Esempio 1

Nel 1601 Keplero formulò la terza legge del moto planetario

$$T = Cx^{3/2}$$

dove x è la distanza dal sole in milione di km., T è il periodo dell'orbita misurato in giorni e C è una costante. Le coppie (x, T) osservate per gli 8 planeti e Plutone sono

Mercurio	57.59	87.99
Venere	108.11	224.70
Terra	149.57	365.26
Marte	227.84	686.98
Giove	778.14	4332.4
Saturno	1427.0	10759
Urano	2870.3	30684
Nettuno	4499.9	60188
Plutone	5909.0	90719

Determinare C cosicchè $T = Cx^{3/2}$.

Esempio 1

La costante C non è immediatamente calcolabile. Infatti:

```
>> x=[57.59;108.11;149.57;227.84;778.14;...
    1427.0;2870.3;4499.9;5909.0];
>> T=[87.99;224.70;365.26;686.98;4332.4;...
    10759 ;30684 ;60188 ;90719 ];
>> format long e
>> C=T./(x.^(3/2))
C =    2.013319499958850e-01
       1.998960771429300e-01
       1.996803707178690e-01
       1.997556793162881e-01
       1.995914513555481e-01
       1.995886992640107e-01
       1.995358969439935e-01
       1.993910306711459e-01
       1.997226267954996e-01
>>
```

Qual'e' il valore attribuire a C?

Esempio 2

Siano m la magnitudine apparente di una cometa, Δ e r rispettivamente le distanze in unità astronomiche della cometa dalla Terra e dal Sole, g la magnitudine assoluta e ξ un coefficiente. Noto che

$$m = g + 5\log_{10}(\Delta) + \xi \log_{10}(r)$$

e determinati alcuni valori di m, Δ, r, determinare g e ξ. Quindi si desiderano calcolare i *migliori* $A \approx g$, $B \approx \xi$ tali che

$$A + B\log_{10}(r) \approx m - 5\log_{10}(\Delta).$$

Esempio. La tabella seguente riporta le stime di magnitudine visuale m della cometa periodica Wild 2, rilevate da J. Bortle, nochè r e Δ. I valori corrispondenti sono stati calcolati dagli elementi orbitali (IAUC 3177). Calcolare g e ξ.

Esempio 2

1978	UT	m	r	Δ
FEB.	4.01	11.4	1.987	1.249
FEB.	5.00	11.5	1.981	1.252
FEB.	9.02	11.5	1.958	1.266
FEB.	10.02	11.3	1.952	1.270
FEB.	25.03	11.5	1.865	1.335
MAR.	7.07	11.3	1.809	1.382
MAR.	14.03	11.5	1.772	1.415
MAR.	30.05	11.0	1.693	1.487
APR.	3.05	11.1	1.674	1.504
APR.	10.06	10.9	1.643	1.532
APR.	26.07	10.7	1.582	1.592
MAG.	1.08	10.6	1.566	1.610
MAG.	8.07	10.7	1.545	1.634
MAG.	26.09	10.8	1.507	1.696

Plot risultati

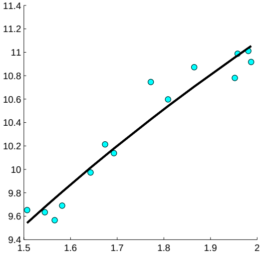

Figura: Grafico che illustra l'approssimazione ai minimi quadrati $A + B\log_{10}(r) \approx m - 5\log_{10}(\Delta)$. Ascissa: r, ordinata: curva e $m - 5\log_{10}(\Delta)$.

Approssimazione ai minimi quadrati

Questi due problemi rientrano nella famiglia più ampia che segue.

- Sia fissata una certa funzione $f : \Omega \subset \mathbb{R} \rightarrow \mathbb{R}$ di cui è noto il valore nei punti $\{x_k\}_{k=1,\ldots,n} \subset \Omega$. Hp. $f \in C(\Omega)$, $\Omega = (a, b)$ anche non limitato (per semplicità , si può generalizzare).

