x_4, x_5]$
x_3	f_3		$f[x_2, x_3, x_4]$		$f[x_1, x_2, x_3, x_4, x_5]$	
		$f[x_3, x_4]$		$f[x_2, x_3, x_4, x_5]$		
x_4	f_4		$f[x_3, x_4, x_5]$			
		$f[x_4, x_5]$				
x_5	f_5					

E' necessario calcolare solo una differenza divisa per ogni ordine.

Forma di Newton del polinomio interpolatore

Dalla definizione di **differenze divise** si ha

$$f[x] = f[x_0] + (x - x_0)f[x, x_0]$$

$$f[x, x_0] = f[x_0, x_1] + (x - x_1)f[x, x_0, x_1]$$

.....

$$f[x, x_0, \dots, x_{n-2}] = f[x_0, x_1, \dots, x_{n-1}] + (x - x_{n-1})f[x, x_0, \dots, x_{n-1}]$$

$$f[x, x_0, \dots, x_{n-1}] = f[x_0, x_1, \dots, x_n] + (x - x_n)f[x, x_0, \dots, x_n]$$

Sostituendo ciascuna di queste relazioni nella precedente si ottiene

$$\begin{aligned} f(x) = & f[x_0] + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2] + \\ & + \dots + (x - x_0)(x - x_1) \cdots (x - x_{n-1})f[x_0, x_1, \dots, x_n] + \\ & + (x - x_0)(x - x_1) \cdots (x - x_n)f[x, x_0, x_1, \dots, x_n] \end{aligned}$$

I primi $n + 1$ termini costituiscono un polinomio di grado n i cui coefficienti dipendono da f . Indicando con

$$P_n(f; x) = f[x_0] + (x - x_0)f[x_0, x_1] + (x - x_0)(x - x_1)f[x_0, x_1, x_2] + \dots + (x - x_0)(x - x_1) \cdots (x - x_{n-1})f[x_0, x_1, \dots, x_n]$$

$$f(x) = P_n(f, x) + \pi_n(x) f[x, x_0, x_1, \dots, x_n, x_{n+1}]$$

Teorema

Il **polinomio di Newton alle differenze divise** $P_n(f; x)$ coincide con il **polinomio interpolatore** $p_n(x)$ di $f(x)$ nei nodi x_0, x_1, \dots, x_n

$P_n(f; x)$ è l'espressione del **polinomio interpolatore** nella base dei **polinomi nodali**

$$\{1, (x - x_0), (x - x_0)(x - x_1), \dots, (x - x_0)(x - x_1) \cdots (x - x_{n-1})\}$$

Errore di troncamento

$$E_n(x) = \pi_n(x) f[x, x_0, x_1, \dots, x_n, x_{n+1}]$$

Espressione dell'errore di troncamento

$$E_n(x) = f(x) - p_n^*(x) = f(x) - P_n(x) =$$

$$= \pi_n(x) f[x, x_0, x_1, \dots, x_n] = \pi_n(x) \frac{f^{(n+1)}(\xi(x))}{(n+1)!}$$

$$\Rightarrow \text{Se } f \in C^k[x_0, x_k] \Rightarrow f[x_0, x_1, \dots, x_k] = \frac{f^{(k)}(t_k)}{k!} \quad t_k \in [x_0, x_k]$$

Stima dell'errore di troncamento

$$\text{Se } f^{(n+1)}(x) \text{ esiste } \Rightarrow f[x_0, x_1, \dots, x_{n+1}] = \frac{f^{(n+1)}(t_{n+1})}{(n+1)!} \quad t_{n+1} \in [x_0, x_{n+1}]$$

$$\text{Se } f^{(n+1)}(x) \text{ varia poco in } [x_0, x_n] \Rightarrow f^{(n+1)}(\xi) \simeq f^{(n+1)}(t_{n+1})$$

$$E_n(x) = \pi_n f[x, x_0, x_1, \dots, x_n] =$$

$$= \pi_n(x) \frac{f^{(n+1)}(\xi)}{(n+1)!} \simeq \pi_n(x) \frac{f^{(n+1)}(t_{n+1})}{(n+1)!}$$

$$= \pi_n(x) f[x_0, x_1, \dots, x_n, x_{n+1}]$$

$$\left. \begin{array}{l} \\ \\ \end{array} \right\} \Rightarrow E_n(x) \simeq \pi_n(x) f[x_0, x_1, \dots, x_n, x_{n+1}]$$

Propagazione degli errori nella tavola alle differenze

Il calcolo delle differenze divise è **molto sensibile** agli **errori sui dati**

