

Metodi Matematici per l'Ingegneria

(A.A. 2019-2020)

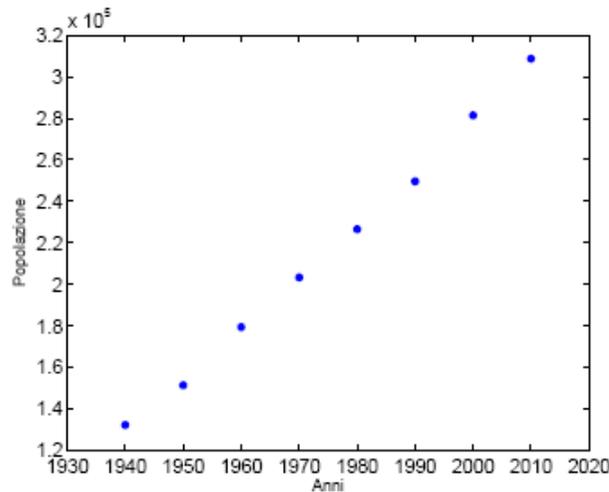
Metodi Numerici

Approssimazione di dati e funzioni

Esempio 1 (Demografia)

Nella **tavola** seguente è riportata la **popolazione** (in migliaia) degli Stati Uniti censita ogni **10 anni** tra il **1940** e il **2010**.

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



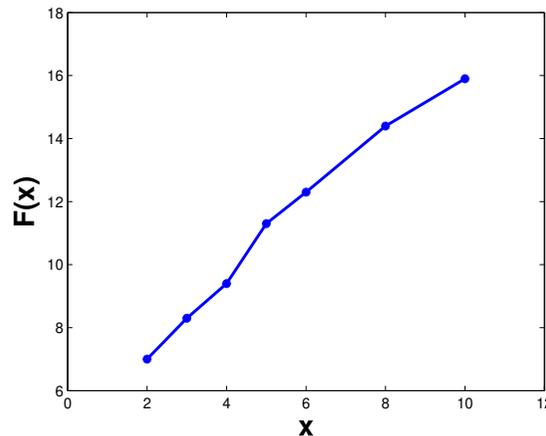
Approssimare la popolazione negli anni **1935**, **1965**, **2015**. Quanto sono **accurati** i valori ottenuti?

Esempio 2 (Fisica)

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.



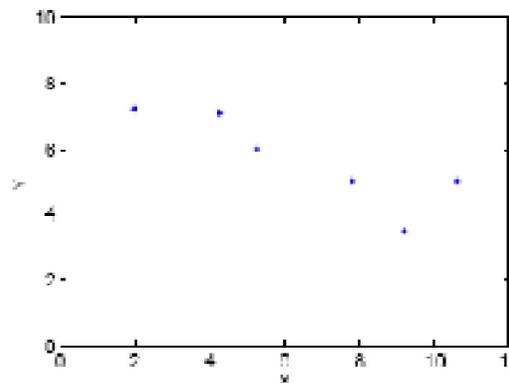
Esempio 3 (Robotica)

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 (Ing. Areospaziale) Localizzazione mediante GPS

Il **Global Positioning System (GPS)**, è un sistema di posizionamento su base satellitare, a copertura globale e continua, gestito dal dipartimento della difesa statunitense.

In orbita ci sono un minimo di **24 satelliti** in orbita ad una distanza 20200 km, più 3 di scorta.

Tenendo conto della configurazione geometrica della terra, da un punto del globo terrestre il ricevitore riesce a vedere solo la metà di essi. Di questi, ne vede solo una parte a causa della loro inclinazione rispetto all' equatore.

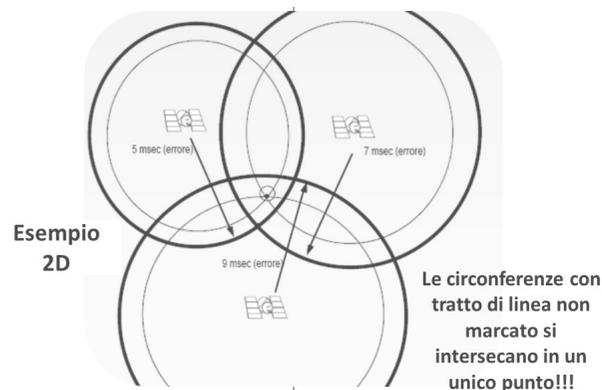
In genere, il ricevitore GPS "seleziona" ulteriormente i satelliti in base alla loro geometria e alla stima degli errori su ciascuno, privilegiando quelli che forniscono maggior precisione.

I satelliti infatti sono disposti su 6 piani orbitali inclinati di 55° rispetto al piano equatoriale (quindi non coprono le zone polari) a forma di ellissi a bassa eccentricità

Ogni piano orbitale ha 3 o 4 satelliti, e i piani sono disposti in modo tale che **ogni utilizzatore sulla terra possa ricevere i segnali di almeno 5 satelliti.**

La localizzazione si basa su un **metodo di posizionamento sferico**, che **misura il tempo impiegato da un segnale radio a percorrere la distanza satellite-ricevitore**.

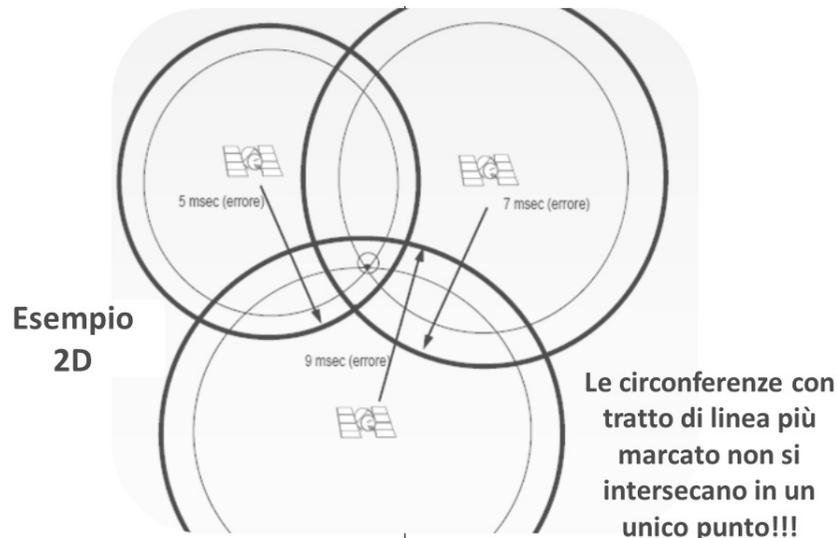
Per il calcolo della differenza dei tempi, il segnale inviato dal satellite è di tipo orario, grazie all'orologio presente sul satellite, e il ricevitore calcola l'esatta distanza di propagazione dal satellite a partire dalla differenza (dell'ordine dei microsecondi) dell'orario pervenuto con quello del proprio orologio perfettamente sincronizzato.



Conoscendo il tempo impiegato dal segnale per giungere al ricevitore e l'esatta posizione di almeno 3 satelliti per avere una posizione 2D (bidimensionale), e 4 per avere una posizione 3D (tridimensionale), è possibile determinare la posizione nello spazio del ricevitore stesso.

In assenza di errori, queste misure permetterebbero di avere informazioni sulla posizione del ricevitore che è data da un punto sulla terra appartenente all'intersezione di una sfera che ha centro nel satellite e raggio uguale alla distanza misurata.

Tuttavia, gli errori (dell'ordine del secondo) negli orologi atomici del ricevitore GPS e del satellite, i disturbi della ionosfera e troposfera, etc., rendono tre satelliti insufficienti per determinare la posizione esatta del ricevitore. Per rendere la misura di localizzazione quanto più precisa possibile, **si usa un numero di satelliti superiore a tre.**



La stima della posizione è un problema di **approssimazione ai minimi quadrati.**

Indicata con $d_i = \sqrt{(x_i - x_0)^2 + (y_i - y_0)^2}$ la distanza misurata dal satellite i -esimo $P_i = (x_i, y_i)$ dal punto $P_0 = (x_0, y_0)$ da determinare e con ϵ_i l'errore corrispondente, si determina P_0 come il punto che rende minima la quantità $\sum_{i=1}^N \epsilon_i^2$ dove $N > 3$ è il numero di satelliti considerati.

Esempio 5 (Ing. delle Comunicazioni)

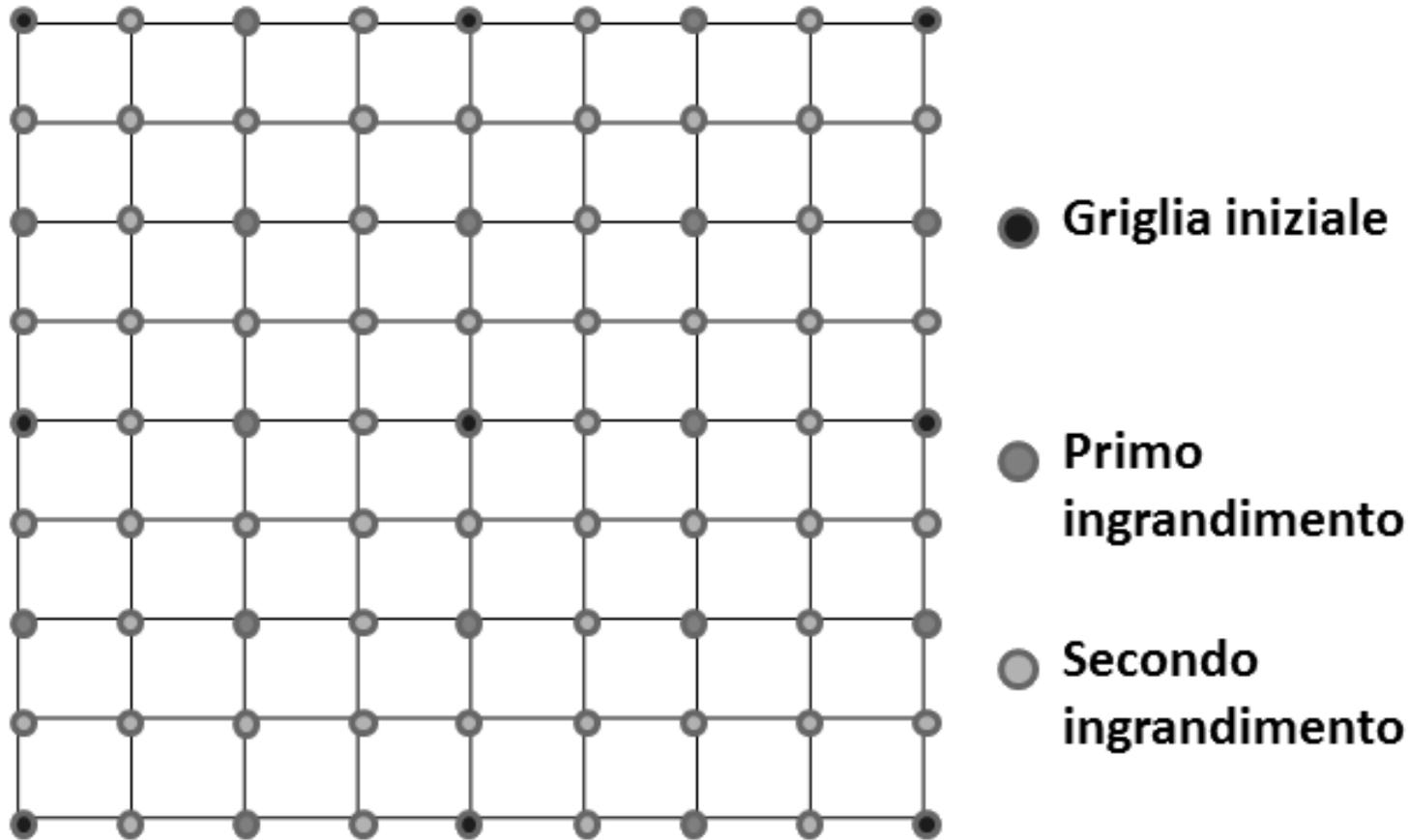
Zoom di immagini

Lo **image zooming** è un problema di approssimazione di funzione, nello specifico di interpolazione.

Data un' immagine di dimensione $n \times m$ pixels, la si vuole ingrandire. Cioè la si vuole portare ad una dimensione maggiore ($2n \times 2m$, $4n \times 4m$, etc.).



In altre parole la funzione immagine $I(x, y)$ è nota su una griglia di punti equispaziata di passo h sul piano (x, y) e la si vuole stimare sui punti di una griglia di passo più piccolo, per esempio $\frac{h}{2}$, $\frac{h}{4}$, etc..

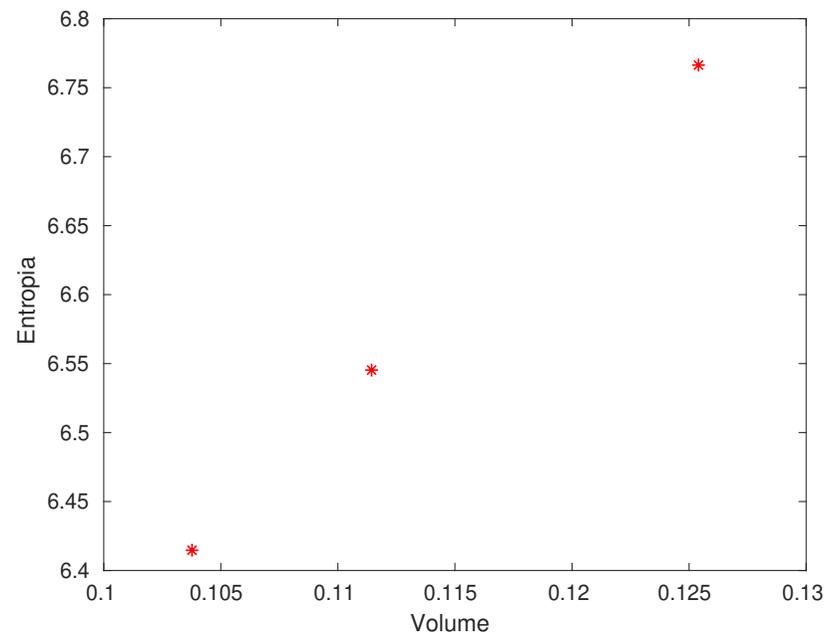


E' necessario stabilire con quale tipo di funzione si vuole 'riempire i buchi' della griglia (funzione lineare, cubica etc.)

Esempio 6 (Ing. Chimica)

Usare le stime di **vapore acqueo** riportate in **Tabella** per H_2O super-riscaldata a 200MPa per avere una stima dell'**entropia** s corrispondente al volume di $0.108\text{m}^3/\text{kg}$ supponendo (a) un'**approssimazione lineare** e (b) una **quadratica**.

$v(\text{m}^3/\text{kg})$	0.10377	0.11144	0.1254
$s(\text{kJ}/\text{kg}\cdot\text{K})$	6.4147	6.5453	6.7664



Approssimazione di dati e funzioni

Problema

Data la **tabella** $\{x_i, y_i\}$, $i = 0, \dots, n$, si vuole trovare una **funzione analitica** 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 $C^n([a, b])$

- **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 a 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 a 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 a rappresentare funzioni con 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 f_n in modo che

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

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

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

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

oppure, introducendo i **pesi** w_i ,

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

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

Esempio: retta di regressione (Esempio 2: costante elastica)

Interpolazione

Interpolazione polinomiale

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

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**

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

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** $\{x_i\}$, $i = 0, \dots, n$, è **regolare** poiché

$$\det V = \prod_{l>h} (x_l - x_h) \neq 0$$

\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**

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

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 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: $\{l_0(x), l_1(x), \dots, l_{n-1}(x), l_n(x)\}$

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

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

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

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

Espressione di Lagrange del polinomio interpolatore

Il **polinomio interpolatore** si definisce come combinazione lineare dei polinomi di base di Lagrange, cioè

$$L_n(x) = \sum_{k=0}^n \alpha_k l_k(x)$$

Ricordando che $l_k(x_i) = \begin{cases} 1 & \text{se } k = i \\ 0 & \text{se } k \neq i \end{cases}$ si ha

$$L_n(x_i) = \sum_{k=0}^n \alpha_k l_k(x_i) = \alpha_i l_i(x_i) = \alpha_i$$

Imponendo la condizione di interpolazione si ha

$$L_n(x) = \sum_{k=0}^n y_k l_k(x)$$


compaiono esplicitamente i **valori** nei nodi

$l_k(x)$ dipende solo dai **nod**

Il vettore $\mathbf{A} = [\alpha_0, \dots, \alpha_n]^T$ è soluzione del sistema lineare

$$V\mathbf{A} = Y$$

con

$$V = \begin{pmatrix} l_0(x_0) & l_1(x_0) & l_2(x_0) & \cdots & l_n(x_0) \\ l_0(x_1) & l_1(x_1) & l_2(x_1) & \cdots & l_n(x_1) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ l_0(x_n) & l_1(x_n) & l_2(x_n) & \cdots & l_n(x_n) \end{pmatrix} = I$$

Poichè V è la matrice identità $\Rightarrow \mathbf{A} = Y$

$L_n(x)$ è un **polinomio di grado n** che soddisfa le **condizioni di interpolazione**

\Rightarrow L'**unicità** del polinomio interpolatore implica $L_n(x) \equiv p_n(x)$

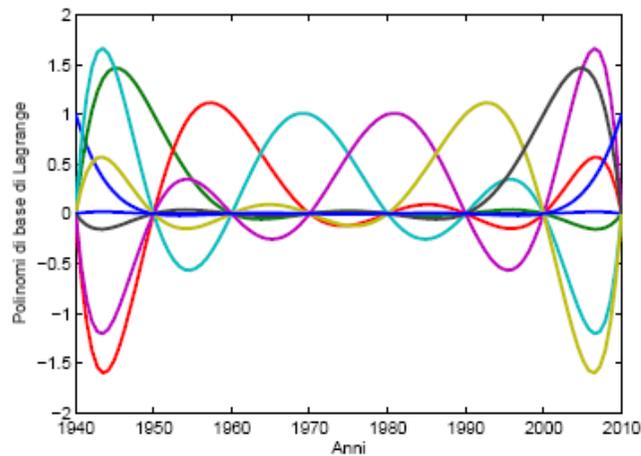
Oss: Non è necessario risolvere il sistema lineare

Esempio 1

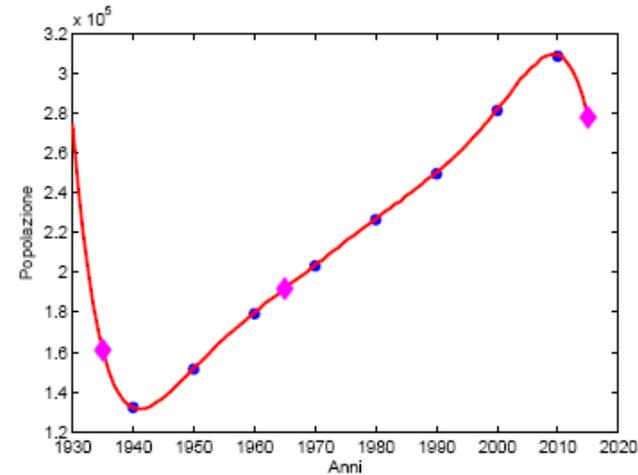
Data la tabella

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 1935, 1965, 2015. Quanto sono **accurati** i valori ottenuti?