Si cerchino $\mathbf{a} = (a_j)_{j=1,\ldots,m}$ per cui la funzione

$$\psi^*(x) = \sum_{j=1}^{m} a_j \phi_j(x)$$

minimizza tra tutte le funzioni del tipo $\psi(x) = \sum_{j=1}^{m} c_j \phi_j(x)$,

$$\|f - \psi\|_{2,d} := \sqrt{\sum_{k=1}^{n} |f(x_k) - \psi(x_k)|^2} = \sqrt{\sum_{k=1}^{n} |f(x_k) - \sum_{j=1}^{m} c_j \phi_j(x_k)|^2}$$

La funzione ψ^* si dice approssimante ai minimi quadrati (discreti)

di f nello spazio $\mathcal{S} = span(\{\phi_k\}_{k=1,\ldots,m})$.

Approssimazione ai minimi quadrati

Questi due problemi rientrano nella famiglia più ampia che segue.

- Sia fissata una certa funzione $f : \Omega \subset \mathbb{R} \to \mathbb{R}$ di cui è noto il valore nei punti $\{x_k\}_{k=1,\ldots,n} \subset \Omega$. Hp. $f \in C(\Omega)$, $\Omega = (a, b)$ anche non limitato (per semplicità , si può generalizzare).
- Siano date m funzioni lin. indip. $\phi_1, \ldots, \phi_m : \Omega \subset \mathbb{R} \to \mathbb{R}$.

Si cerchino $\mathbf{a} = (a_j)_{j=1,\ldots,m}$ per cui la funzione

$$\psi^*(x) = \sum_{j=1}^{m} a_j \phi_j(x)$$

minimizza tra tutte le funzioni del tipo $\psi(x) = \sum_{j=1}^{m} c_j \phi_j(x)$,

$$\|f - \psi\|_{2,d} := \sqrt{\sum_{k=1}^{n} |f(x_k) - \psi(x_k)|^2} = \sqrt{\sum_{k=1}^{n} |f(x_k) - \sum_{j=1}^{m} c_j \phi_j(x_k)|^2}$$

La funzione ψ^* si dice approssimante ai minimi quadrati (discreti) di f nello spazio $\mathcal{S} = span(\{\phi_k\}_{k=1,\ldots,m})$.

Approssimazione ai minimi quadrati e sistemi lineari

Posto $\|\mathbf{x}\|_2 := \sqrt{\sum_{k=1}^{n} |x_k|^2}$ e definiti

- V la matrice $n \times m$ le cui componenti sono $V_{i,j} = \phi_j(x_i)$,
- $\mathbf{y}$ il vettore $n \times 1$ le cui componenti sono $y_k = f(x_k)$,
- $\mathbf{a}$ il vettore $m \times 1$ le cui componenti sono a_k,

il problema di approssimazione ai minimi quadrati consiste nel risolvere il sistema sovradeterminato $V\mathbf{a} = \mathbf{y}$ cosicchè sia minima $\|V\mathbf{a} - \mathbf{y}\|_2$ in quanto

$$\begin{aligned}\|V\mathbf{a} - \mathbf{y}\|_2 &= \sqrt{\sum_{k=1}^{n} |y_k - (V\mathbf{a})_k|^2} = \sqrt{\sum_{k=1}^{n} |y_k - \sum_{j=1}^{m} V_{k,j} a_j|^2} \\ &= \sqrt{\sum_{k=1}^{n} |f(x_k) - \sum_{j=1}^{m} a_j \phi_j(x_k)|^2} = \|f - \sum_{j=1}^{m} a_j \phi_j\|_{2,d}\end{aligned}$$

NB: Si osservi che V è rettangolare e il problema $V\mathbf{a} = \mathbf{y}$ potrebbe non avere sol. classica $\mathbf{a}$.

Funzione peso e prodotto scalare

Al momento non risulta chiaro come risolvere questo sistema sovradeterminato $V\mathbf{a} = \mathbf{y}$ cosicchè sia minima $\|V\mathbf{a} - \mathbf{y}\|_2$.