Esempio: per nodi equidistanti con passo h

x_0	f_0				
		$f[x_0, x_1]$			
x_1	f_1		$f[x_0, x_1, x_2] + \frac{\varepsilon}{2h^2}$		
		$f[x_2, x_1] + \frac{\varepsilon}{h}$		$f[x_0, x_1, x_2, x_3] - \frac{\varepsilon}{2h^3}$	
x_2	$f_2 + \varepsilon$		$f[x_1, x_2, x_3] - \frac{\varepsilon}{h^2}$		$f[x_0, x_1, x_2, x_3, x_4] + \frac{\varepsilon}{4h^4}$
		$f[x_3, x_2] - \frac{\varepsilon}{h}$		$f[x_1, x_2, x_3, x_4] + \frac{\varepsilon}{2h^3}$	
x_3	f_3		$f[x_2, x_3, x_4] + \frac{\varepsilon}{2h^2}$		
		$f[x_4, x_3]$			
x_4	f_4				

- L'**amplificazione** della perturbazione nel calcolo delle differenze divise è tanto **maggiore** quanto maggiore è l'**ordine** della differenza divisa che si vuole calcolare
- Poiché le differenze divise al crescere dell'ordine hanno **valori** via via più **piccoli**, si possono verificare fenomeni di **cancellazione numerica**

Approssimazione ai minimi quadrati

Problema.

Data la **tabella** $\{x_i, y_i\}$, $i = 0, 1, \dots, n$, si vuole trovare una **funzione analitica** φ_M che **approssimi** i dati.

In questo caso la **tabella** è il risultato di **misure sperimentali** ciascuna delle quali è affetta da un **errore di misura** ε_i .

Metodo di approssimazione: si sceglie la funzione approssimante φ_M in modo da **minimizzare**

$$\sum_{i=0}^n [\varphi_M(x_i) - y_i]^2 \quad \text{Scarto quadratico}$$

oppure, introducendo i **pesi** $w_i > 0$, $\forall i$,

$$\sum_{i=0}^n w_i [\varphi_M(x_i) - y_i]^2 \quad \text{Scarto quadratico pesato}$$

Caso lineare

Funzione approssimante:

$\varphi_M(x)$ dipende **linearmente** da M **parametri**:

$$a_0, a_1, \dots, a_M \quad M \ll n$$

$$\Rightarrow \varphi_M(x) = a_0\psi_0(x) + a_1\psi_1(x) + \dots + a_M\psi_M(x)$$

dove $\{\psi_k(x)\}_{k=0,\dots,M}$ è una base per lo spazio di approssimazione

Metodo di approssimazione: si minimizza lo **scarto quadratico**

$$\sigma^2(a_0, a_1, \dots, a_M) = \sum_{i=0}^n \left[\underbrace{a_0\psi_0(x_i) + a_1\psi_1(x_i) + \dots + a_M\psi_M(x_i)}_{\varphi_M(x_i)} - y_i \right]^2$$

Risolvere il problema dell'**approssimazione ai minimi quadrati** vuol dire individuare i **coefficienti reali** a_k che rendono **minimo** $\sigma^2(a_0, \dots, a_M)$.

Nota. L'approssimante ai minimi quadrati in generale **non passa** per i valori $\{x_i, y_i\}$ ma "*vicino*" ad essi.

Polinomio algebrico ai minimi quadrati

Tabella: $\{x_i, y_i\}$ $i = 0, 1, \dots, n$

Funzione approssimante:

$$P_M(x) = a_0 + a_1x + \dots + a_{M-1}x^{M-1} + a_Mx^M \quad M \ll n$$

Metodo di approssimazione: si minimizza lo **scarto quadratico**

$$\sigma^2(a_0, a_1, \dots, a_M) = \sum_{i=0}^n \underbrace{[a_0 + a_1x_i + \dots + a_{M-1}x_i^{M-1} + a_Mx_i^M - y_i]^2}_{P_M(x_i)}$$

Funzioni di base:

$$\psi_0(x) = 1, \quad \psi_1(x) = x, \quad \dots, \quad \psi_k(x) = x^k, \quad \dots, \quad \psi_M(x) = x^M$$

Risolvere il problema dell'**approssimazione polinomiale ai minimi quadrati** vuol dire individuare il polinomio P_M , cioè gli $M+1$ coefficienti reali a_k , per il quale σ è **minimo**. Il polinomio che si determina non passa per i punti $\{x_i, y_i\}$ ma “vicino” ad essi e tra tutti i polinomi dello stesso grado è quello che minimizza lo scarto quadratico σ^2 .