Polinomi di base di Lagrange

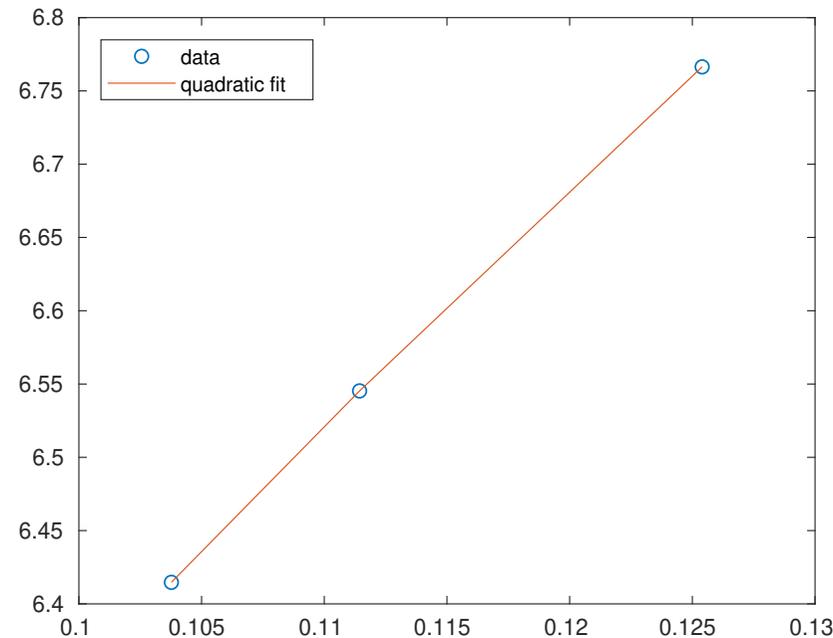


Polinomio interpolatore
(il simbolo \diamond indica i punti
di interpolazione)

Esempio 6 (Ing. Chimica)

Usare le stime di **vapore acqueo** riportate in **Tabella** per H_2O super-riscaldata a 200MPa per avere una stima dell'**entropia** s corrispondente al volume di $0.108\text{m}^3/\text{kg}$ supponendo (a) un'**approssimazione lineare** e (b) una **quadratica**.

$v(\text{m}^3/\text{kg})$	0.10377	0.11144	0.1254
$s(\text{kJ}/\text{kg}\cdot\text{K})$	6.4147	6.5453	6.7664



Da Matlab: polyfit e polyval

```
v = [0.10377 0.11144 0.1254];
```

```
s = [6.4147 6.5453 6.7664];
```

```
figure, plot(v,s,'r*')
```

```
xlabel('Volume'), ylabel('Entropia')
```

```
%Crea un polinomio di grado due dai dati:
```

```
p = polyfit(v,s,2);
```

```
% Valuta il polinomio in v e grafica
```

```
f = polyval(p,v);
```

```
plot(v,s,'o',v,f,'-')
```

```
legend('data','quadratic fit','Location','Northwest')
```

Interpolazione di polinomi

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

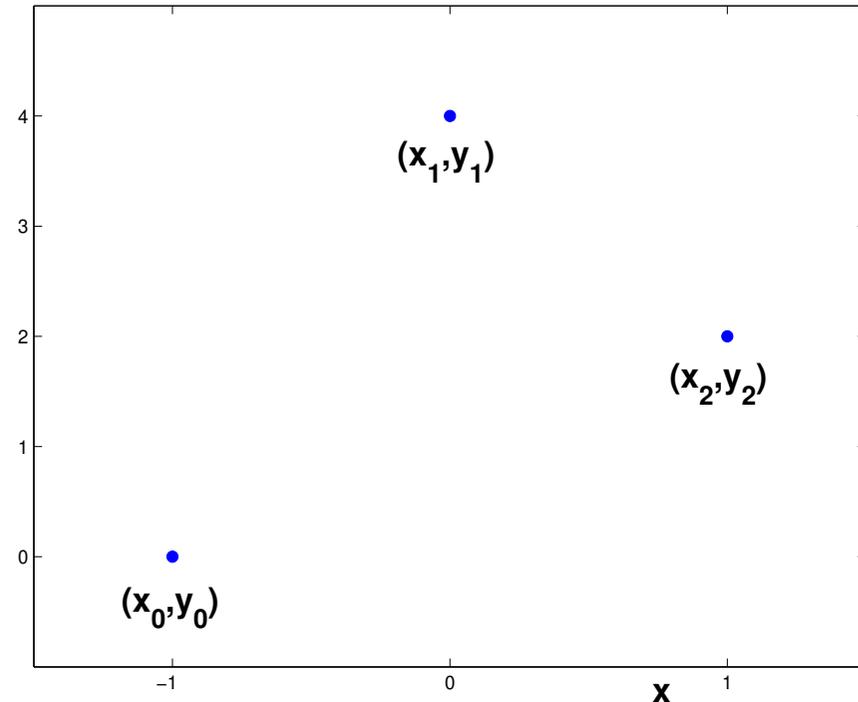
$$\{x_k, q_m(x_k)\} \quad k = 0, \dots, n \Rightarrow L_n(x) = \sum_{k=0}^n q_m(x_k) l_k(x) \equiv q_m(x)$$

Esempio.

Data la tavola

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

costruire il relativo **polinomio interpolatore**.



Soluzione

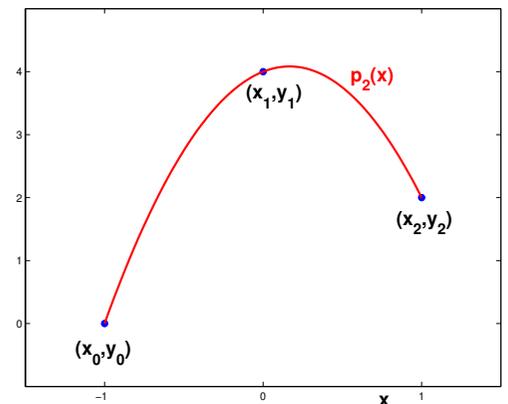
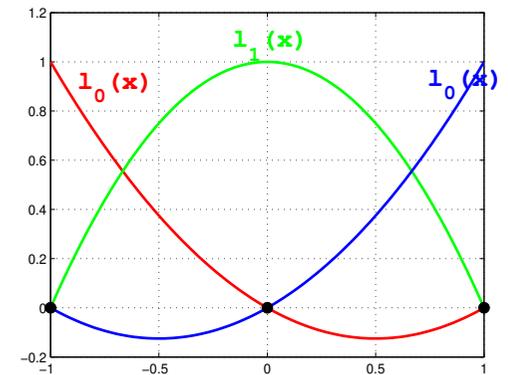
I dati sono $n + 1 = 3$ quindi il polinomio interpolatore è un polinomio di **grado 2**.

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

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

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

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



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**

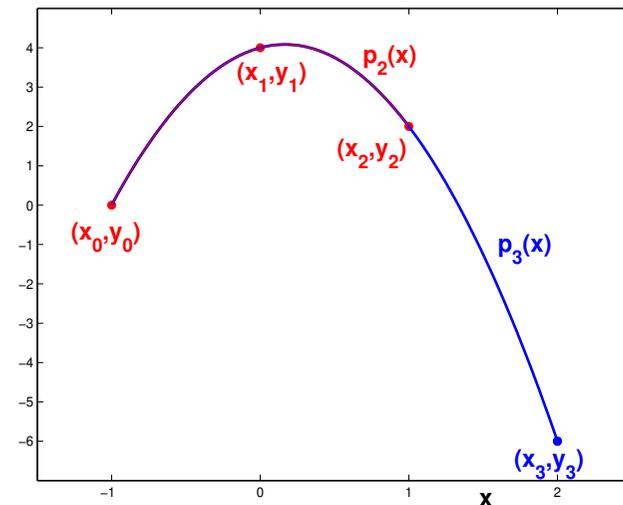
$$l_0(x) = \frac{(x - x_1)(x - x_2)(x - x_3)}{(x_0 - x_1)(x_0 - x_2)(x_0 - x_3)} = -\frac{1}{3}x(x - 1)(x - 2)$$

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

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

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

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



Interpolazione di funzioni costanti

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

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

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

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



$$\boxed{\sum_{k=0}^n l_k(x) = 1}$$

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)

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

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

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

Allora, definendo il **polinomio nodale**:

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

l' errore di troncamento è dato da

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

cioè $R_n(x) = \frac{f^{(n+1)}(\xi(x))}{(n+1)!}$

dim. 1

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

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

\Downarrow

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

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

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 $x_i = x_0 + ih \quad (i = 0, \dots, n)$

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

$$x = x_0 + sh \quad \pi_n(x) = s(s-1) \cdots (s-n) h^{n+1}$$

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

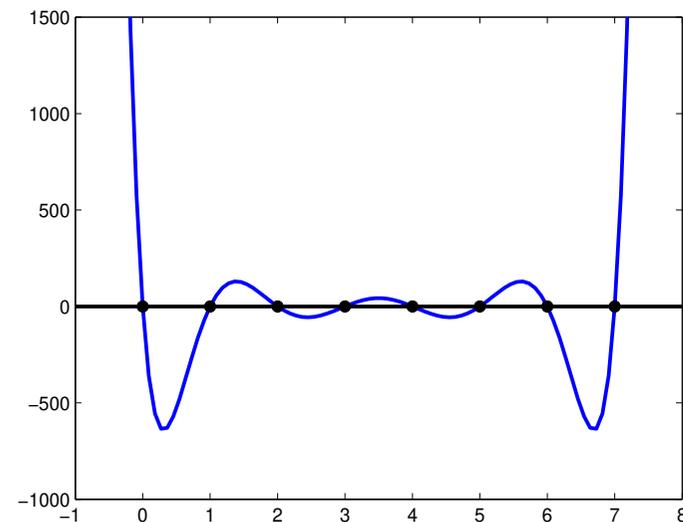


Grafico di $\pi_7(x)$

Errore di propagazione

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

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

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

Se $|\epsilon_i| \leq \epsilon$ si ottiene la limitazione

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

Funzione di Lebesgue:

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

rappresenta il **coefficiente di amplificazione** degli errori sui dati e si valuta nel punto in cui si vuole stimare l'errore di interpolazione

Costante di Lebesgue:

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

Proprietà della funzione di Lebesgue

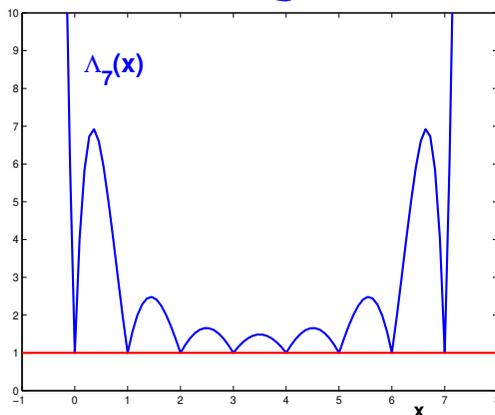
- $\Lambda(x)$ dipende solo dai polinomi fondamentali di Lagrange e, quindi, dalla **distribuzione dei nodi**

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

Nodi equispaziati in $[a, b]$

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

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

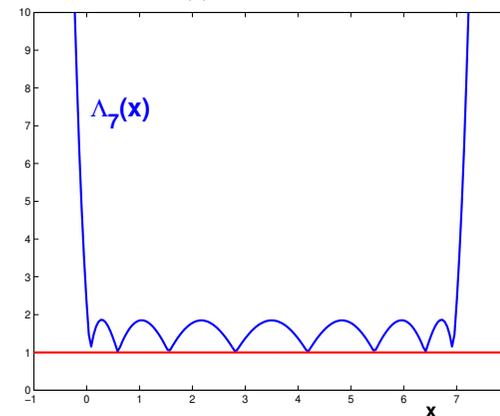


$n = 7$ Intervallo: $[0, 7]$

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

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.

Traccia della 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: $E_3(x) = \frac{\pi_3(x)}{4!} f^{(4)}(\tau) \quad \tau \in (0.0, 1.2)$

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

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

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**.

Inoltre $|E_3(t_0)| \leq 0.05$, $|E_3(t_2)| \leq 0.05 \rightarrow$ **un decimale esatto**, mentre $|E_3(t_1)| \leq 0.005 \rightarrow$ **due decimali esatti**.

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

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

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

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

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

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

Convergenza dei polinomi interpolatori

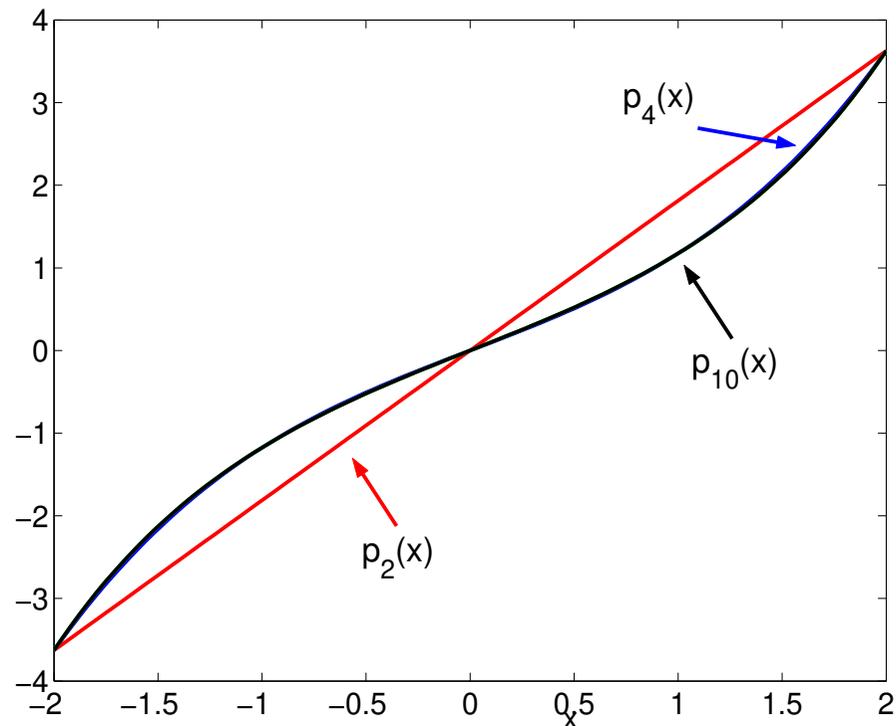
Si ha **convergenza** se, in **assenza** di **errori di arrotondamento**,

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

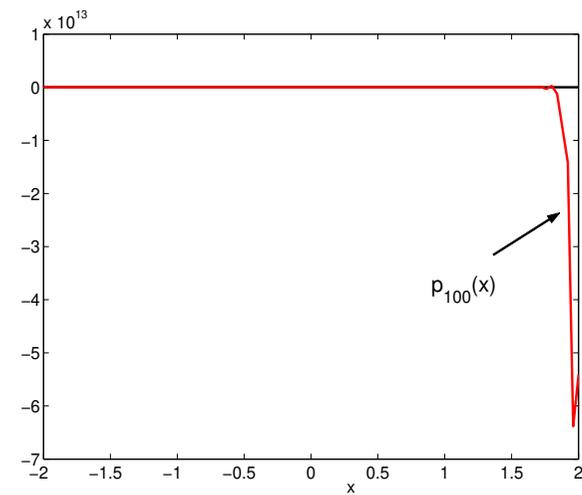
cioè se all'**aumentare del numero dei nodi** il polinomio interpolatore approssima sempre **meglio** la funzione.

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 $f(x)$.

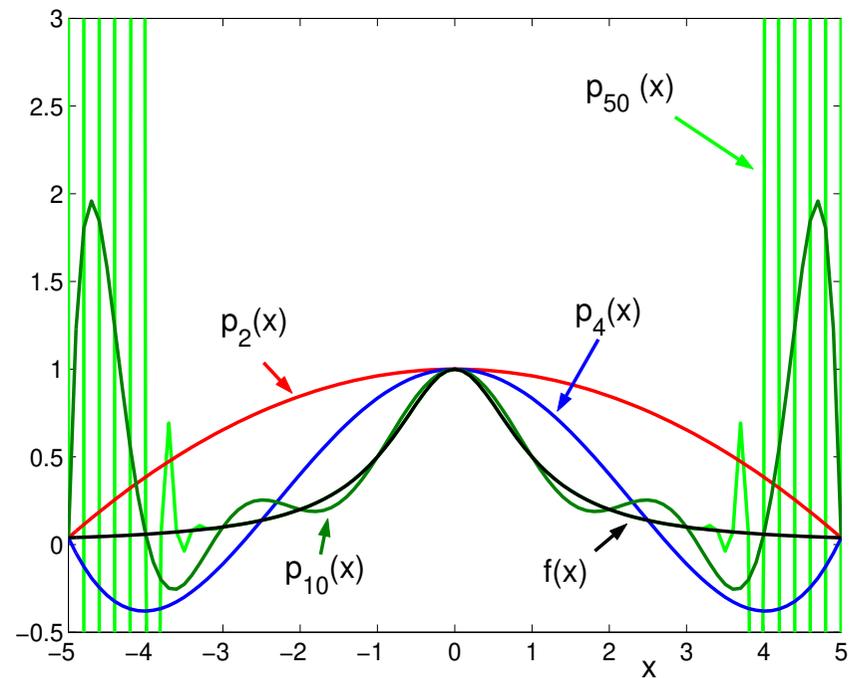


Nota. All'aumentare di n aumentano gli **errori di arrotondamento** che possono distruggere il risultato.