Sia $w : \Omega = (a, b) \rightarrow \mathbb{R}$ una funzione peso (continua, strettamente positiva in (a, b), singolarità integrabili agli estremi sono ammesse).

- Jacobi: $w(x) = (1 - x)^{\alpha}(1 + x)^{\beta}$, $x \in (-1, 1)$, $\alpha, \beta > -1$. Casi speciali: $\alpha = \beta = 0$ (Legendre), $\alpha = \beta = -1/2$ (Chebyshev), $\alpha = \beta$ (Gegenbauer).

Due classici prodotti scalari di due funzioni *w-integrabili* sono

- $(f, g)_c := \int_{\Omega} f(x)g(x)w(x)dx$, caso continuo;

Proprietá: commutativa, linearitá, $(f, f) \geq 0$ per ogni f.

Funzione peso e prodotto scalare

Al momento non risulta chiaro come risolvere questo sistema sovradeterminato $V\mathbf{a} = \mathbf{y}$ cosicchè sia minima $\|V\mathbf{a} - \mathbf{y}\|_2$.

Sia $w : \Omega = (a, b) \rightarrow \mathbb{R}$ una funzione peso (continua, strettamente positiva in (a, b), singolarità integrabili agli estremi sono ammesse).

- Jacobi: $w(x) = (1-x)^{\alpha}(1+x)^{\beta}$, $x \in (-1, 1)$, $\alpha, \beta > -1$. Casi speciali: $\alpha = \beta = 0$ (Legendre), $\alpha = \beta = -1/2$ (Chebyshev), $\alpha = \beta$ (Gegenbauer).
- Gauss: $w(x) = \exp(-x^2)$, $x \in \mathbb{R}$.

Due classici prodotti scalari di due funzioni *w-integrabili* sono

- $(f, g)_c := \int_{\Omega} f(x)g(x)w(x)dx$, caso continuo;
- $(f, g)_d := \sum_{i=1}^{n} f(x_i)g(x_i)w_i$, caso discreto.

  I coeffs. w_i del caso discreto dipendono dalla funz. peso w.

Proprietá: commutativa, linearitá, $(f, f) \geq 0$ per ogni f.

Definiamo $\|f\| = \sqrt{(f,f)}$. Osserviamo che:

- $\|\mathbf{x}\|_2 := \sqrt{\sum_{k=1}^{n} |x_k|^2} = \sqrt{(\mathbf{x},\mathbf{x})_{\mathbb{R}^n}}$ dove $(\cdot,\cdot)_{\mathbb{R}^n}$ è l'usuale prodotto scalare di $\mathbb{R}^n$.
- $\|f-\psi\|_{2,d} := \sqrt{\sum_{k=1}^{n} |f(x_k)-\psi(x_k)|^2} = \sqrt{(f-\psi, f-\psi)_d}$

Alcune definizioni ulteriori.

- Due funzioni f, g si dicono ortogonali se $(f,g) = 0$.
- Una sequenza finita o infinita di funzioni $\{\phi_k\}_{k=1,\ldots}$ si dice sistema ortogonale se $(\phi_i,\phi_j) = 0$ per $i \neq j$, $(\phi_i,\phi_i) > 0$.
- Un sistema ortogonale $\{\phi_k\}_{k=1,\ldots}$ per cui $(\phi_i,\phi_i) = 1$ si dice sistema ortonormale.

Riassunto

- Dobbiamo cercare una funzione $\psi^* = \sum_{k=1}^{m} \phi_k$ che meglio approssimi f nei punti 2 a 2 distinti $\{x_k\}$, nel senso di minimizzare

$$\|f - \psi\|_{2,d} := \sqrt{\sum_{k=1}^{n} |f(x_k) - \psi(x_k)|^2} = \sqrt{(f - \psi, f - \psi)_d},$$

con $(f, g)_d := \sum_{i=1}^{n} f(x_i)g(x_i)$, quindi per $w_i = 1$.