Osservazione

Imponendo le condizioni di interpolazione

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

per determinare i coefficienti di P_M si ottiene il **sistema lineare**

$$VA = Y, \quad V \in \mathbf{R}^{n \times M}, \quad A \in \mathbf{R}^M, \quad Y \in \mathbf{R}^n$$

che risulta essere **sovradeterminato** ($M \ll n$) e quindi non risolvibile in senso classico, ma che può essere risolto mediante i **minimi quadrati**.

Nota. Un esempio tipico di problema che conduce ad un sistema sovradeterminato è l'approssimazione di dati sperimentali.

Minimizzazione di σ

- Per **minimizzare** σ bisogna annullare il **gradiente**

$$\Rightarrow \frac{\partial \sigma^2}{\partial a_k} = \frac{\partial}{\partial a_k} \sum_{i=0}^n [P_M(x_i) - y_i]^2 = 0 \quad k = 0, 1, \dots, M$$

$$\Rightarrow 2 \sum_{i=0}^n (P_M(x_i) - y_i) \frac{\partial P_M(x_i)}{\partial a_k} = 0 \quad k = 0, 1, \dots, M$$

$$\Rightarrow 2 \sum_{i=0}^n (a_0 + a_1 x_i + \dots + a_{M-1} x_i^{M-1} + a_M x_i^M - y_i) x_i^k = 0, \quad k = 0, 1, \dots, M$$

$$\Rightarrow a_0 \sum_{i=0}^n x_i^k + a_1 \sum_{i=0}^n x_i^{k+1} + \dots + a_M \sum_{i=0}^n x_i^{M+k} = \sum_{i=0}^n y_i x_i^k, \quad k = 0, \dots, M$$

- Per trovare gli $M + 1$ coefficienti incogniti a_k bisogna risolvere il sistema lineare ottenuto (**sistema delle equazioni normali**).
- Per verificare che la soluzione del sistema sia un **minimo** bisogna studiare l'**hessiano**

$$\left[\frac{\partial^2 \sigma^2}{\partial a_j \partial a_k} \right] = 2 \left[\sum_{i=0}^n x_i^{k+j} \right]_{j,k=0,\dots,M} = 2H$$

Sistema delle equazioni normali

Definizioni: $s_k := \sum_{i=0}^n x_i^k$ $v_k := \sum_{i=0}^n y_i x_i^k$ $k = 0, 1, \dots, M$

Il sistema delle $M + 1$ equazioni normali nelle incognite a_k diventa

$$\left\{ \begin{array}{l} s_0 a_0 + s_1 a_1 + \dots + s_M a_M = v_0 \\ s_1 a_0 + s_2 a_1 + \dots + s_{M+1} a_M = v_1 \\ \dots \\ s_M a_0 + s_{M+1} a_1 + \dots + s_{2M} a_M = v_M \end{array} \right. \iff HA = B$$

$$A = [a_0, a_1, \dots, a_M]^T$$

$$B = [v_0, v_1, \dots, v_M]^T$$

$$H = \begin{bmatrix} s_0 & s_1 & \dots & s_M \\ s_1 & s_2 & \dots & s_{M+1} \\ \dots & \dots & \dots & \dots \\ s_M & s_{M+1} & \dots & s_{2M} \end{bmatrix} \in \mathbf{R}^{(M+1) \times (M+1)}$$

Unicità della soluzione

Definiamo il vettore $Y = [y_0, y_1, \dots, y_n]^T \Rightarrow B = V^T Y \quad H = V^T V$

$$\text{dove } V = \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^M \\ 1 & x_1 & x_1^2 & \cdots & x_1^M \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & x_n & x_n^2 & \cdots & x_n^M \end{bmatrix} \in \mathbf{R}^{(n+1) \times (M+1)}$$

V è la matrice di Vandermonde dei nodi $\{x_i\}$.

\Rightarrow la matrice V ha rango $M + 1$ ed è **regolare**

\Rightarrow Per ogni $X \in \mathbf{R}^{M+1}$ si ha $X^T H X = (V X)^T (V X) = \|V X\|_2^2 \geq 0$.
Inoltre, per la regolarità di V , l'uguaglianza vale se e solo se $X = 0$

$\Rightarrow H$ è **definita positiva** e quindi **regolare** \Rightarrow il sistema delle equazioni normali ammette un' **unica soluzione** $\bar{A} = H^{-1} B$

La matrice hessiana $2H$ è **definita positiva** \Rightarrow la soluzione \bar{A} del sistema delle equazioni normali corrisponde a un **minimo**.

$\bar{A} = (V^T V)^{-1} V^T Y$ è il vettore dei coefficienti del polinomio $P_M(x)$ che ricostruisce i dati nel senso dei minimi quadrati.