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

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 $[a, b]$.

dim: Si deve dimostrare che

$\forall \varepsilon > 0, \exists N$ indipendente da $x : \forall n > N, \forall x$ accade che $|f(x) - p_n(x)| < \varepsilon$.

D'altra parte $|f(x) - p_n(x)| = |E_n(x)| \leq \frac{(b-a)^{n+1}}{(n+1)!} M_{n+1} \rightarrow 0$

con $M_{n+1} = \max_{[a,b]} f^{n+1}(x)$.

Teorema 2. Sia $F(z)$ l'estensione di $f(x)$ nel piano complesso. Se $F(z)$ è **olomorfa** in una regione A del piano complesso contenente $[a, b]$ e, detta d la **distanza** di ∂A da $[a, b]$, se

$$d > b - a$$

allora $\lim_{n \rightarrow \infty} p_n(x) = f(x)$ uniformemente in $[a, b]$.

Nota. Per l'esempio di Runge, la funzione $F(z) = \frac{1}{1+z^2}$ ha due **singularità** in $\pm i$. Per l'intervallo $[-5, 5]$ si ha $d = 1 < b - a = 10$, quindi **non** si ha convergenza. Se si considera invece l'intervallo $[2, 3]$ si ha convergenza poiché in questo caso $d = \sqrt{5} > b - a = 1$.

Nota.

1. Le funzioni con derivata equilimitata in $[a, b]$ soddisfano le ipotesi del teorema.
Esempi: $\sin x$, $\cos x$, e^x .
2. La convergenza dei polinomi interpolatori è assicurata se i punti di singolarità della funzione da interpolare sono sufficientemente distanti dall'intervallo di interpolazione.
3. 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 $2 \cos \theta \cos n\theta = \cos(n+1)\theta + \cos(n-1)\theta$

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

$\Rightarrow \forall n, T_n(x)$ 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 $[-1, 1]$

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

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

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

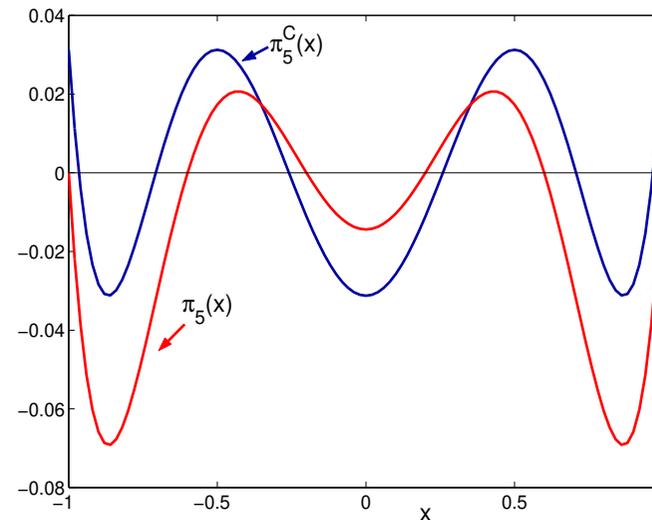
$$x_i^C = \frac{b-a}{2} \cos\left(\frac{2i+1}{n+1} \cdot \frac{\pi}{2}\right) + \frac{b+a}{2} \quad i = 0, 1, \dots, n$$

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**.

Interpolazione tramite funzioni splines

Se il numero n di **dati** nell'intervallo $[a, b]$ è molto **elevato** anche il **grado** n del polinomio interpolatore è molto **elevato**.



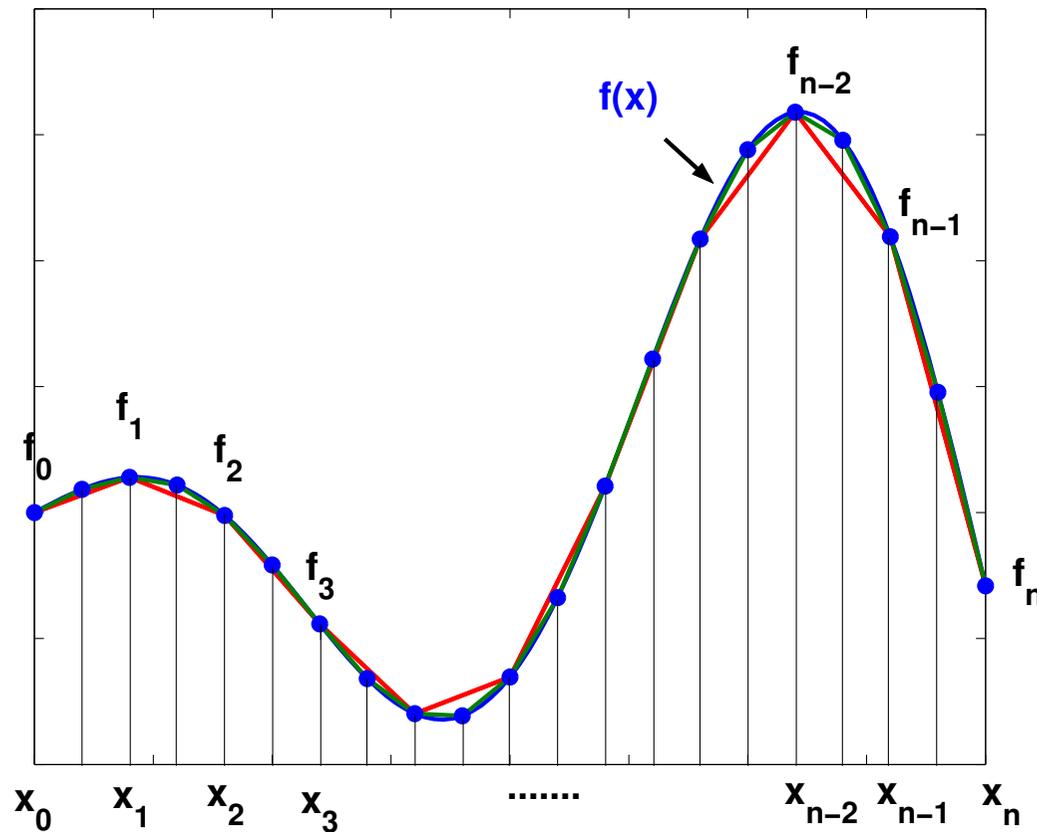
Il polinomio interpolatore può presentare delle **oscillazioni** estranee alla funzione da approssimare (**Esempio: fenomeno di Runge**).

Per evitare questo fenomeno si usa un **approccio differente**: invece di costruire un **unico** polinomio interpolatore su tutto l'intervallo $[a, b]$ si **divide** l'intervallo in un insieme di **sottointervalli** e si costruisce un **differente** polinomio interpolatore in ciascun **sottointervallo**, cioè si sceglie come funzione approssimante un **polinomio a tratti**.

Esempio

Polinomio lineare a tratti: si uniscono i dati $\{x_i, f_i\}$, $i = 0, \dots, n$, con una **spezzata**.

Svantaggio: non è **differenziabile** nei nodi $\{x_i\}$.



Funzioni splines

Definizione. Si definisce **partizione** Δ di un intervallo $[a, b]$ un insieme di punti x_i , $i = 0, 1, \dots, n$ tali che

$$\Delta : a = x_0 < x_1 < x_2 < \dots < x_{n-1} < x_n = b$$

Definizione. Si definisce **funzione spline** di grado m associata alla **partizione** Δ di $[a, b]$ una funzione $S_m(x)$ tale che

- $S_m(x)$ è un **polinomio** di grado m in ciascun **sottointervallo** $T_i = [x_{i-1}, x_i]$, $i = 1, 2, \dots, n$
- $S_m(x) \in C^{m-1}[a, b]$

Nota. $S_m(x)$ è un **polinomio** per $x \in (x_{i-1}, x_i) \implies S_m(x) \in C^\infty(x_{i-1}, x_i)$

$S_m^{(k)}(x_i^-) = S_m^{(k)}(x_i^+)$ per $k = 0, 1, \dots, m-1 \implies$ la spline è '**liscia**'

Splines lineari interpolanti

- Una **spline lineare** $S_1(x)$ è un **polinomio lineare a tratti** nell'intervallo $[a, b]$ che nei nodi x_i deve soddisfare le condizioni di

$$\text{Continuità : } S_1(x_i^+) = S_1(x_i^-) \quad i = 0, 1, \dots, n$$

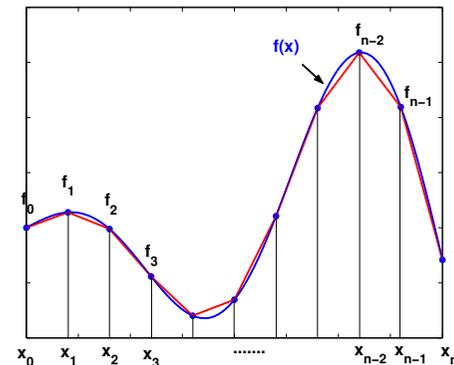
- Una **spline lineare interpolante** nei nodi x_i deve soddisfare anche le

$$\text{Condizioni di interpolazione: } S_1(x_i) = y_i \quad i = 0, 1, \dots, n$$

Per $x \in T_i = [x_{i-1}, x_i]$ $S_1(x) \in \mathbf{P}_1$. Dalle condizioni di interpolazione

$$\Rightarrow S_1(x) = \frac{1}{h_i} \left[(x_i - x)y_{i-1} + (x - x_{i-1})y_i \right] \quad h_i = x_i - x_{i-1} \quad x \in T_i$$

\Rightarrow La spline lineare interpolante **esiste** ed è **unica**



Una base per lo spazio delle splines lineari

Se introduciamo una **base** (**B-splines**) $B_i(x)$, $i = 0, \dots, n$, per lo spazio delle splines lineari tale che

$$B_i(x_j) = \begin{cases} 1 & \text{se } i = j \\ 0 & \text{se } i \neq j \end{cases} \quad \Rightarrow \quad S_1(x) = \sum_{i=0}^n y_i B_i(x) \quad x \in [a, b]$$

La funzione $B_i(x)$ è un polinomio lineare in T_i che soddisfa il **problema di interpolazione**: $B_i(x_j) = e_j$, $j = 0, \dots, n$, dove $e_i = (0, 0, \dots, \underbrace{1}_i, \dots, 0)$

$$B_i(x) = \begin{cases} \frac{x - x_{i-1}}{h_i} & \text{se } x \in [x_{i-1}, x_i] \\ \frac{x_{i+1} - x}{h_{i+1}} & \text{se } x \in [x_i, x_{i+1}] \\ 0 & \text{altrove} \end{cases} \quad i = 1, \dots, n-1$$

$$B_0(x) = \begin{cases} \frac{x_1 - x}{h_1} & \text{se } x \in [x_0, x_1] \\ 0 & \text{altrove} \end{cases}$$

$$B_n(x) = \begin{cases} \frac{x - x_{n-1}}{h_n} & \text{se } x \in [x_{n-1}, x_n] \\ 0 & \text{altrove} \end{cases}$$

Splines cubiche interpolanti

Splines cubiche ($m = 3$): $S_3^{(k)}(x_i^-) = S_3^{(k)}(x_i^+)$ $k = 0, 1, 2$

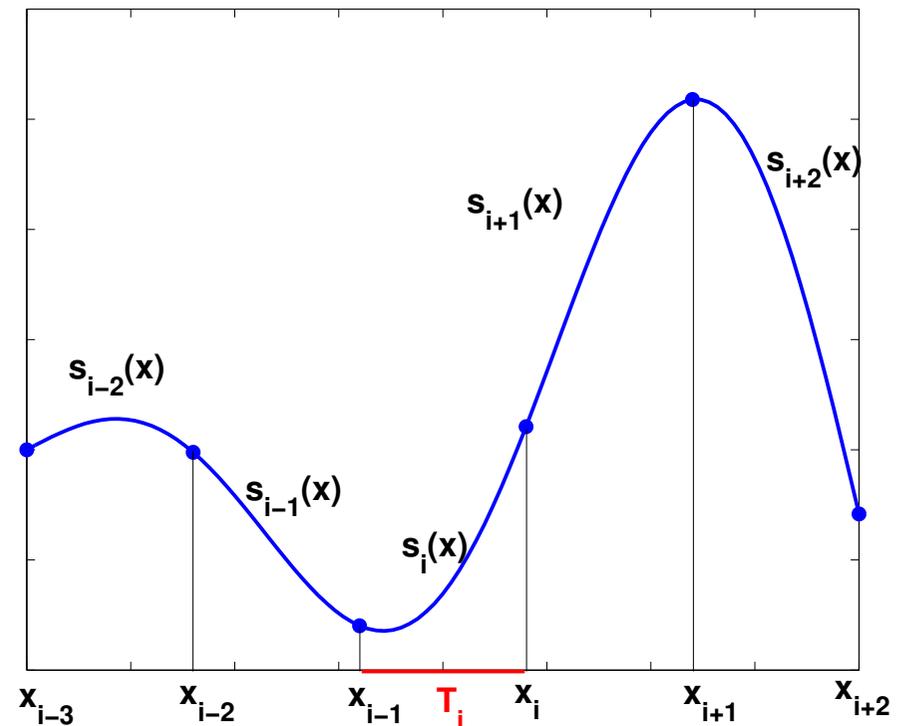
Condizioni di interpolazione: $S_3(x_i) = y_i$ $i = 0, 1, \dots, n$

Costruzione della spline cubica interpolante

$S_3(x)$ è un **polinomio** di grado 3
in ogni **sottointervallo**

$T_i = [x_{i-1}, x_i]$, $i = 1, 2, \dots, n$:

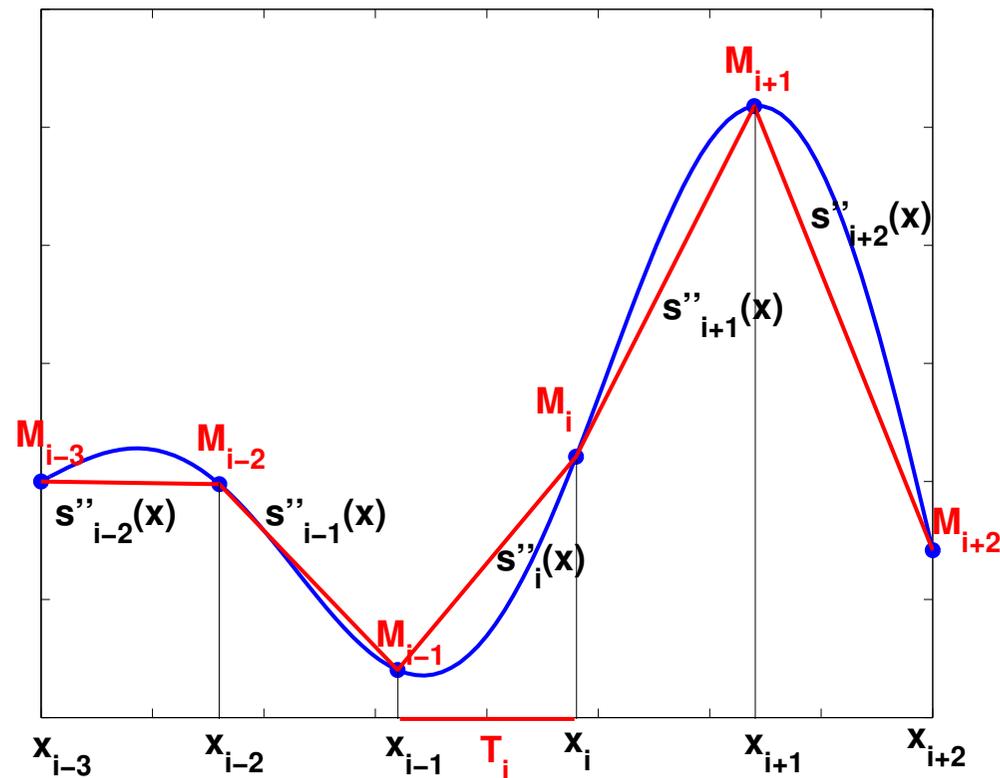
$$S_3(x)|_{x \in T_i} = s_i(x) \in \mathbf{P}_3$$



La **derivata seconda** di $S_3(x)$ è una funzione **lineare a tratti**:

$$S_3''(x)|_{x \in T_i} = s_i''(x) = \frac{1}{h_i} \left[(x_i - x)M_{i-1} + (x - x_{i-1})M_i \right] \quad h_i = x_i - x_{i-1}$$

$$\Rightarrow s_i''(x_{i-1}) = M_{i-1}, \quad s_i''(x_i) = M_i \Rightarrow S_3''(x_i) = M_i \quad i = 0, 1, \dots, n$$



Nota. Le costanti M_i sono **incognite** da determinare e **individuano** la spline.