Riassunto

- Dobbiamo cercare una funzione $\psi^* = \sum_{k=1}^{m} \phi_k$ che meglio approssimi f nei punti 2 a 2 distinti $\{x_k\}$, nel senso di minimizzare

$$\|f - \psi\|_{2,d} := \sqrt{\sum_{k=1}^{n} |f(x_k) - \psi(x_k)|^2} = \sqrt{(f - \psi, f - \psi)_d},$$

  con $(f, g)_d := \sum_{i=1}^{n} f(x_i)g(x_i)$, quindi per $w_i = 1$.
- Il problema si converte in una questione di algebra lineare: risolvere il sistema sovradeterminato $V\mathbf{a} = \mathbf{y}$ cosicchè sia minima

$$\|V\mathbf{a} - \mathbf{y}\|_2 = \sqrt{(V\mathbf{a} - \mathbf{y}, V\mathbf{a} - \mathbf{y})_{\mathbb{R}^n}}.$$

Alcuni teoremi

Teorema. [Pitagora] Se $(f, g) = 0$ allora

$$\|f + g\|^2 = \|f\|^2 + \|g\|^2.$$

Teorema. Se $\phi_1, \ldots, \phi_n$ è un sistema ortogonale, allora i ϕ_j sono linearmente indipendenti.

Teorema. Se $\phi_1, \ldots, \phi_n$ è un sistema ortogonale, allora

$$\|\sum_{j=1}^{m} c_j \phi_j\|^2 = (\sum_{j=1}^{m} c_j \phi_j, \sum_{k=1}^{m} c_k \phi_k) = \sum_{j=1}^{m} c_j^2 \|\phi_j\|^2$$

per ogni $\{c_j\}_{j=1,\ldots,m}$.

Teorema. Siano $\{\phi_j\}_{1,\ldots,m} \in C(\Omega)$ funzioni linearmente indipendenti. Allora la soluzione del problema

$$\|f - \psi^*\|_2 = \min_{\psi \in \text{span}\{\phi_j\}_{j=1,\ldots,m}} \|f - \psi\|_2$$

è

$$\psi^* = \sum_{1,\ldots,m} c_j^* \phi_j$$

dove i coefficienti c_j^* verificano le cosidette equazioni normali

$$\sum_{k=1}^{m} (\phi_j, \phi_k) c_k^* = (\phi_j, f), \; j = 1, \ldots, m.$$

La soluzione è caratterizzata dalla proprietà di ortogonalitá cioè che $\psi^* - f$ è ortogonale a tutti gli ϕ_k, con $k = 1, \ldots, m$, cioè

$$(\psi^*, \phi_k) = (f, \phi_k), \; k = 1, \ldots, m. \tag{1}$$

Teorema miglior approssimazione minimi quadrati: dim.

Sia $c = (c_k)_{1,\ldots,m}$, una vettore di coefficienti e supponiamo che per almeno un indice j sia $c_j \neq c_j^*$, cioè $c \neq c^* = (c_k^*)_{1,\ldots,m}$. Allora

$$\begin{aligned} \sum_{j=1}^{m} c_j \phi_j - f &= \left(\sum_{j=1}^{m} c_j \phi_j - \psi^* \right) + (\psi^* - f) \\ &= \sum_{j=1}^{m} (c_j - c_j^*) \phi_j + (\psi^* - f) \end{aligned} \quad (2)$$

Se $u = \psi^* - f$ è ortogonale a tutti i ϕ_j, allora è ortogonale pure alla combinazione lineare di ϕ_j come ad esempio

$$v = \sum_{j=1}^{m} (c_j - c_j^*) \phi_j = \sum_{j=1}^{m} c_j \phi_j - \psi^* \in \text{ span } \{\phi_k\}_{k=1,\ldots,m}.$$

Teorema miglior approssimazione minimi quadrati: dim.