Nota. La matrice H è **malcondizionata**. L'approssimazione polinomiale ai minimi quadrati è dunque un problema mal condizionato \Rightarrow **Metodi di regolarizzazione**.

Retta di regressione

Un caso di particolare interesse nelle applicazioni è la costruzione della **retta di regressione**, ossia il **polinomio di grado 1**

$$P_1(x) = a_0 + a_1x$$

che approssima i dati $\{x_i, y_i\}$, $i = 0, 1, \dots, n$, ($n \gg 1$) nel senso dei **minimi quadrati**.

Equazioni normali \Rightarrow **Soluzione**

$$\begin{cases} a_0s_0 + a_1s_1 = v_0 \\ a_0s_1 + a_1s_2 = v_1 \end{cases} \Rightarrow \begin{cases} a_0 = \frac{v_0s_2 - v_1s_1}{s_0s_2 - s_1^2} \\ a_1 = \frac{s_0v_1 - s_1v_0}{s_0s_2 - s_1^2} \end{cases}$$

dove

$$s_0 = n + 1 \quad s_1 = \sum_{i=0}^n x_i \quad s_2 = \sum_{i=0}^n x_i^2$$

$$v_0 = \sum_{i=0}^n y_i \quad v_1 = \sum_{i=0}^n y_i x_i$$

Esempio 2

La forza $F(x)$ necessaria per allungare una molla fino alla lunghezza x è data da $F(x) = k(x - l)$ (**Legge di Hooke**) dove k è la **costante elastica** e l è la **lunghezza a riposo** della molla.

Nella tabella sono riportate le **misure sperimentali** relative alla **forza** $F(x)$ necessaria per allungare una **molla** fino alla **lunghezza** x .

i	0	1	2	3	4	5	6
x_i	2	3	4	5	6	8	10
$F(x_i)$	7.0	8.3	9.4	11.3	12.3	14.4	15.9

Determinare la **costante elastica** della molla.

Esempio 2: soluzione

Per trovare la costante elastica della molla si devono approssimare i dati in tabella con il polinomio ai minimi quadrati di primo grado (**retta di regressione**):

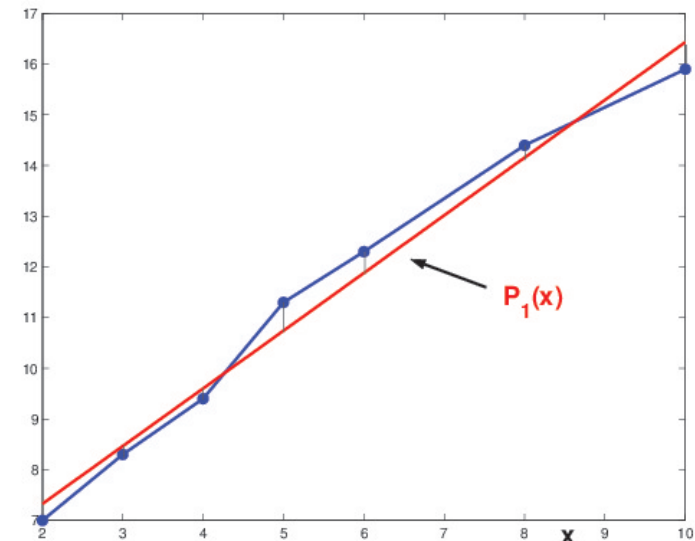
$$P_1(x) = a_0 + a_1 x$$

Il coefficiente a_1 fornisce l'approssimazione della **costante elastica**. Risolvendo il sistema delle equazioni normali si ottiene

$$a_0 = 5.049 \quad a_1 = 1.1383.$$

Per questi valori dei coefficienti si ha

$$\begin{aligned} \sigma^2(a_0, a_1) &= \sum_{i=0}^6 [a_0 + a_1 x_i - y_i]^2 = \\ &= 1.0071 \end{aligned}$$



Interpretazione probabilistica - 1

Siano x una **variabile deterministica** e $y = a_0 + a_1x$ la **variabile dipendente**, legata a x da una **relazione lineare**. Per ogni coppia $\{x_i, y_i\}$ valgono le relazioni $y_i = a_0 + a_1x_i$, $i = 0, \dots, n$.
Supporremo che i **dati** $\{x_i, y_i + \varepsilon_i\}$ siano affetti da **rumore** con **errore statistico** ε_i .