Poichè $S_3''(x)$ è **continua** si può **integrare** due volte per ottenere $S_3(x)$:

$$\begin{aligned} S_3'(x)|_{x \in T_i} &= s_i'(x) = \int_{T_i} s_i''(x) dx = \\ &= \frac{1}{2h_i} \left[-(x_i - x)^2 M_{i-1} + (x - x_{i-1})^2 M_i \right] + C_i \end{aligned}$$

$$\begin{aligned} S_3(x)|_{x \in T_i} &= s_i(x) = \int_{T_i} s_i'(x) dx = \\ &= \frac{1}{6h_i} \left[(x_i - x)^3 M_{i-1} + (x - x_{i-1})^3 M_i \right] + C_i(x - x_{i-1}) + D_i \end{aligned}$$

Le **costanti di integrazione** C_i e D_i e le **incognite** M_i **individuano** la spline cubica interpolante e si **determinano** imponendo che $S_3(x)$ soddisfi:

i) le **condizioni di interpolazione**: $S_3(x_i) = y_i \quad i = 0, 1, \dots, n$

ii) la **continuità della derivata prima** nei nodi interni:

$$S_3'(x_i^-) = S_3'(x_i^+) \Rightarrow s_{i+1}'(x_i) = s_i'(x_i) \quad i = 1, \dots, n-1$$

Le **condizioni di interpolazione** danno

$$\left\{ \begin{array}{l} y_i = s_i(x_i) = \frac{1}{6h_i} \underbrace{(x_i - x_{i-1})^3}_{h_i^3} M_i + \underbrace{(x_i - x_{i-1})}_{h_i} C_i + D_i = \frac{1}{6} h_i^2 M_i + h_i C_i + D_i \\ y_{i-1} = s_i(x_{i-1}) = \frac{1}{6h_i} \underbrace{(x_i - x_{i-1})^3}_{h_i^3} M_{i-1} + D_i = \frac{1}{6} h_i^2 M_{i-1} + D_i \end{array} \right.$$



$$\left\{ \begin{array}{l} C_i = \frac{1}{h_i} (y_i - y_{i-1}) - \frac{h_i}{6} (M_i - M_{i-1}) \\ D_i = y_{i-1} - \frac{h_i^2}{6} M_{i-1} \end{array} \right. \quad i = 1, 2, \dots, n$$

$$\begin{aligned} S_3(x)|_{x \in T_i} = s_i(x) &= \frac{1}{6h_i} \left[(x_i - x)^3 M_{i-1} + (x - x_{i-1})^3 M_i \right] + \\ &+ \left[\frac{1}{h_i} (y_i - y_{i-1}) - \frac{h_i}{6} (M_i - M_{i-1}) \right] (x - x_{i-1}) + y_{i-1} - \frac{h_i^2}{6} M_{i-1} \end{aligned}$$

La **continuità della derivata prima** dà

$$s'_{i+1}(x_i) = s'_i(x_i) \quad i = 1, \dots, n-1$$

⇓

$$-\frac{1}{2h_{i+1}} \underbrace{(x_{i+1} - x_i)^2}_{h_{i+1}^2} M_i + C_{i+1} = \frac{1}{2h_i} \underbrace{(x_i - x_{i-1})^2}_{h_i^2} M_i + C_i$$

⇓

$$-\frac{1}{2}h_{i+1}M_i + \frac{1}{h_{i+1}}(y_{i+1} - y_i) - \frac{h_{i+1}}{6}(M_{i+1} - M_i) = \frac{1}{2}h_iM_i + \frac{1}{h_i}(y_i - y_{i-1}) - \frac{h_i}{6}(M_i - M_{i-1})$$

⇓

$$\frac{h_i}{6}M_{i-1} + \frac{h_i + h_{i+1}}{3}M_i + \frac{h_{i+1}}{6}M_{i+1} = \frac{y_{i+1} - y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}$$

Per $i = 1, 2, \dots, n - 1$ le relazioni

$$\frac{h_i}{6}M_{i-1} + \frac{h_i + h_{i+1}}{3}M_i + \frac{h_{i+1}}{6}M_{i+1} = \frac{y_{i+1} - y_i}{h_{i+1}} - \frac{y_i - y_{i-1}}{h_i}$$

formano un **sistema lineare** di $n - 1$ **equazioni** nelle $n + 1$ **incognite** $M_0, M_1, \dots, M_n \Rightarrow$ restano **due parametri** liberi.

Spline cubiche naturali

Si pone $M_0 = M_n = 0$, il che equivale a richiedere che la spline sia **lineare** al di fuori dell'intervallo $[a, b]$.

Spline cubiche naturali

$$M_0 = M_n = 0$$

$$\begin{array}{c}
 \Downarrow \\
 \left(\begin{array}{cccccc}
 \frac{h_1+h_2}{3} & \frac{h_2}{6} & 0 & 0 & \cdots & 0 \\
 \frac{h_2}{6} & \frac{h_2+h_3}{3} & \frac{h_3}{6} & 0 & \cdots & 0 \\
 0 & \frac{h_3}{6} & \frac{h_3+h_4}{3} & \frac{h_4}{6} & \cdots & 0 \\
 \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
 0 & 0 & \cdots & 0 & \frac{h_{n-1}}{6} & \frac{h_{n-1}+h_n}{3}
 \end{array} \right)
 \begin{array}{c}
 \left(\begin{array}{c}
 M_1 \\
 M_2 \\
 \vdots \\
 M_{n-2} \\
 M_{n-1}
 \end{array} \right)
 =
 \begin{array}{c}
 \left(\begin{array}{cc}
 \frac{y_2-y_1}{h_2} & -\frac{y_1-y_0}{h_1} \\
 \frac{y_3-y_2}{h_3} & -\frac{y_2-y_1}{h_2} \\
 \vdots & \vdots \\
 \frac{y_{n-1}-y_{n-2}}{h_{n-1}} & -\frac{y_{n-2}-y_{n-3}}{h_{n-2}} \\
 \frac{y_n-y_{n-1}}{h_n} & -\frac{y_{n-1}-y_{n-2}}{h_{n-1}}
 \end{array} \right)
 \end{array}
 \end{array}$$

$$\underbrace{\hspace{15em}}_C \quad \underbrace{\hspace{5em}}_M \quad = \quad \underbrace{\hspace{15em}}_F$$

La **matrice dei coefficienti** C è **simmetrica**, **tridiagonale**, **diagonalmente dominante** e ha elementi diagonali **positivi**.

\Rightarrow La matrice C è **definita positiva** e, quindi, **regolare**.

\Rightarrow La **spline cubica naturale interpolante** è **unica**.

Teoremi di convergenza

Teorema 1. Per ogni funzione $v \in C^2[a, b]$, passante per i punti (x_i, y_i) , si ha

$$\int_a^b |S_3''(x)|^2 dx \leq \int_a^b |v''(x)|^2 dx$$

Nota. Poichè la **curvatura** di una funzione è legata alla sua **derivata seconda**, la **spline cubica naturale** è quella che ha **minima curvatura globale** tra tutte le funzioni interpolanti $v \in C^2[a, b]$.

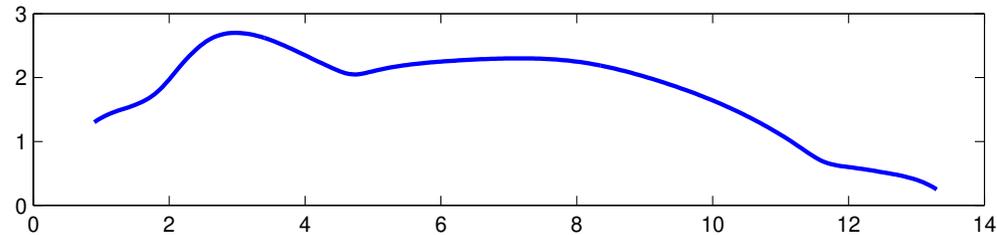
Teorema 2. Sia $f \in C^4[a, b]$, posto $h = \max_{1 \leq i \leq n} h_i$, risulta

$$\max_{x \in [a, b]} |f^{(k)}(x) - S_3^{(k)}| = O(h^{4-k}), \quad k = 0, 1, 2, 3$$

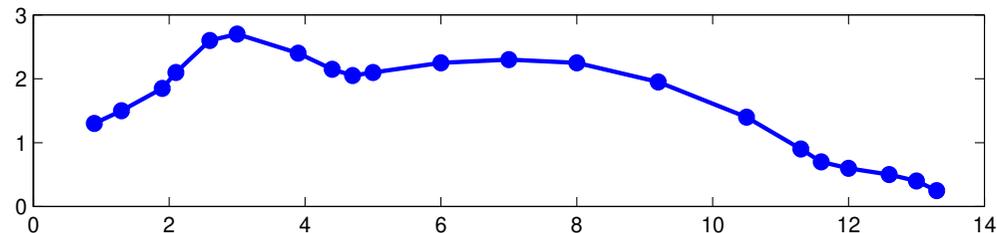
Nota. La spline cubica naturale interpolante converge alla **funzione f** e alle sue **derivate** sotto ipotesi **meno restrittive** rispetto all'interpolazione polinomiale.

Esempio

La figura qui sotto mostra il profilo di un'anatra in volo.



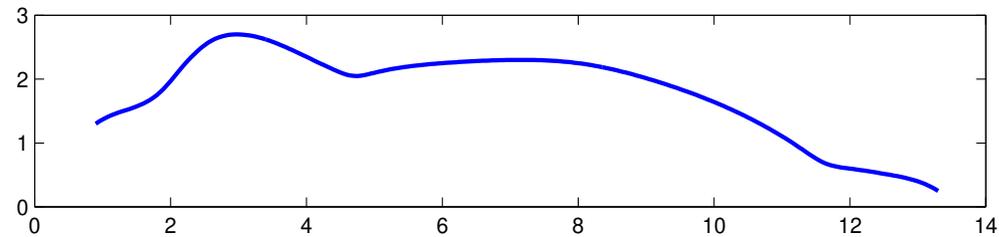
Per approssimare il profilo superiore scegliamo alcuni punti lungo la curva in modo che siano più fitti dove la curva varia più rapidamente.



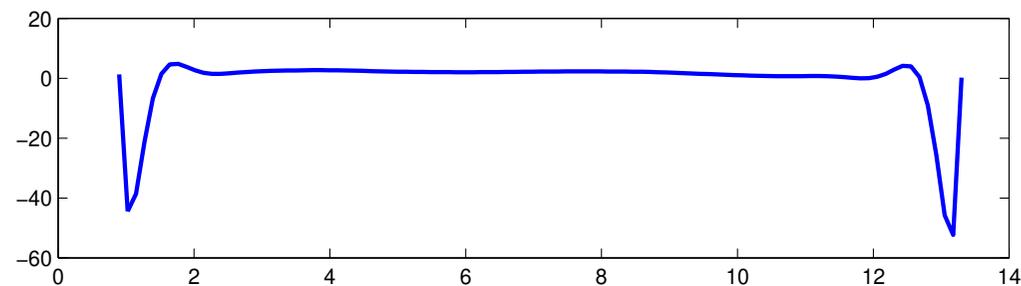
I punti scelti corrispondono alla seguente tabella.

x	0.9	1.3	1.9	2.1	2.6	3.0	3.9	4.4	4.7	5.0	6.0
$f(x)$	1.3	1.5	1.85	2.1	2.6	2.7	2.4	2.15	2.05	2.1	2.25
x	7.0	8.0	9.2	10.5	11.3	11.6	12.0	12.6	13.0	13.3	
$f(x)$	2.3	2.25	1.95	1.4	0.9	0.7	0.6	0.5	0.4	0.25	

La spline cubica naturale che interpola i dati in tabella è mostrata nella figura: la spline è praticamente indistinguibile dal profilo dell'anatra.



Nella figura qui sotto è mostrato il polinomio interpolatore che approssima gli stessi dati: si tratta di un polinomio di grado 20 che presenta delle forti oscillazioni soprattutto in prossimità dei bordi dell'intervallo di integrazione



Esercizio

Mostrare che la seguente funzione è una spline cubica naturale passante per i punti $(0,1)$ $(1,1)$ $(2,0)$ $(3,10)$

$$s(x) = \begin{cases} s_1(x) & 0 \leq x < 1 \\ s_2(x) & 1 \leq x < 2 \\ s_3(x) & 2 \leq x \leq 3 \end{cases} = \begin{cases} 1 + x - x^3 & 0 \leq x < 1 \\ 1 - 2(x-1) - 3(x-1)^2 + 4(x-1)^3 & 1 \leq x < 2 \\ 4(x-2) + 9(x-2)^2 - 3(x-2)^3 & 2 \leq x \leq 3 \end{cases}$$

Suggerimento: Per verificare che $s(x)$ è una spline cubica naturale necessario verificare:

- la continuità di $s(x)$ anche agli estremi dei sottointervalli: $s_1(1) = s_2(1); s_2(2) = s_3(2)$
- le condizioni di interpolazione : $s_1(0) = 1; s_2(1) = 1; s_3(2) = 0; s_3(10) = 1$
- la continuità di $s'(x)$ anche agli estremi dei sottointervalli: $s'_1(1) = s'_2(1); s'_2(2) = s'_3(2)$
- la continuità di $s''(x)$ anche agli estremi dei sottointervalli: $s''_1(1) = s''_2(1); s''_2(2) = s''_3(2)$
- la condizione di spline cubica naturale: $s''(0) = s''(3) = 0$

Esercizio

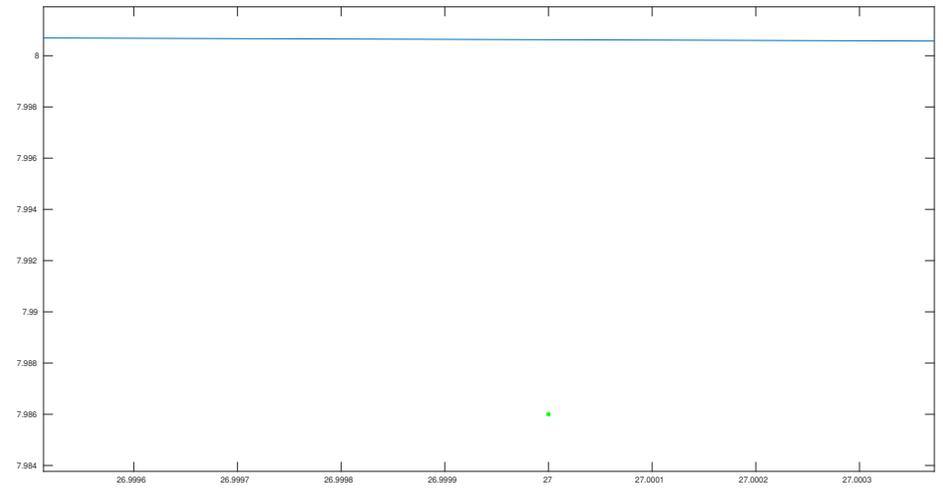
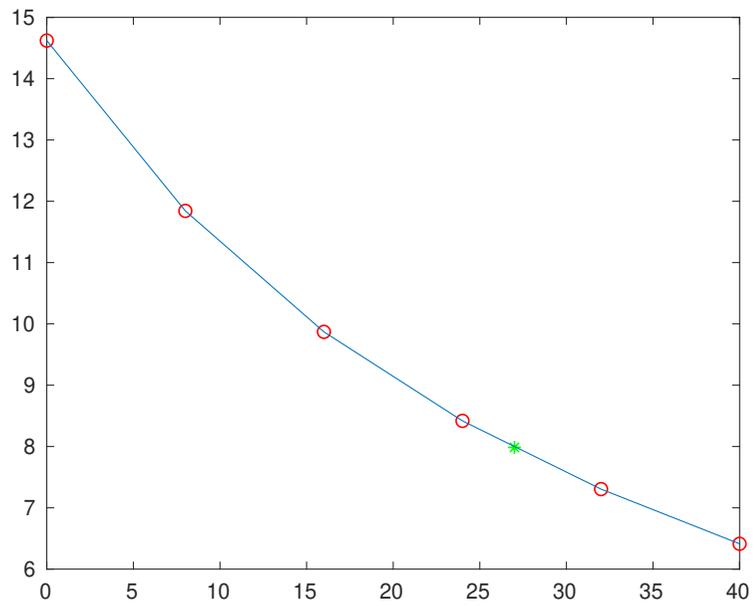
La Tabella contiene i livelli di ossigeno disciolto in acqua di mare in funzione della temperatura:

$T, ^\circ\text{C}$	0	8	16	24	32	40
$o, \text{mg/L}$	14.621	11.843	9.870	8.418	7.305	6.413

Stimare $o(27)$ (valore esatto: 7.986 mg/L)

con interpolazione spline cubica.

Soluzione



Da Matlab:

```
T = [0816243240];
```

```
o = [14.62111.8439.8708.4187.3056.413];
```

```
Tv = 27;
```

```
ov = 7.986;
```

```
figure, plot(T,o,'ro')
```

```
hold on
```

```
plot(Tv,ov,'g*')
```

```
oo = spline(T,o,T);
```

```
plot(T,oo);
```

Metodi di approssimazione

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

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

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

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

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

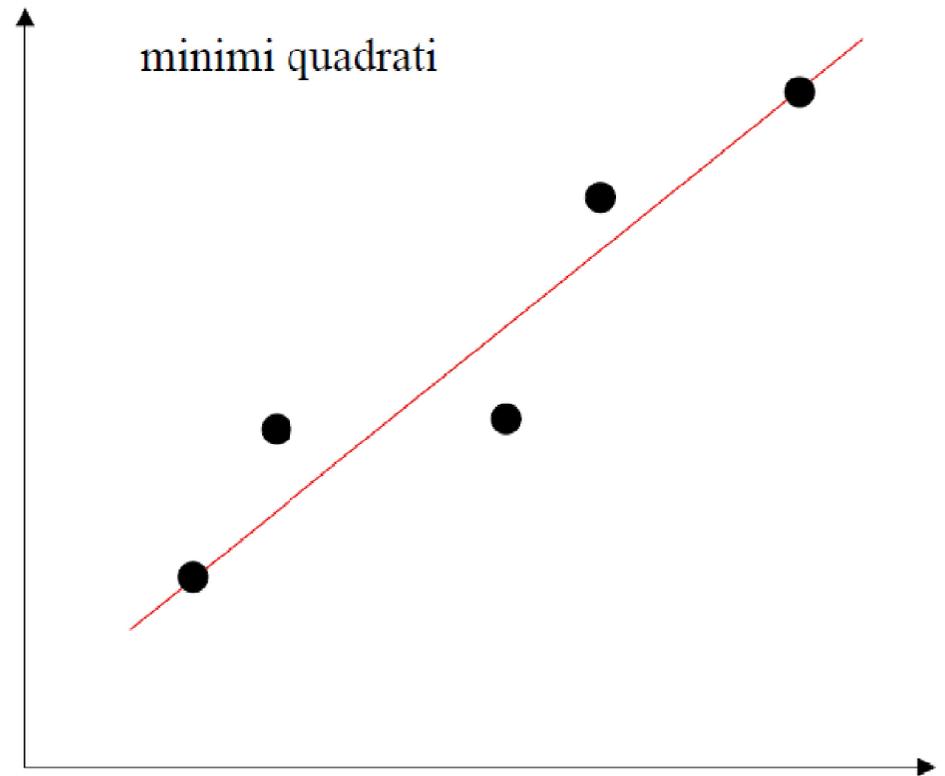
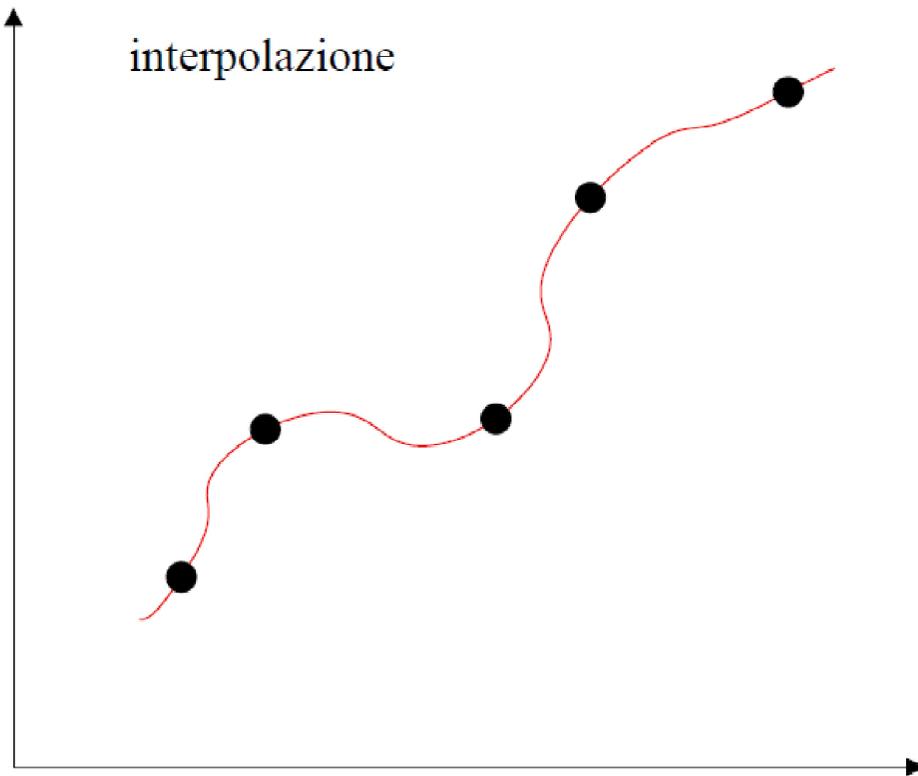
oppure, introducendo i **pesi** w_i ,

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

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

Esempio: retta di regressione (Esempio 2: costante elastica)

In sintesi



Approssimazione ai minimi quadrati

Problema

Problema.