Dal teorema di Pitagora, poichè $(u, v) = 0$ implica $\|u + v\|^2 = \|u\|^2 + \|v\|^2$, da $\|\sum_{j=1}^m (c_j - c_j^*)\phi_j\| > 0$ poichè $c \neq c^*$ abbiamo

$$
\begin{aligned}
\|\sum_{j=1}^{m} c_j\phi_j - f\|^2 &= \|\left(\sum_{j=1}^{m} c_j\phi_j - \psi^*\right) + (\psi^* - f)\|^2 \\
&= \|\sum_{j=1}^{m} c_j\phi_j - \psi^*\|^2 + \|\psi^* - f\|^2 \\
&= \|\sum_{j=1}^{m} (c_j - c_j^*)\phi_j\|^2 + \|\psi^* - f\|^2 \\
&> \|\psi^* - f\|^2 \qquad (3)
\end{aligned}
$$

Teorema miglior approssimazione minimi quadrati: dim.

Di conseguenza se $\psi^* \in \text{span}\{\phi_k\}_{k=1,\ldots,m}$ e $\psi^* - f$ è ortogonale a tutti i ϕ_k allora ψ^* è la miglior approssimazione di f in $\text{span}\{\phi_k\}_{k=1,\ldots,m}$. Rimane allora da mostrare che le condizioni di ortogonalità

$$\left(\sum_{j=1}^{m} c_j^* \phi_j - f, \phi_k\right) = 0, \quad k = 1, \ldots, m$$

possano essere soddisfatte per un qualche $c^* = (c_j)_{j=1,\ldots,m}$. Questo problema è equivalente alla soluzione del sistema di equazioni normali

$$\sum_{k=1}^{m} (\phi_j, \phi_k) c_k^* = (\phi_j, f), \; j = 1, \ldots, m \tag{4}$$

Teorema miglior approssimazione minimi quadrati: dim.

Se $\phi_1, \ldots, \phi_m$ sono m vettori linearmente indipendenti che formano un sistema ortogonale, si ha che

$$\sum_{k=1}^{m} (\phi_j, \phi_k) c_k^* = (\phi_j, \phi_j) c_j^*$$

e quindi da (**??**) che

$$(\phi_k, \phi_k) c_k^* = (\phi_k, f).$$

Visto che $(\phi_k, \phi_k) \neq 0$ (se così non fosse $0 = (\phi_k, \phi_k) = \|\phi_k\|^2$ avremmo $\phi_k = 0$ e quindi $\{\phi_j\}_{j=1,\ldots,m}$ non sarebbe un sistema di vettori linearmente indipendenti) si vede subito che $(c_k^*)_k$ esistono unici e uguali a

$$c_k^* = \frac{(\phi_k, f)}{(\phi_k, \phi_k)}.$$

Se invece $\phi_1, \ldots, \phi_m$ non formano un sistema ortogonale, il sistema di equazioni normali ha una e una sola soluzione se il sistema omogeneo di equazioni

$$\sum_{k=1}^{m} (\phi_j, \phi_k) c_k^* = 0, \; j = 1, \ldots, m \tag{5}$$

ha la sola soluzione nulla.

Teorema miglior approssimazione minimi quadrati: dim.

Se così non fosse, esisterebbe $c^* = (c_j)_{j=1,\ldots,m}$ per cui da (**??**)

$$\begin{aligned}
\|\sum_{j=1}^{m} c_j \phi_j\|^2 &= \left(\sum_{j=1}^{m} c_j \phi_j, \sum_{k=1}^{m} c_k \phi_k\right) \\
&= \sum_{k=1}^{m} c_k \sum_{j=1}^{m} c_j \, (\phi_j, \phi_k) \\
&= \sum_{k=1}^{m} c_k \cdot 0 \\
&= 0 \qquad (6)
\end{aligned}$$

e quindi essendo $\|\cdot\|$ una norma, necessariamente $\sum_{j=1}^{m} c_j^* \phi_j = 0$, il che contraddice il fatto che i ϕ_k erano linearmente indipendenti.