Definizione. A partire dai dati $\{x_i, y_i\}$ si definiscono la **varianza** e la **covarianza** rispettivamente come

$$\text{var}(x) = \frac{1}{n+1} \sum_i (x_i - \bar{x})^2 \quad \text{cov}(x, y) = \frac{1}{n+1} \sum_i (x_i - \bar{x})(y_i - \bar{y})$$

dove

$$\bar{x} = \frac{1}{n+1} \sum_i x_i \quad \bar{y} = \frac{1}{n+1} \sum_i y_i$$

sono le **medie osservate**.

Interpretazione probabilistica - 2

Ipotesi:

- 1) x è una **variabile deterministica**
- 2) $E(\varepsilon_i) = 0$ (**valore atteso**)
- 3) $var(\varepsilon_i)$ **costante** per ogni i
- 4) $cov(\varepsilon_i, \varepsilon_j) = 0$ per ogni $i \neq j$

Nel metodo dei **minimi quadrati** si minimizza la quantità

$$\sigma^2(a_0, a_1) = \sum_i (a_0 + a_1 x_i - y_i)^2 = \sum_i \varepsilon_i^2$$

Se valgono **Hp. 1-4** i coefficienti a_0 e a_1 , soluzione del problema di minimo, possono essere scritti come

$$a_1 = \frac{(n+1)(\sum_i x_i y_i) - (\sum_i x_i)(\sum_i y_i)}{(n+1)(\sum_i x_i^2) - (\sum_i x_i)^2} = \frac{cov(x, y)}{var(x)}$$

$$a_0 = \frac{(\sum_i y_i)(\sum_i x_i^2) - (\sum_i x_i)(\sum_i x_i y_i)}{(n+1)(\sum_i x_i^2) - (\sum_i x_i)^2} = \bar{y} - a_1 \bar{x} = \bar{y} - \frac{cov(x, y)}{var(x)} \bar{x}$$

Esercizio 3

Determinare la **retta di regressione** relativa ai dati in tabella

x_i	-1	0	1	2	3
y_i	-1.07	1.92	4.85	8.18	11.05

Soluzione

La retta di regressione ha equazione

$$y = a_1 x + a_0$$

dove

$$a_0 = \frac{v_0 s_2 - v_1 s_1}{s_0 s_2 - s_1^2}, \quad a_1 = \frac{s_0 v_1 - s_1 v_0}{s_0 s_2 - s_1^2}$$

$$s_k = \sum_{i=0}^n x_i^k, \quad v_k = \sum_{i=0}^n y_i x_i^k$$

Con $n = 4$ si ottiene

$$s_0 = 5, \quad s_1 = 5, \quad s_2 = 15$$

$$v_0 = 24.93, \quad v_1 = 55.43, \quad v_2 = 135.95$$

$$\Rightarrow a_1 = \frac{5 \cdot 55.43 - 5 \cdot 24.93}{5 \cdot 15 - 5^2} = 3.05$$

$$\Rightarrow a_0 = \frac{15 \cdot 24.93 - 5 \cdot 55.43}{5 \cdot 15 - 5^2} = 1.936$$

per cui la retta di regressione è

$$y = 3.05x + 1.936$$

Esercizio 4

Trovare il **polinomio di secondo grado** che meglio approssima i dati in tabella

x_i	0	0.5	1	1.5	2	2.5
y_i	0	0.25	1	2.25	4	6.25

Soluzione

I coefficienti del polinomio $p_2(x) = a_0 + a_1x + a_2x^2$ si ottengono risolvendo il sistema

$$\begin{pmatrix} s_0 & s_1 & s_2 \\ s_1 & s_2 & s_3 \\ s_2 & s_3 & s_4 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} v_0 \\ v_1 \\ v_2 \end{pmatrix}$$

dove

$$s_k = \sum_{i=0}^5 x_i^k, \quad v_k = \sum_{i=0}^5 y_i x_i^k$$

Si deve quindi risolvere il seguente sistema

$$\begin{pmatrix} 6 & 7.5 & 13.75 \\ 7.5 & 13.75 & 28.125 \\ 13.75 & 28.125 & 61.1875 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 13.75 \\ 28.125 \\ 61.1875 \end{pmatrix}$$

la cui soluzione è $[a_0 \ a_1 \ a_2]^T = [0 \ 0 \ 1]^T \Rightarrow p_2(x) = x^2$

Riferimenti bibliografici

L. Gori, *Calcolo Numerico*:

Cap. 6 §§ **6.1-6.5, 6.10-6.13**

L. Gori, M.L. Lo Cascio, F. Pitoli, *Esercizi di Calcolo Numerico*:

Cap. 3: Es. **3.1, 3.3-3.6, 3.8-3.9, 3.20-3.21**

Esercizi d'esame: Es. **7.2, 7.37, 7.50, 7.70, 7.79, 7.83**