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

Problema: si deve scegliere la funzione approssimante φ_M in modo da **minimizzare**

$$\sum_{i=0}^n [\varphi_M(x_i) - y_i]$$

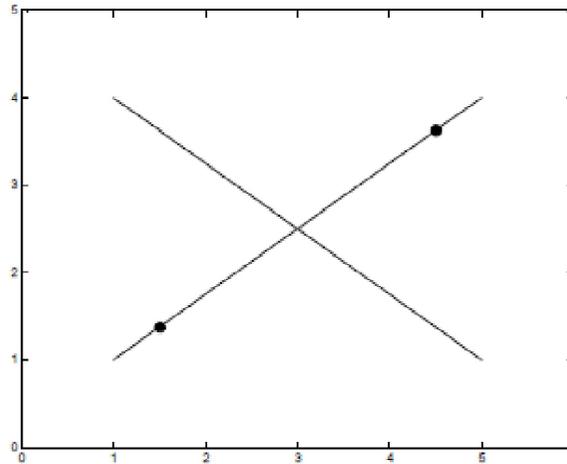
oppure, introducendo i **pesi** $w_i > 0$, $\forall i$,

$$\sum_{i=0}^n w_i [\varphi_M(x_i) - y_i]$$

Alcune considerazioni preliminari

Prima strategia: semplice **somma degli errori**

$$\sum_{i=0}^n [\varphi_M(x_i) - y_i]$$



Soluzione: retta che congiunge i due punti

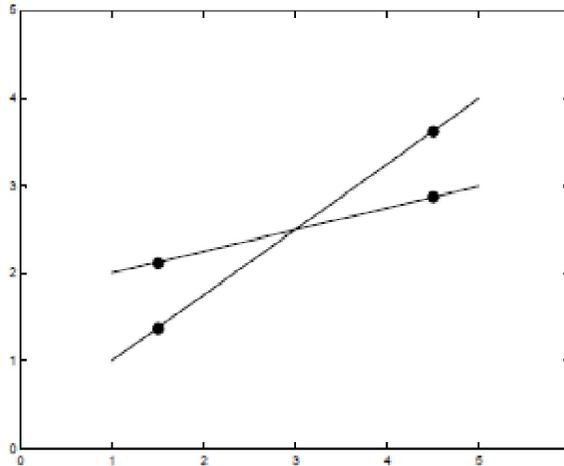
Problema: qualsiasi retta che passa per il punto medio rende minima la quantità sopra !!!

Nota: trascuriamo i pesi (per ora)!

Alcune considerazioni preliminari

Seconda strategia: somma dei valori assoluti degli errori

$$\sum_{i=0}^n |\varphi_M(x_i) - y_i|$$



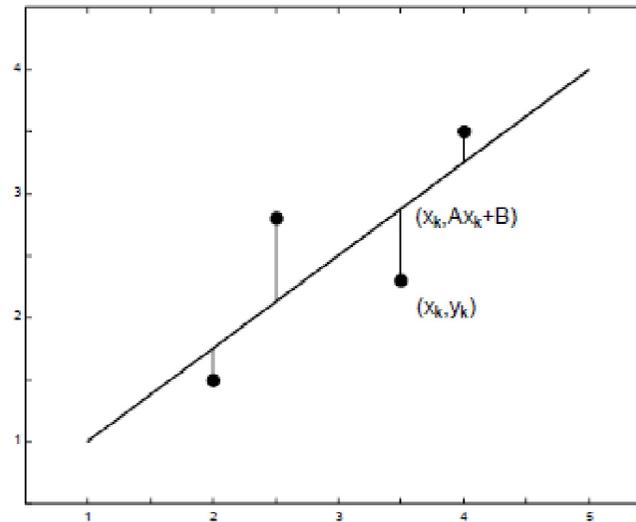
Soluzione: qualsiasi retta passante per il punto di intersezione delle rette che uniscono i punti a due a due e compresa fra esse.

Problema: c'è piú di una soluzione !!!

Alcune considerazioni preliminari

Terza strategia (corretta): somma dei quadrati degli errori

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



Soluzione: un'**unica soluzione** che minimizza la distanza verticale tra i dati e la retta.

Nota: l'errore sulle ascisse é nullo

Approfondimento del caso discreto

Problema.

Data la **tabella** $\{x_i, y_i\}$, $i = 0, \dots, n$, si vuole trovare una **funzione analitica** 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 ai minimi quadrati

Funzione approssimante:

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

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

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

Risolvere il problema dell'**approssimazione polinomiale ai minimi quadrati** vuol dire individuare il polinomio P_M , cioè i **coefficienti reali** a_k , per il quale σ è **minimo**.

Minimizzazione

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

$$\Rightarrow \frac{\partial \sigma^2}{\partial a_k} = 0 \quad k = 0, 1, \dots, M$$

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

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

- Per trovare i 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** $\left[\frac{\partial^2 \sigma^2}{\partial a_j \partial a_k} \right] = 2 \left[\sum_{i=0}^n x_i^{k+j} \right] = 2H.$

Unicità della soluzione

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

dove $V = \begin{bmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^M \\ 1 & x_1 & x_1^2 & \cdots & x_1^M \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 1 & x_n & x_n^2 & \cdots & x_n^M \end{bmatrix} \in \mathbf{R}^{n+1 \times M+1}$ è la **matrice di Vandermonde** dei nodi $\{x_i\}$.

\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**.

\Rightarrow La matrice hessiana $2H$ è **definita positiva** \Rightarrow la soluzione corrisponde a un **minimo**.

Nota 1 La matrice H è **malcondizionata**

Nota 2 Se $M = n$, ci si riconduce ad un problema di interpolazione

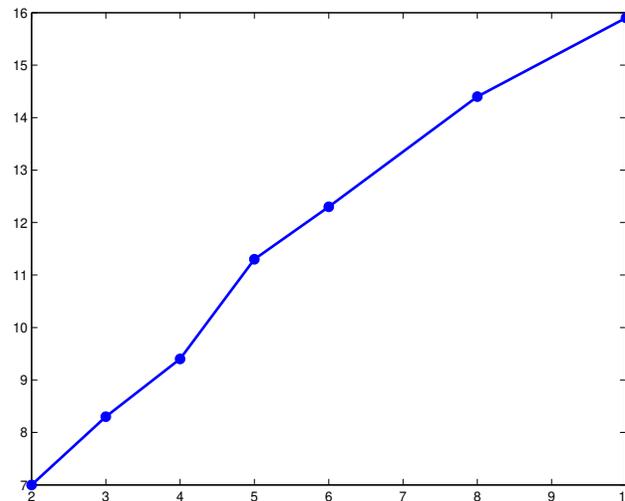
Esempio

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 qui di seguito sono riportate le misure sperimentali relative a una particolare molla.

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.



Retta di regressione

Problema. Costruire il **polinomio di grado 1** $P_1(x) = a_0 + a_1x$ che approssima i dati $\{x_i, y_i\}$, $i = 0, \dots, n \gg 1$ nel senso dei **minimi quadrati**.

Equazioni normali

\Rightarrow

Soluzione

$$\begin{cases} a_0s_0 + a_1s_1 = v_0 \\ a_0s_1 + a_1s_2 = v_1 \end{cases}$$

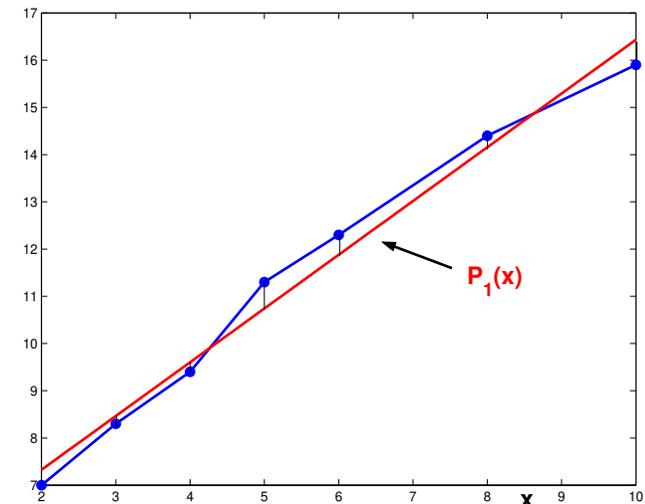
\Rightarrow

$$a_0 = \frac{v_0s_2 - v_1s_1}{s_0s_2 - s_1^2}$$
$$a_1 = \frac{s_0v_1 - s_1v_0}{s_0s_2 - s_1^2}$$

Esempio

Approssimando ai minimi quadrati i dati relativi alla molla si ottiene $a_0 = 5.049$ e $a_1 = 1.1383$. Il coefficiente a_1 fornisce l'approssimazione della costante elastica. Per questi valori dei coefficienti si ha

$$\sigma^2(a_0, a_1) = \sum_{i=0}^6 [a_0 + a_1x_i - y_i]^2 = 1.0071$$



Interpretazione probabilistica - 1

Siano x una **variabile deterministica** e $y = a_0 + a_1x$ la **variabile dipendente**, legata a x da una **relazione lineare**.

Nel caso in cui 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

La tabella seguente riporta le misure della densità relativa ρ dell'aria a diverse altezze h .

h (km)	0	1.525	3.050	4.575	6.100	7.625	9.150
ρ	1	0.8617	0.7385	0.6292	0.5328	0.4481	0.3741

Si approssimi ρ con un polinomio di secondo grado e si stimi il valore di ρ in corrispondenza di $h = 10.5$ km.

Si cerca il polinomio $p_2(h) = a_0 + a_1h + a_2h^2$ i cui coefficienti sono le soluzioni del seguente 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

$$s_0 = \sum_{i=0}^6 h_i^0 = 7$$

$$s_1 = \sum_{i=0}^6 h_i^1 = 1.525 + 3.050 + 4.575 + 6.100 + 7.625 + 9.150 = 32.0250$$

$$s_2 = \sum_{i=0}^6 h_i^2 = 1.525^2 + 3.050^2 + 4.575^2 + 6.100^2 + 7.625^2 + 9.150^2 = 211.6319$$

$$s_3 = \sum_{i=0}^6 h_i^3 = 1564.0410$$

$$s_4 = \sum_{i=0}^6 h_i^4 = 12304.4095$$

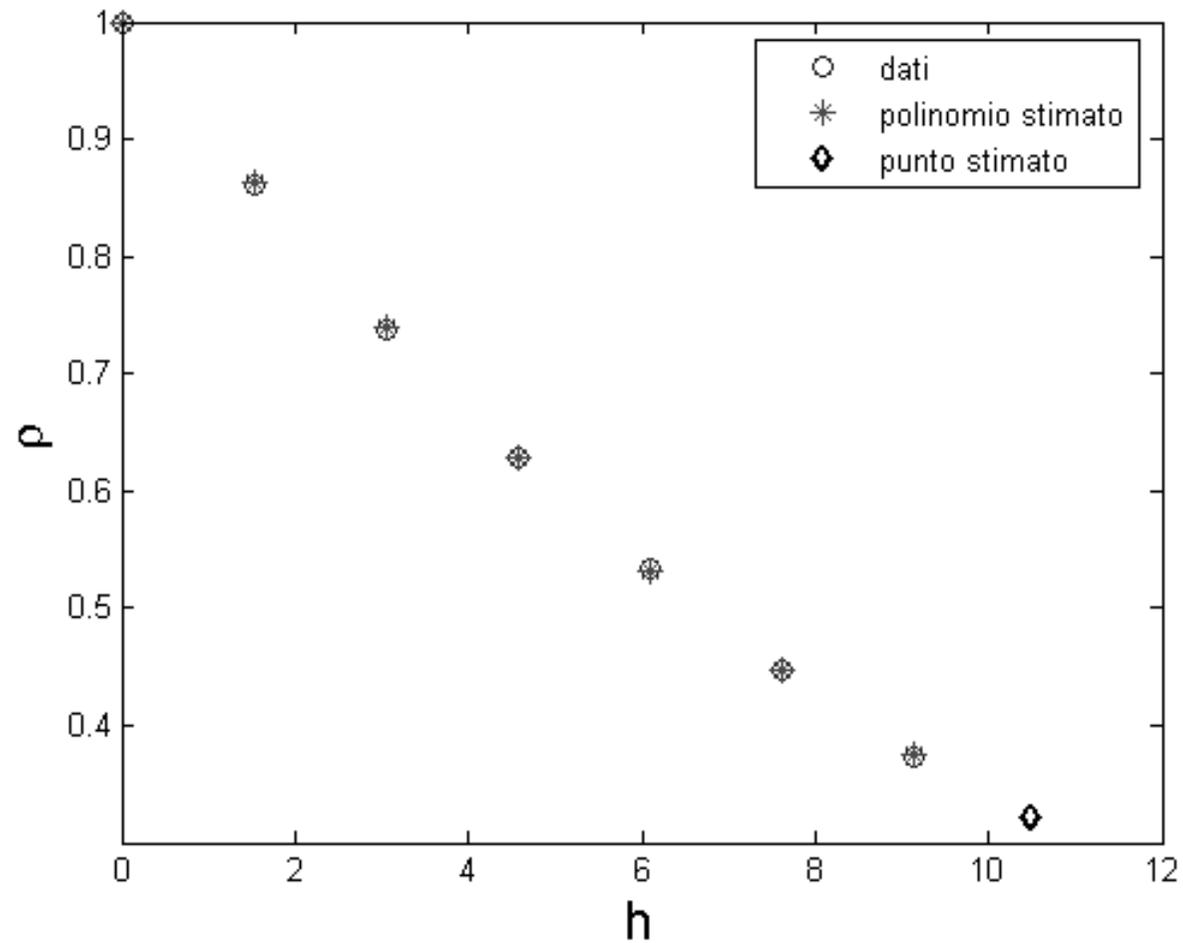
$$v_0 = \sum_{i=0}^6 h_i^0 \rho_i = 4.5844$$

$$v_1 = \sum_{i=0}^6 h_i \rho_i = 16.5350$$

$$v_2 = \sum_{i=0}^6 h_i^2 \rho_i = 99.2423$$

Risolvere il sistema e verificare che $a_2 = 0.0028$ $a_1 = -0.0934$ $a_0 = 0.9989$ e

$$\sigma^2(a_0, a_1, a_2) = \sum_{i=0}^6 (a_0 + a_1 h_i + a_2 h_i^2 - \rho_i)^2 = 2.9921 \cdot 10^{-5}$$



Quindi

$$p_2(h) = 0.9989 - 0.0934h + 0.0028h^2$$

da cui $p_2(10.5) = 0.3269$

Esercizio

Trovare il polinomio di secondo grado che meglio approssima i seguenti dati ai minimi quadrati:

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

E' necessario risolvere il seguente sistema

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

da cui si deduce facilmente che $[a_0 \ a_1 \ a_2]^T = [0 \ 0 \ 1]^T$

e quindi $p_2(x) = x^2$

Se i valori y_i sono affetti da errore, cioè

$$y_i = [0.0674 \ -0.9156 \ 1.6253 \ 3.0377 \ 3.3535 \ 7.9409],$$

il sistema diventa

$$\begin{pmatrix} 6 & 7.5 & 13.75 \\ 7.5 & 13.75 & 28.125 \\ 13.75 & 28.125 & 61.1875 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 15.1093 \\ 32.2834 \\ 71.276 \end{pmatrix}$$

da cui $[a_0 \ a_1 \ a_2]^T = [-0.1812 \ -0.3221 \ 1.3537]^T$ e quindi

$$p_2(x) = -0.1812 - 0.3221x + 1.3537x^2$$

Uso dei pesi

Si può vedere che:

$$a_0 = \bar{y} - \bar{x} \frac{\sum w_i (x_i - \bar{x})(y_i - \bar{y})}{\sum w_i (x_i - \bar{x})^2}$$

$$a_1 = \frac{\sum w_i (x_i - \bar{x})(y_i - \bar{y})}{\sum w_i (x_i - \bar{x})^2}$$

$$\bar{x} = \frac{\sum w_i x_i}{\sum w_i}$$

$$\bar{y} = \frac{\sum w_i y_i}{\sum w_i}$$

Matlab

```
function [a0,a1,xs,ys] = bestfit(x,y,w)

den1 = sum(w);

xs = (w * x') / den1;

ys = (w * y') / den1;

den2 = w * ((x - xs).^2)';

a1 = (w * ((x - xs) .* (y - ys))') / den2;

a0 = ys - xs * a1;
```

Matlab

```
x = [1 2 3 4 5 6 7 8 9 10];
```

```
y = [9 22 27 47 50 58 62 70 81 89];
```

```
w = ones(1,10);
```