Corollario miglior approssimazione minimi quadrati

Teorema. Se $\{\phi_j\}_{j=1,\ldots,m}$ è un sistema ortogonale allora i coefficienti c_j^* (detti in questo caso *di Fourier*) sono calcolabili più semplicemente con la formula

$$c_j^* = \frac{(f,\phi_j)}{(\phi_j,\phi_j)}, \;\; j = 1,\ldots,m.$$

Risoluzione minimi quadrati iniziale

- Si determina una base ortogonale $\{\phi_k^{\perp}\}$ dello spazio vett. $\mathcal{S} = span(\{\phi_k\}_{k=1,\ldots,m})$ (algoritmo di Gram-Schmidt);

NB: Quando visto si può generalizzare, con le dovute cautele, in ambiti più astratti e a certi spazi $\mathcal{S}$ che non sono di dimensione finita.

Risoluzione minimi quadrati iniziale

- Si determina una base ortogonale $\{\phi_k^{\perp}\}$ dello spazio vett. $\mathcal{S} = span(\{\phi_k\}_{k=1,\ldots,m})$ (algoritmo di Gram-Schmidt);
- Si calcola $c_j^* = \frac{(f,\phi_j)}{(\phi_j,\phi_j)}, \;\; j = 1, \ldots, m$;

NB: Quando visto si può generalizzare, con le dovute cautele, in ambiti più astratti e a certi spazi $\mathcal{S}$ che non sono di dimensione finita.

Risoluzione minimi quadrati iniziale

- Si determina una base ortogonale $\{\phi_k^{\perp}\}$ dello spazio vett. $\mathcal{S} = span(\{\phi_k\}_{k=1,\ldots,m})$ (algoritmo di Gram-Schmidt);
- Si calcola $c_j^* = \frac{(f,\phi_j)}{(\phi_j,\phi_j)}$, $j = 1, \ldots, m$;
- Si ottiene $\psi^* = \sum_j c_j^* \phi_k^{\perp}$.

NB: Quando visto si può generalizzare, con le dovute cautele, in ambiti più astratti e a certi spazi $\mathcal{S}$ che non sono di dimensione finita.

Minimi quadrati in Matlab/Octave

Per risolvere il problema ai minimi quadrati in Matlab (sistema sovradeterminato), basta usare il comando \.

Risolviamo l'esempio 1, salvando nel file `esempio1.m` il seguente codice.

```
x=[57.59;108.11;149.57;227.84;778.14;...
    1427.0;2870.3;4499.9;5909.0];
T=[87.99;224.70;365.26;686.98;4332.4;...
    10759 ;30684 ;60188 ;90719 ];
%T=C x^{3/2}.
V=x.^(3/2); C=V\T;
format long e;
C
[T C*V]
```

Risultati esempio 1

```
% dati=[UT m r delta]
dati=[ 4.01   11.4    1.987    1.249
       5.00   11.5    1.981    1.252
       9.02   11.5    1.958    1.266
       10.02  11.3    1.952    1.270
       25.03  11.5    1.865    1.335
       7.07   11.3    1.809    1.382
       14.03  11.5    1.772    1.415
       30.05  11.0    1.693    1.487
       3.05   11.1    1.674    1.504
       10.06  10.9    1.643    1.532
       26.07  10.7    1.582    1.592
       1.08   10.6    1.566    1.610
       8.07   10.7    1.545    1.634
       26.09  10.8    1.507    1.696  ];
UT=dati(:,1);m=dati(:,2);r=dati(:,3);delta=dati(:,4);
% g+x*log(r)=m-5*log(delta), DETERMINARE g,x.
% phi1(x)=1, phi2(x)=log(r), c1=g, c2=x.
V=[ones(size(UT)) log10(r)]; y=m-5*log10(delta); c=V\y;
format long e; c, [m-5*log10(delta) V*c]
```

Risultati esempio 2

```
>> esempio2
c =
      7.302213778674195e+00
      1.257799522762583e+01
ans =
      1.091718780812932e+01     1.105294512806965e+01
      1.101197835562795e+01     1.103642529515424e+01
      1.098783147159332e+01     1.097263236899629e+01
      1.078098139522022e+01     1.095586748446904e+01
      1.087259367149703e+01     1.070681088792163e+01
      1.059745978480910e+01     1.054027484404377e+01
      1.074621780069845e+01     1.042738923226925e+01
      1.013844515739023e+01     1.017825990653182e+01
      1.021376081872188e+01     1.011660880693023e+01
      9.973706173517074e+00     1.001450202245726e+01
      9.690284682991749e+00     9.807831922881672e+00
      9.565870619840751e+00     9.752303577681410e+00
      9.633739739018015e+00     9.678555345780794e+00
      9.652870760396526e+00     9.542521416264551e+00
>>
```