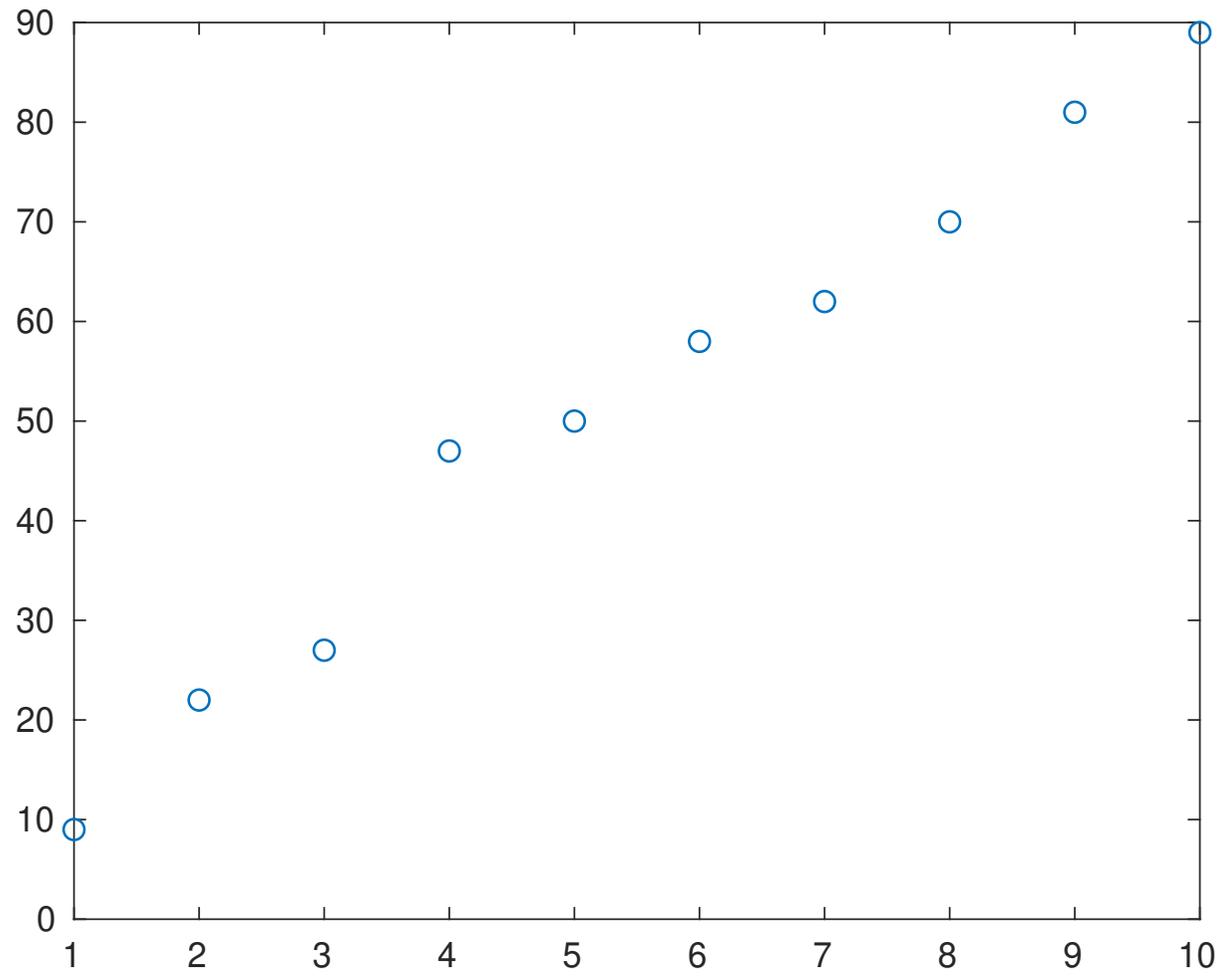
```
[a0,a1,xs,ys] = bestfit(x,y,w);
```

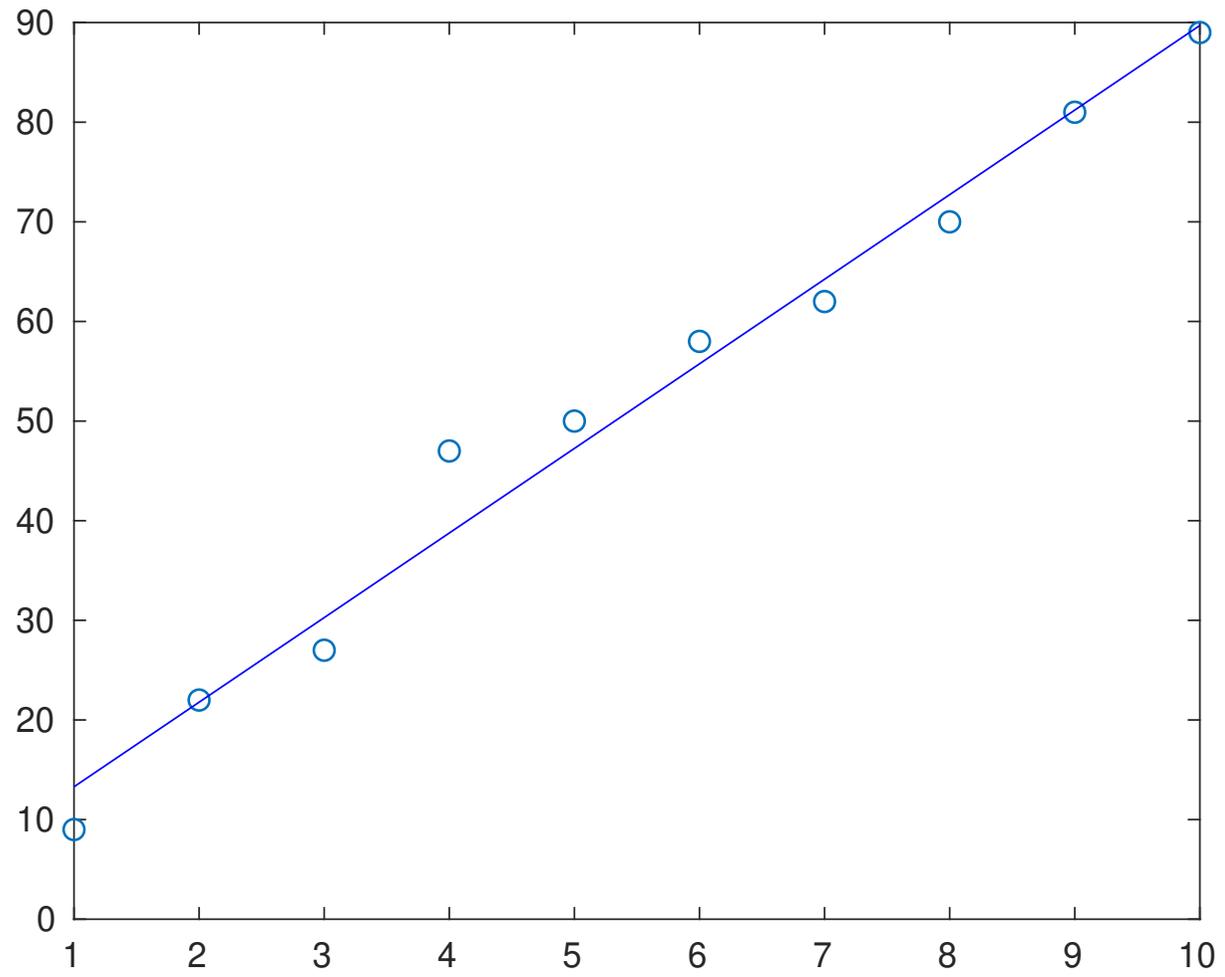
```
yc = ys + a1 * (x - xs);
```

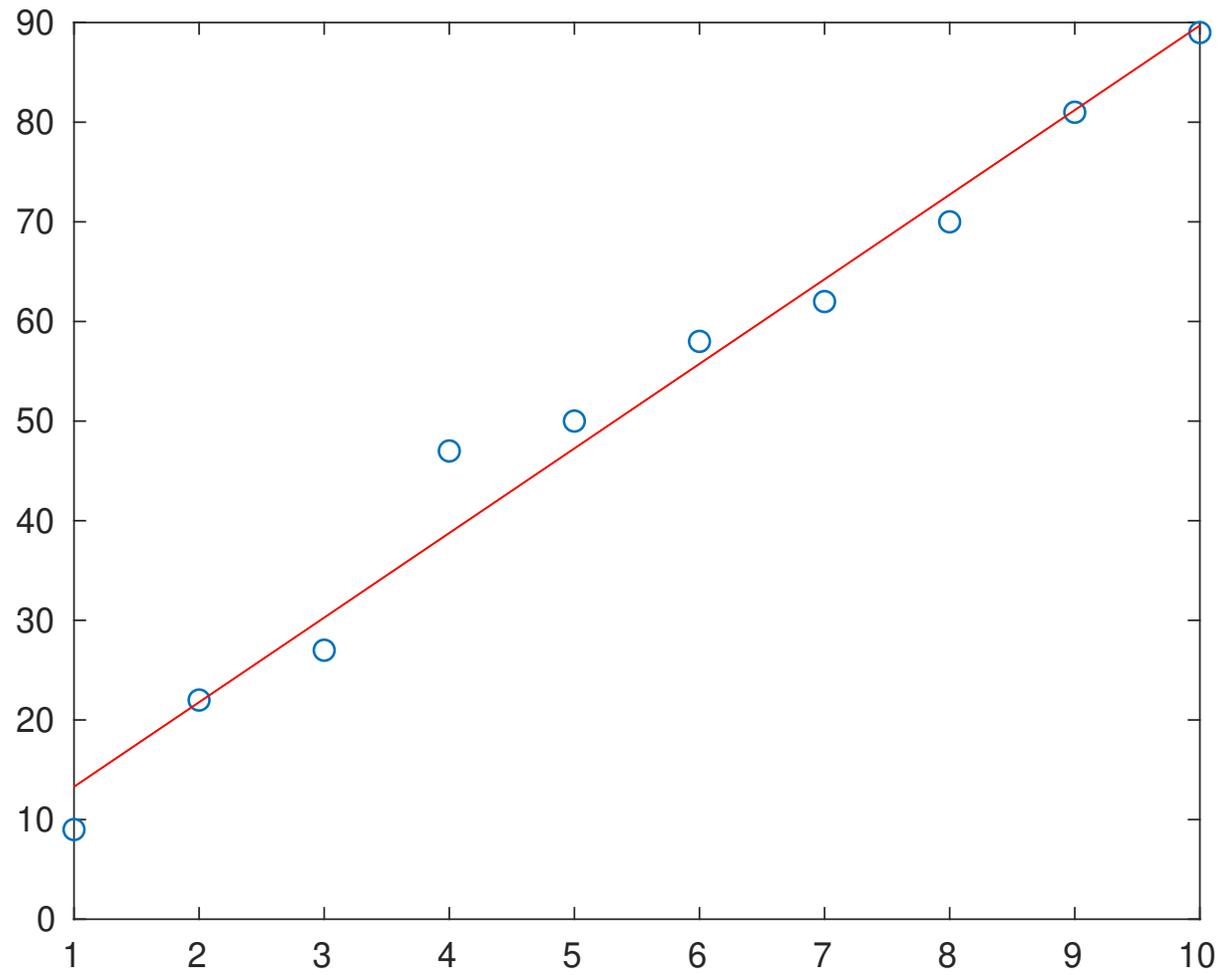
```
yc1 = a0 + a1 * x;
```

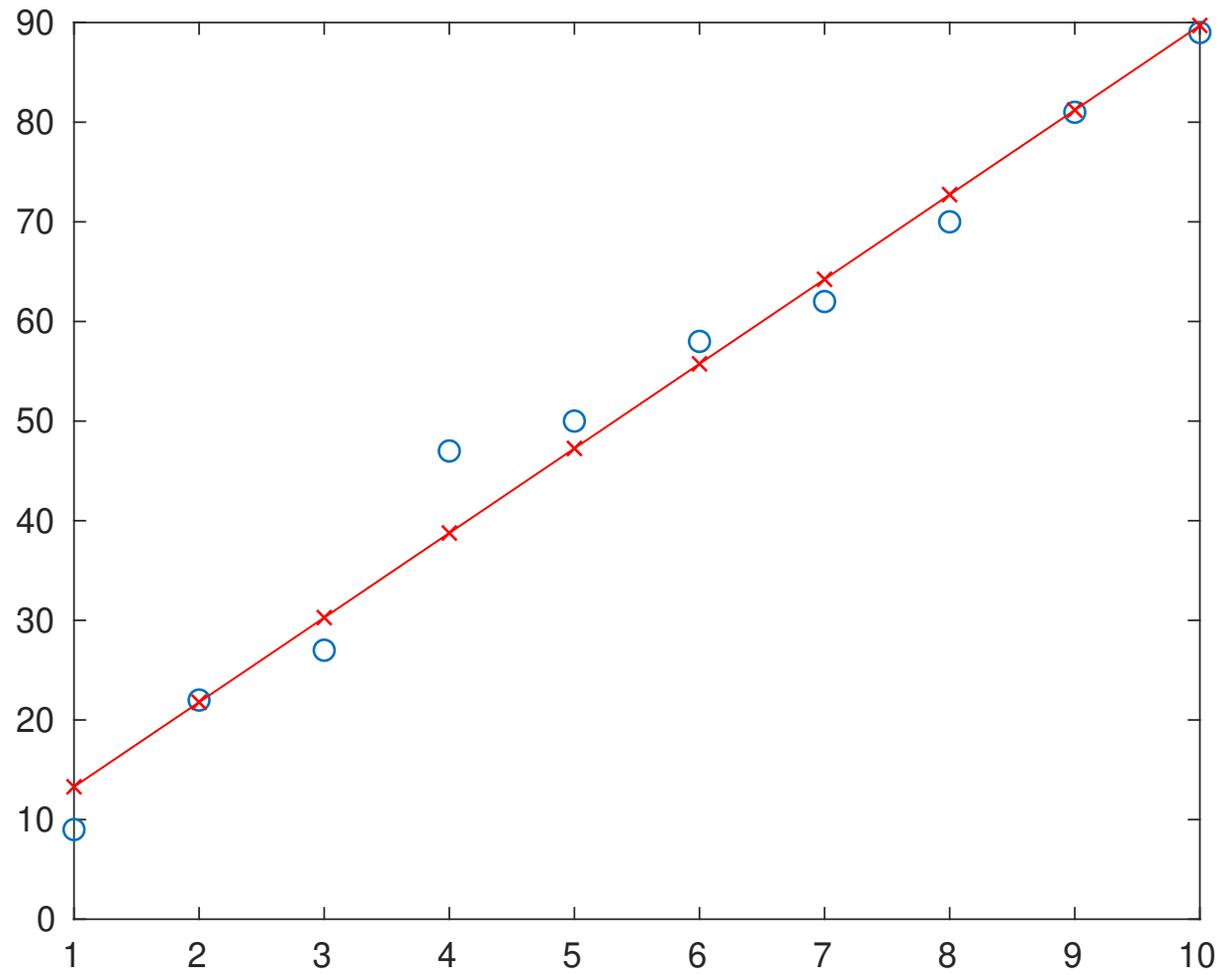
```
figure, plot(y,'o'), pause, hold on, plot(yc,'b'), pause,
```

```
plot(yc1,'r'), pause, plot(yc1,'xr'), pause, close all
```



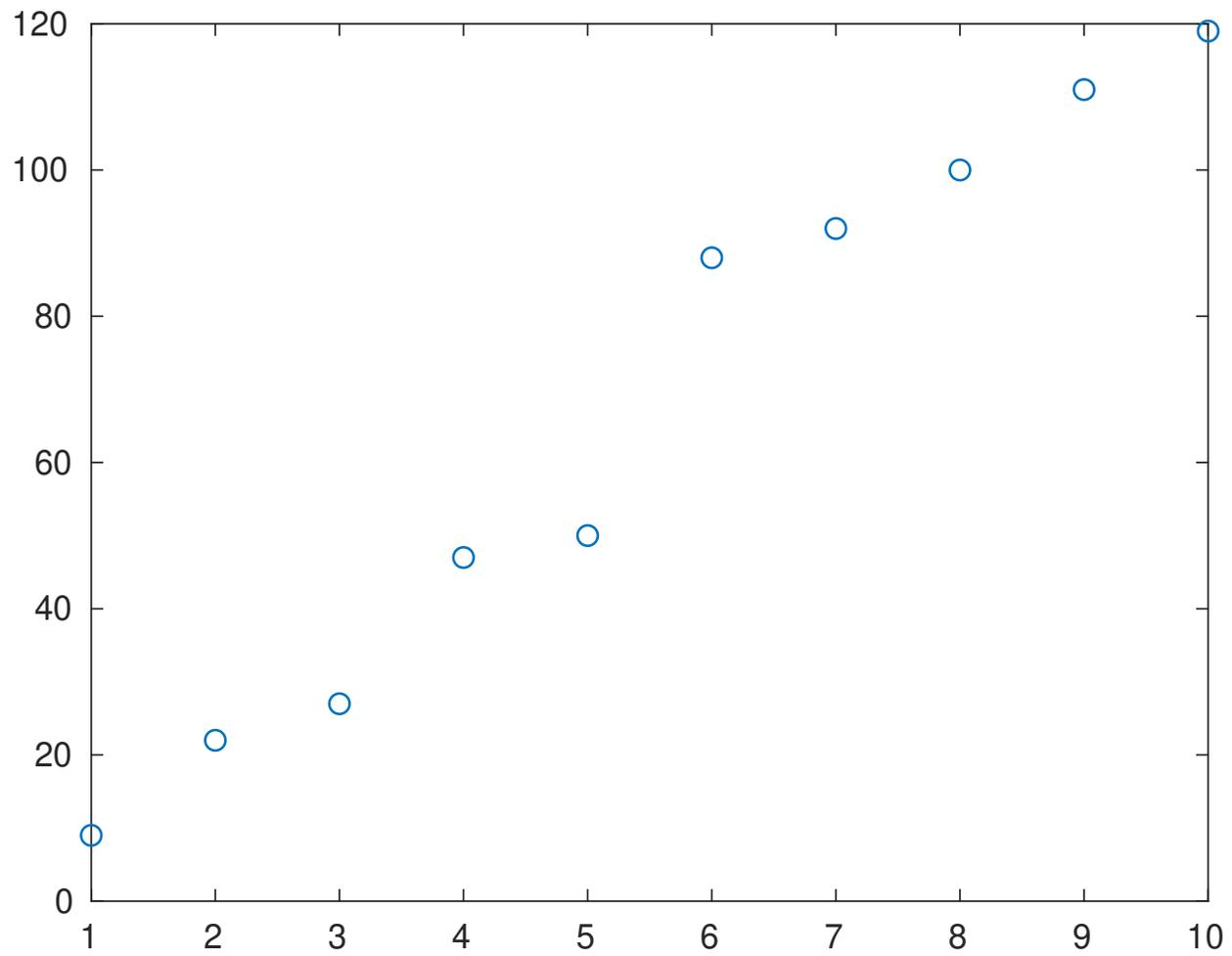


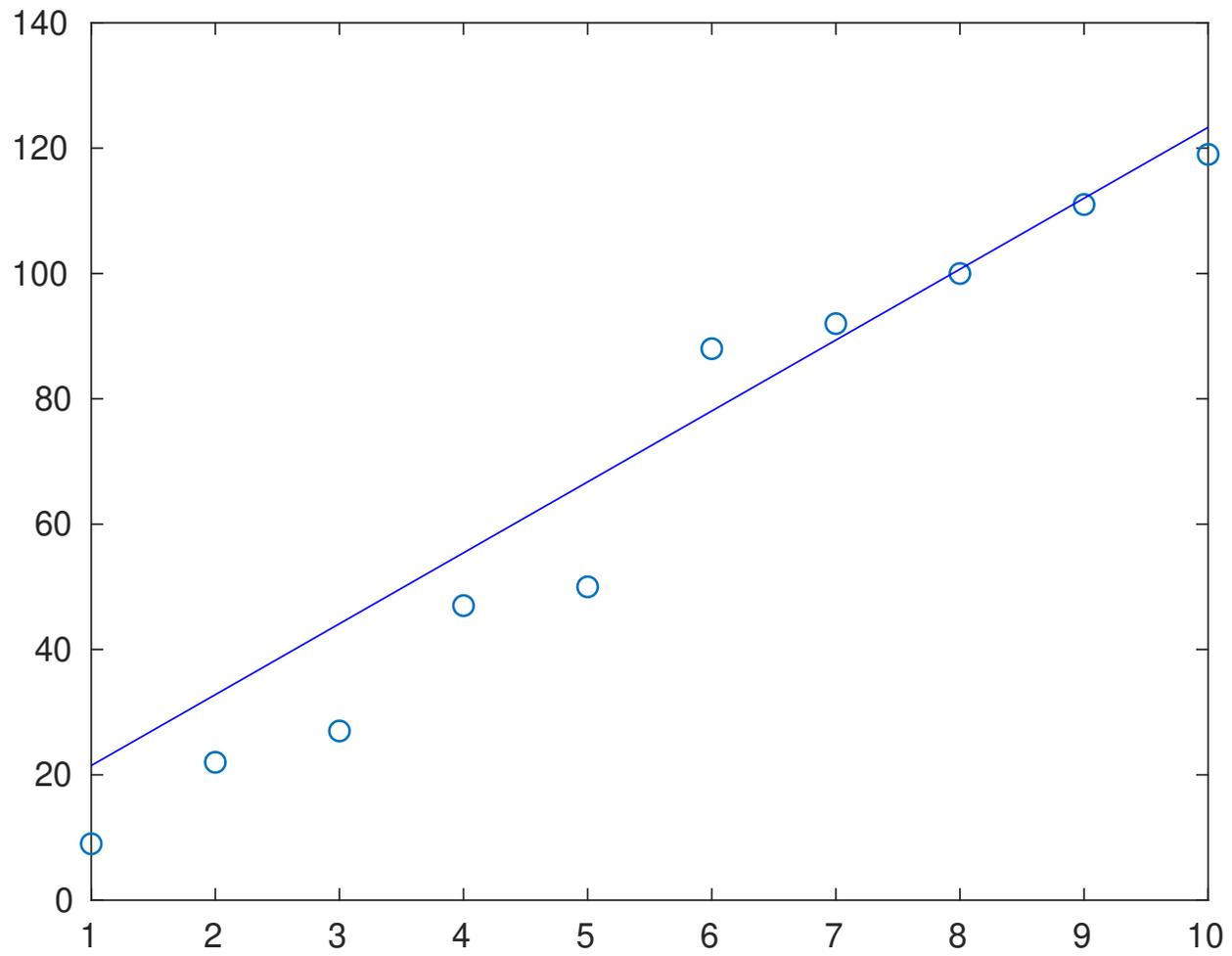


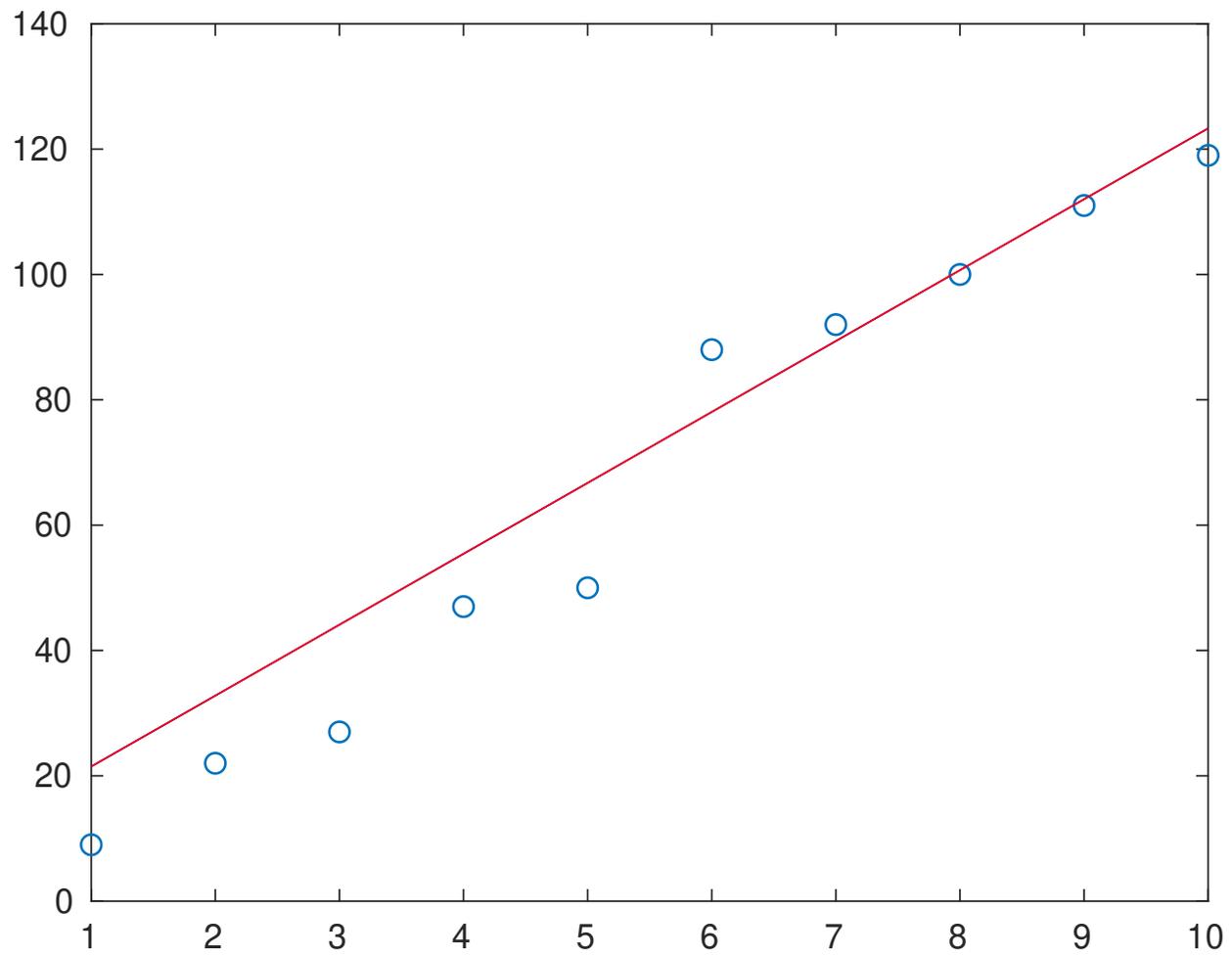


Matlab

```
x = [1 2 3 4 5 6 7 8 9 10];  
  
y = [9 22 27 47 50 58 62 70 81 89];  
  
y(6:10) = y(6:10) + 30;           % NOTA BENE !!!  
  
w = [ones(1,5) ones(1,5)*10];    % NOTA BENE !!!  
  
[a1,a2,xs,ys] = bestfit(x,y,w);  
  
yc = ys + a2 * (x - xs);  
  
yc1 = a1 + a2 * x;  
  
figure, plot(y,'o'), pause, hold on,  
  
plot(yc,'b'), pause, plot(yc1,'r'), pause, close all
```







FINE

Linearizzazione di problemi non lineari

Esempio 1: Modello esponenziale

$$y = \alpha_1 e^{\beta_1 x}$$

Non lineare, crescente per $\beta_1 > 0$, decrescente per $\beta_1 < 0$ ($\beta_1 \neq 0$)

Nota: usato per crescita di una popolazione o decadimento radioattivo

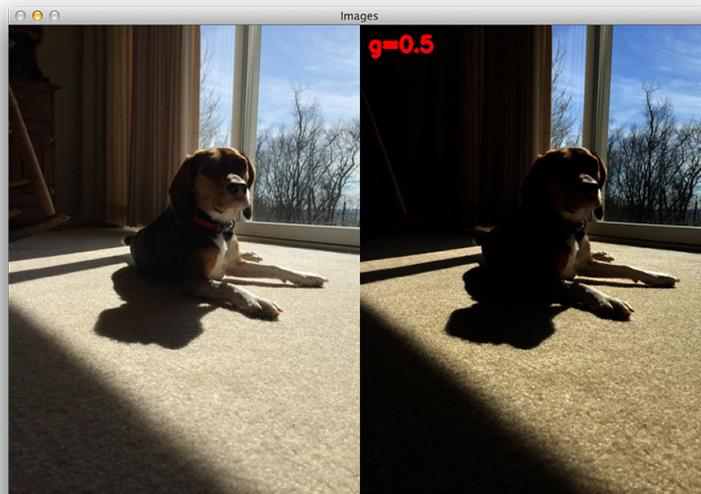
Linearizzazione di problemi non lineari

Esempio 2: Modello potenza

$$y = \alpha_2 x^{\beta_2}$$

Non lineare, con α_2, β_2 costanti e per ($\beta_2 \neq 0$ o $\beta_2 = 1$)

Nota: usato in vari settori (gamma correction in image processing)



Linearizzazione di problemi non lineari

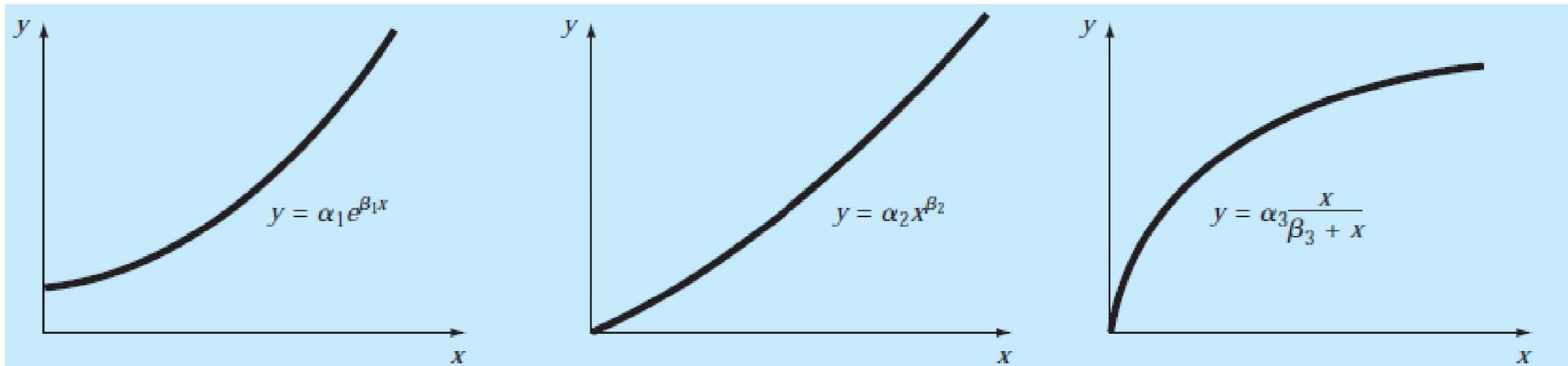
Esempio 3: Modello per la crescita con saturazione

$$y = \alpha_3 \frac{x}{\beta_3 + x}$$

Non lineare, con α_3, β_3 costanti

Nota: usato per crescita con vincoli

Linearizzazione di problemi non lineari



Linearizzazione di problemi non lineari

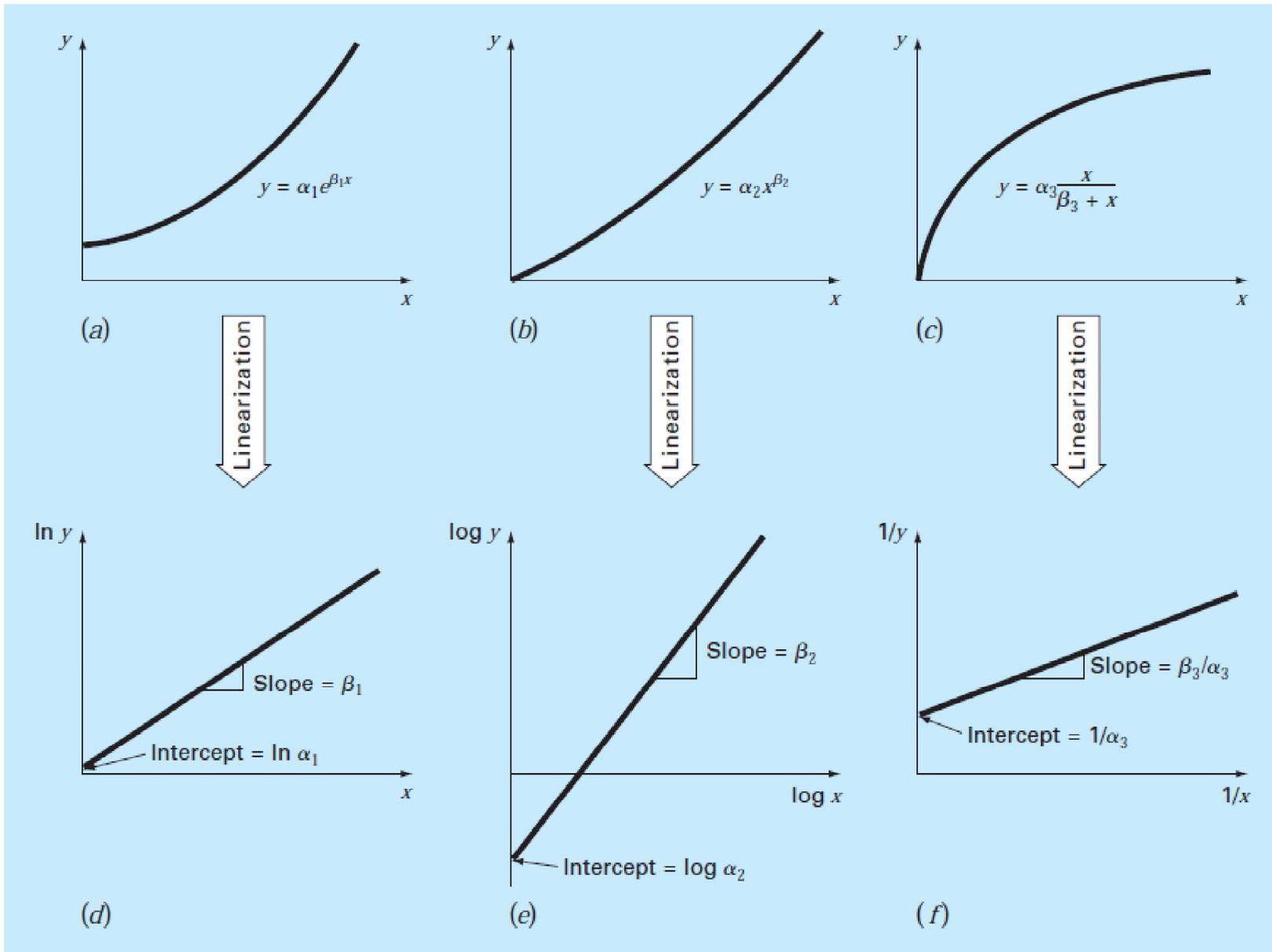
Linearizzazione

$$1) : y = \alpha_1 e^{\beta_1 x} \Rightarrow \ln y = \beta_1 x + \ln \alpha_1$$

$$2) : y = \alpha_2 x^{\beta_2} \Rightarrow \log y = \beta_2 \log x + \log \alpha_2$$

$$3) : y = \alpha_3 \frac{x}{\beta_3 + x} \Rightarrow \frac{1}{y} = \frac{\beta_3}{\alpha_3} \frac{1}{x} + \frac{1}{\alpha_3}$$

Linearizzazione di problemi non lineari



Esempio su equazione potenza

Problema:

x	y	log x	log y
1	0.5	0	-0.301
2	1.7	0.301	0.226
3	3.4	0.477	0.534
4	5.7	0.602	0.753
5	8.4	0.699	0.922

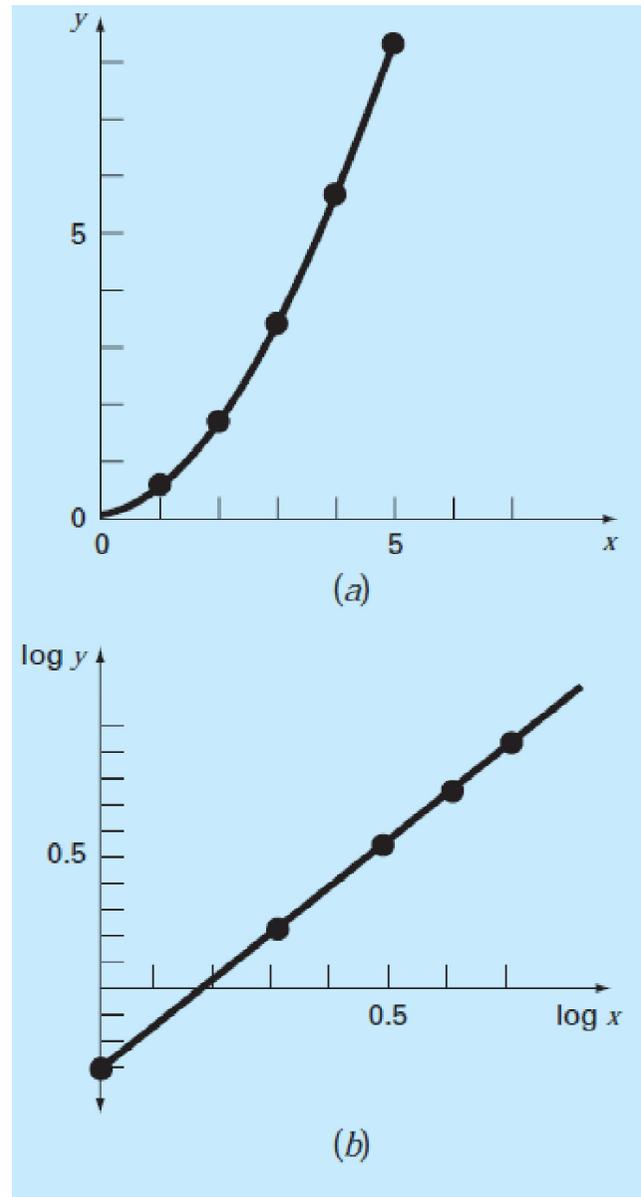
Soluzione:

$$\log y = \beta_2 \log x + \log \alpha_2 \quad \Rightarrow \quad \log y = 1.75 \log x - 0.300$$

Quindi: $\log \alpha_2 = -0.300 \quad \Rightarrow \quad \alpha_2 = 10^{-0.3} = 0.5012, \beta_2 = 1.75$

$$y = 0.5012 x^{1.75}$$

Linearizzazione di problemi non lineari



Esempio Ing. Chimica - BioIngegneria

Crescita di una popolazione

$$\frac{dp}{dt} = kp$$

dove p é la popolazione al tempo t con fattore di crescita k (con dimensionalitá t^{-1}).