Digitiamo sulla shell di Matlab

```
>> x=0:0.01:2*pi;
>> y=sin(2*x)+(10^(-1))*rand(size(x));
>> plot(x,y,'r-');
```

- Interpretazione: perturbazione della funzione $\sin(2x)$ nell'intervallo $[0, 2\pi]$.

Minimi quadrati e smoothing

Digitiamo sulla shell di Matlab

```
>> x=0:0.01:2*pi;
>> y=sin(2*x)+(10^(-1))*rand(size(x));
>> plot(x,y,'r-');
```

- Interpretazione: perturbazione della funzione $\sin(2x)$ nell'intervallo $[0, 2\pi]$.
- Necessitá: ricostruire $\sin(2x)$ (e non funz. perturbata).

Digitiamo sulla shell di Matlab

```
>> x=0:0.01:2*pi;
>> y=sin(2*x)+(10^(-1))*rand(size(x));
>> plot(x,y,'r-');
```

- Interpretazione: perturbazione della funzione $\sin(2x)$ nell'intervallo $[0, 2\pi]$.
- Necessitá: ricostruire $\sin(2x)$ (e non funz. perturbata).
- Nota: non ha senso utilizzare un interpolante polinomiale p di grado n nè una spline interpolante visto che ricostruirebbero la funzione perturbata.

Minimi quadrati e polyfit

Scriviamo sulla shell di Matlab/Octave `help polyfit`. In una recente release di Matlab appare

```
POLYFIT Fit polynomial to data.
   POLYFIT(X,Y,N) finds the coefficients of a
      polynomial P(X)
   of degree N that fits the data, P(X(I))~=Y(I), in a
       least
   -squares sense.
```

In altri termini `polyfit` calcola i coefficienti del polinomio p_N di grado N che *meglio* approssima (in norma 2 discreta) la funzione f avente nel vettore di nodi X i valori Y (cioè $Y(i) := f(X(i))$). Operativamente si cerca il polinomio p_N per cui risulta minima

$$\|f - p_N\|_{2,d} = \sqrt{\sum_i |f(x_i) - p_N(x_i)|^2}.$$

- Posto $f(x) = \sin(x)$ e $\tilde{f}(x) = \sin(x) + \delta(x)$, confrontiamo graficamente per $n = 2, \ldots, 8$, nei nodi $x_k = kh$, $h = 2\pi/999$, $k = 0, \ldots, 999$, la funzione perturbata $f(x) = \sin(x) + \delta(x)$ con la approssimante ai minimi quadrati p_n^*.

Applicazione `polyfit`

- Posto $f(x) = \sin(x)$ e $\tilde{f}(x) = \sin(x) + \delta(x)$, confrontiamo graficamente per $n = 2, \ldots, 8$, nei nodi $x_k = kh$, $h = 2\pi/999$, $k = 0, \ldots, 999$, la funzione perturbata $f(x) = \sin(x) + \delta(x)$ con la approssimante ai minimi quadrati p_n^*.
- Valutiamo per $n = 2, \ldots, 8$ la quantità $\|f - p_n^*\|_{2,d}$ e $\|\tilde{f} - p_n^*\|_{2,d}$.

Applicazione `polyfit`

- Posto $f(x) = \sin(x)$ e $\tilde{f}(x) = \sin(x) + \delta(x)$, confrontiamo graficamente per $n = 2, \ldots, 8$, nei nodi $x_k = kh$, $h = 2\pi/999$, $k = 0, \ldots, 999$, la funzione perturbata $f(x) = \sin(x) + \delta(x)$ con la approssimante ai minimi quadrati p_n^*.
- Valutiamo per $n = 2, \ldots, 8$ la quantità $\|f - p_n^*\|_{2,d}$ e $\|\tilde{f} - p_n^*\|_{2,d}$.
- Valutiamo per $n = 2, \ldots, 8$ la quantità $\|f - p_n^*\|_{\infty,d}$ e $\|\tilde{f} - p_n^*\|_{\infty,d}$, dove ricordiamo

$$\|g\|_{\infty,d} = \max_{k=0,\ldots,999} |g(x_k)|.$$

Applicazione polyfit

Salviamo in esempio3.m:

```
x=linspace(0,2*pi,1000);
y=sin(2*x); yy=y+(10^(-2))*rand(size(x));
for n=2:8
    coeff=polyfit(x,yy,n); % COEFFS. BEST APPROX (B.A.)
    z=polyval(coeff,x);    % VALORE B.A. NEI NODI "x".
    plot(x,yy,'r-',x,z,'k-');
    err2=norm(z-y,2); err2p=norm(z-yy,2); % ERRS.
    errinf=norm(z-y,inf); errinfp=norm(z-yy,inf); %
        ERRS.
    fprintf('\n\t[DEG]:%2.0f',n);
    fprintf('  [2]:%2.2e %2.2e',err2,err2p);
    fprintf('  [INF]:%2.2e %2.2e',errinf,errinfp);
    pause(2);
end
fprintf('\n \n');
```

Risultati

```
>> % [ERR.][2]: SIN.-SIN PERT. [INF]: SIN.-SIN PERT.
>> esempio3

 [DEG]: 2 [2]:2.06e+01 2.06e+01 [INF]:1.13e+00 1.13e+00
 [DEG]: 3 [2]:1.89e+01 1.89e+01 [INF]:1.16e+00 1.16e+00
 [DEG]: 4 [2]:1.89e+01 1.89e+01 [INF]:1.16e+00 1.16e+00
 [DEG]: 5 [2]:6.86e+00 6.86e+00 [INF]:6.67e-01 6.64e-01
 [DEG]: 6 [2]:6.86e+00 6.86e+00 [INF]:6.68e-01 6.64e-01
 [DEG]: 7 [2]:1.23e+00 1.22e+00 [INF]:1.48e-01 1.44e-01
 [DEG]: 8 [2]:1.23e+00 1.22e+00 [INF]:1.47e-01 1.44e-01

>>
```

Plot risultati

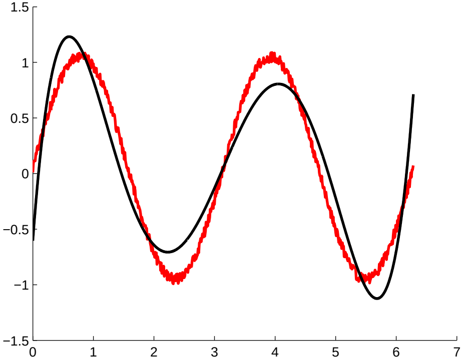

Figura: Grafico che illustra l'approssimazione ai minimi quadrati di grado 5 su una *perturbazione* della funzione $\sin(2x)$ (campionamento in nodi equispaziati)).

Nota: `polyfit` e interpolazione

Supponiamo fissati i punti $\{x_k\}_{k=1,\ldots,m}$ (a due a due distinti) e sia p_{m-1} il polinomio che interpola le coppie $(x_k, f(x_k))$ per $k = 1, \ldots, m$ Evidentemente da $f(x_i) = p_{m-1}(x_i)$ per $i = 1, \ldots, m$ abbiamo

$$\|f - p_{m-1}\|_{2,d} = \sqrt{\sum_{i=1}^{m} |f(x_i) - p_{m-1}(x_i)|^2} = 0,$$

e quindi il polinomio interpolatore risulta la approssimante ai minimi quadrati di f (relativa alla norma 2 discreta basata sui punti $\{x_k\}_{k=1,\ldots,m}$).

Di conseguenza, il comando `coeffs=polyfit(x,y,n-1)` darà i coefficienti del polinomio interpolatore qualora i vettori x, y abbiano dimensione n.