La soluzione viene dalla teoria sulle equazioni differenziali:

$$p(t) = p_0 e^{kt}$$

con p_0 la popolazione al tempo $t = 0$.

L'equazione appena scritta porterebbe a:

$$\lim_{t \rightarrow +\infty} p(t) = +\infty$$

Nota: Caso non realistico.

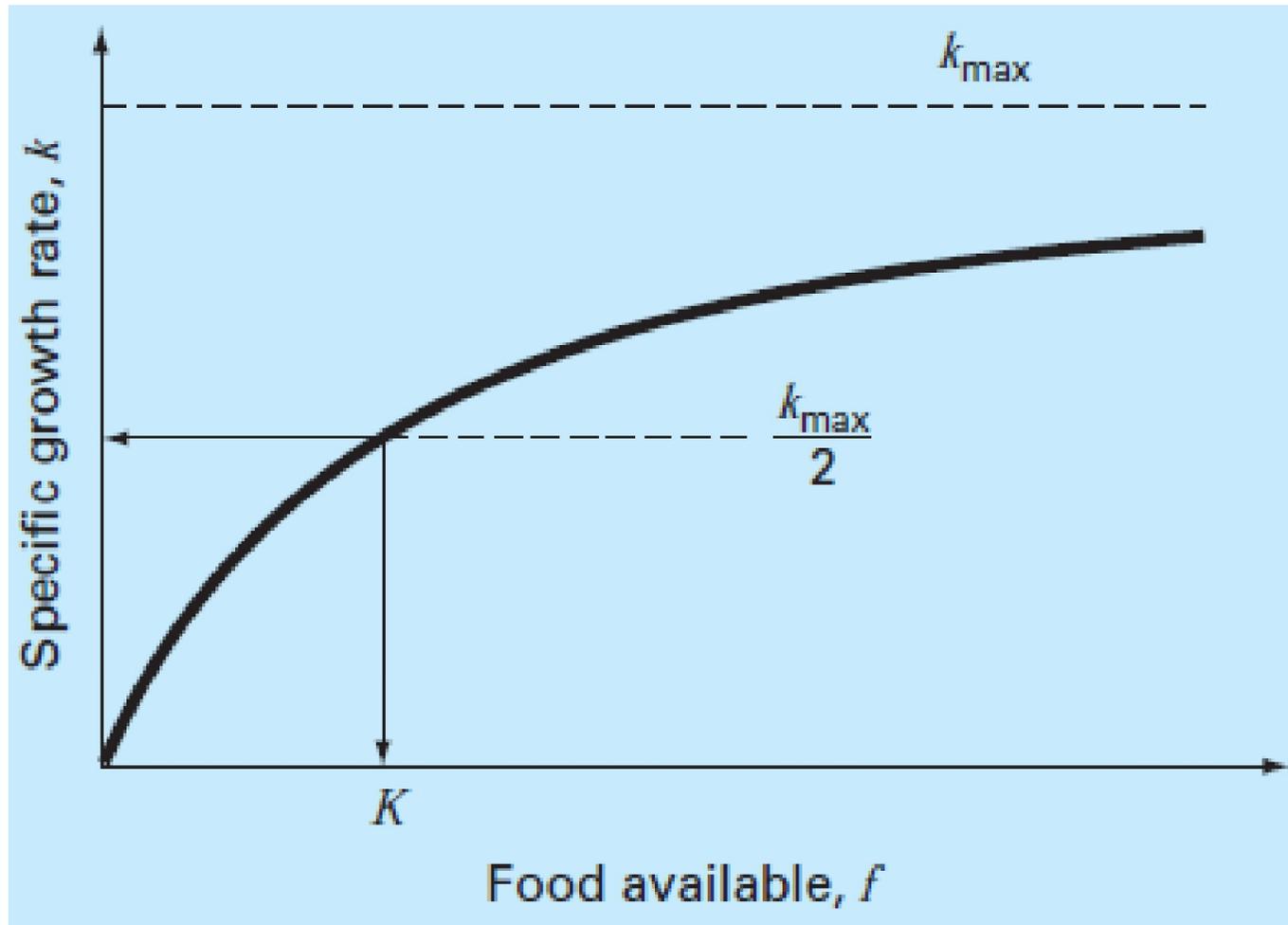
Quindi, generalmente si usa il modello:

$$k = k_{max} \frac{f}{K + f}$$

dove k_{max} é massima crescita ottenibile mentre K é la costante di mezza saturazione:

quando $f = K$ allora $k = k_{max}/2$, cioè:

K é la quantità di cibo che supporta un tasso di crescita pari alla metà del massimo (rate di crescita).



Problema:

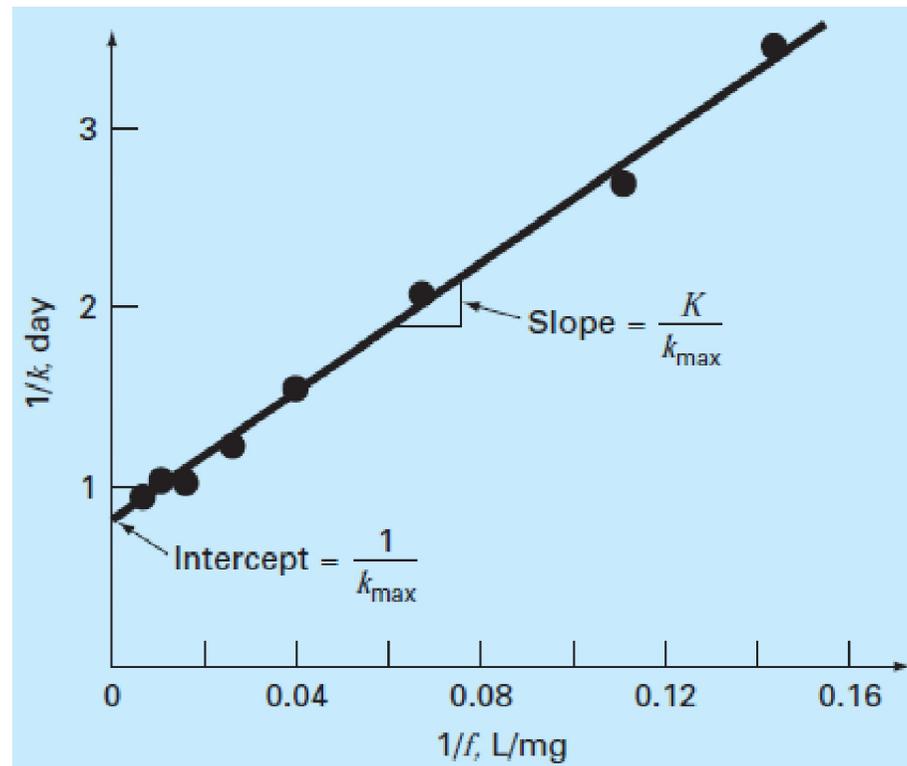
Determinare le costanti K e k_{max} nel caso in cui la popolazione p rappresenti il lievito utilizzato nella produzione commerciale di birra e f é la concentrazione di carbonio necessario alla fermentazione, usare i valori in tabella:

TABLE 20.1 Data used to evaluate the constants for a saturation-growth-rate model to characterize microbial kinetics.

f , mg/L	k , day ⁻¹	$1/f$, L/mg	$1/k$, day
7	0.29	0.14286	3.448
9	0.37	0.11111	2.703
15	0.48	0.06666	2.083
25	0.65	0.04000	1.538
40	0.80	0.02500	1.250
75	0.97	0.01333	1.031
100	0.99	0.01000	1.010
150	1.07	0.00666	0.935

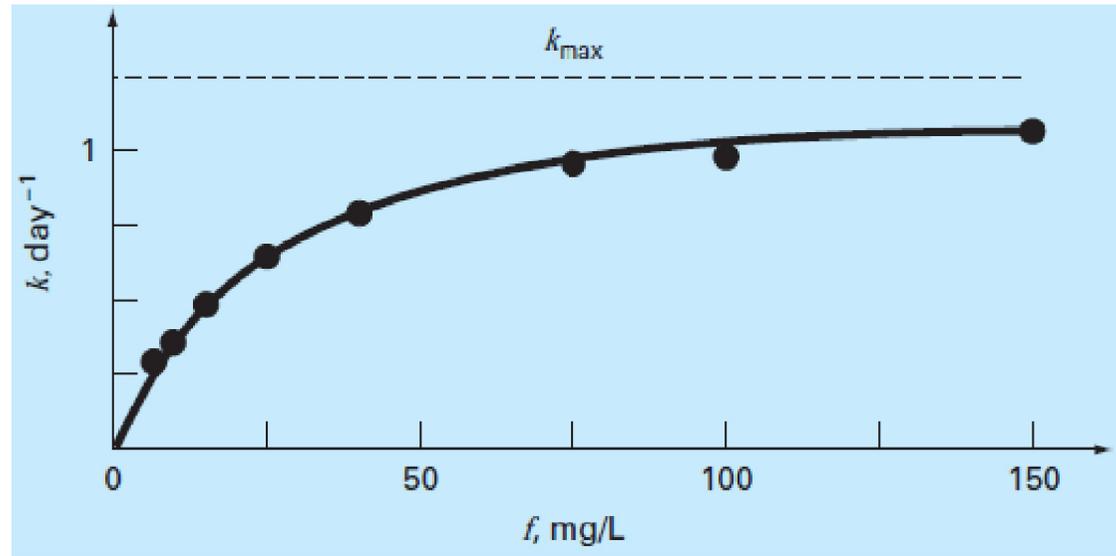
Linearizzando, si ha una retta con pendenza K/k_{max} e intercetta $1/k_{max}$:

$$\frac{1}{k} = \frac{K + f}{k_{max}f} = \frac{K}{k_{max}f} + \frac{1}{k_{max}}$$



che porta alle stime: $k_{max} = 1.23 \text{ day}^{-1}$ e $K = 22.18 \text{ mg/L}$:

ossia



$$\frac{dp}{dt} = 1.23 \frac{f}{22.18 + f} p$$

Nota: somma dei quadrati dei residui = 0.001305.

Minimi quadrati non lineari: Metodo di Gauss-Newton

Esempio: $f(x) = a_0(1 - e^{-a_1x}) + e$

Non può essere linearizzata

ma può essere risolta con un **metodo iterativo**

Idea: linearizzare l'equazione con uno sviluppo in serie di Taylor attorno ai parametri

Il problema può essere scritto:

$$y_i = f(x_i; a_0, a_1, \dots, a_m) + e_i \quad *)$$

dove a_0, a_1, \dots, a_m sono i **parametri** da stimare e e è un **rumore**.

Metodo di Gauss-Newton

Nel caso di due parametri:

$$f(x_i)_{j+1} = f(x_i)_j + \frac{\partial f(x_i)_j}{\partial a_0} \Delta a_0 + \frac{\partial f(x_i)_j}{\partial a_1} \Delta a_1$$

dove j é la stima al passo precedente, $j + 1$ é la predizione, mentre

$$\Delta a_0 = a_{0,j+1} - a_{0,j}, \quad \Delta a_1 = a_{1,j+1} - a_{1,j}$$

e tenendo conto di *) si ha:

$$y_i - f(x_i)_j = \frac{\partial f(x_i)_j}{\partial a_0} \Delta a_0 + \frac{\partial f(x_i)_j}{\partial a_1} \Delta a_1 + e_i$$

Metodo di Gauss-Newton

In forma matriciale: $\{D\} = [Z_j]\{\Delta A\} + E$ dove

$$[Z] = \begin{pmatrix} \frac{\partial f_1}{\partial a_0} & \frac{\partial f_1}{\partial a_1} \\ \frac{\partial f_2}{\partial a_0} & \frac{\partial f_2}{\partial a_1} \\ \dots & \dots \\ \dots & \dots \\ \frac{\partial f_n}{\partial a_0} & \frac{\partial f_n}{\partial a_1} \end{pmatrix} \quad \{D\} = \begin{pmatrix} y_1 - f(x_1) \\ y_2 - f(x_2) \\ \dots \\ \dots \\ y_n - f(x_n) \end{pmatrix} \quad \{\Delta A\} = \begin{pmatrix} \Delta a_0 \\ \Delta a_1 \\ \dots \\ \dots \\ \Delta a_m \end{pmatrix}$$

con n pari alla **cardinalità dell'insieme dei dati**, $\frac{\partial f_i}{\partial a_k}$ la **derivata parziale** rispetto al parametro a_k della funzione in corrispondenza del punto i -esimo e Δa_k **incremento dei valori dei parametri**.

Metodo di Gauss-Newton

Le equazioni normali diventano:

$$[[Z_j]^T [Z_j]] \{\Delta A\} = \{[Z_j]^T D\}$$

che, risolvendo per $\{\Delta A\}$ si usa per il calcolo dei 'nuovi' parametri:

$$a_{0,j+1} = a_{0,j} + \Delta a_0 \quad a_{1,j+1} = a_{1,j} + \Delta a_1$$

La procedura viene ripetuta fino alla convergenza, i.e. la quantità:

$$|\epsilon_a|_k = \left| \frac{a_{k,j+1} - a_{k,j}}{a_{k,j+1}} 100\% \right|$$

risulti minore di una soglia fissata.

Metodo di Gauss-Newton

Nota: Se le derivate della funzione sono difficili, possono essere approssimate:

$$\frac{\partial f_i}{\partial a_k} \simeq \frac{f(x_i; a_0, \dots, a_k + \delta a_k, \dots, a_m) - f(x_i; a_0, \dots, a_k, \dots, a_m)}{\delta a_k}$$

con δ abbastanza piccolo!

Puó accadere che il metodo:

- converga lentamente
- possa oscillare (con cambi di direzione)
- non converga

Si usano tecniche non lineari, ad esempio, minimizzando gli scarti quadratici sistematicamente $\sum_{i=1}^n (y_i - f_i)^2$.

Esercizio

Calcolare il fit della funzione $f(x; a_0, a_1) = a_0(1 - e^{-a_1x})$ sapendo che

x	0.25	0.75	1.25	1.75	2.25
y	0.28	0.57	0.68	0.74	0.79

Supponiamo una stima iniziale $a_0 = 1.0$, $a_1 = 1.0$ (lo somma iniziale dei quadrati dei residui é 0.0248).

Sol.: Calcoliamo

$$\frac{\partial f}{\partial a_0} = 1 - e^{-a_1x} \qquad \frac{\partial f}{\partial a_1} = a_0 x e^{-a_1x}$$

da cui:

$$[Z_0] = \begin{bmatrix} 0.2212 & 0.1947 \\ 0.5276 & 0.3543 \\ 0.7135 & 0.3581 \\ 0.8262 & 0.3041 \\ 0.8946 & 0.2371 \end{bmatrix} \quad [[Z_0]^T [Z_0]] = \begin{bmatrix} 2.3193 & 0.9489 \\ 0.9489 & 0.4404 \end{bmatrix}$$

$$[[Z_0]^T [Z_0]]^{-1} = \begin{bmatrix} 3.6397 & -7.8421 \\ -7.8421 & 19.1678 \end{bmatrix}$$

e il vettore $\{D\}$ contenente le differenze tra misure e predizioni:

$$\{D\} = \left\{ \begin{array}{l} 0.28 - 0.2212 \\ 0.57 - 0.5276 \\ 0.68 - 0.7135 \\ 0.74 - 0.8262 \\ 0.79 - 0.8946 \end{array} \right\} = \left\{ \begin{array}{l} 0.0588 \\ 0.0424 \\ -0.0335 \\ -0.0862 \\ -0.1046 \end{array} \right\}$$

da cui: $[Z_0]^T \{D\} = \begin{bmatrix} -0.1533 \\ -0.0365 \end{bmatrix}$

e risolvendo: $[[Z_j]^T [Z_j]] \{\Delta A\} = \{[Z_j]^T D\}$

si ha: $\Delta A = \begin{Bmatrix} -0.2714 \\ 0.5019 \end{Bmatrix}$

che porta ai nuovi parametri:

$$\begin{Bmatrix} a_0 \\ a_1 \end{Bmatrix} = \begin{Bmatrix} 1.0 \\ 1.0 \end{Bmatrix} + \begin{Bmatrix} -0.2714 \\ 0.5019 \end{Bmatrix} = \begin{Bmatrix} 0.7286 \\ 1.5019 \end{Bmatrix}$$

La somma dei quadrati dei residui risulta ora: 0.0242, mentre $\epsilon_0 = 37\%$, $\epsilon_1 = 33\%$. Il processo può essere iterato (con un opportuno criterio di stop), arrivando a $a_0 = 0.79186$, $a_1 = 1.6751$ con una somma dei quadrati dei residui : 0.000662.